Journal of Advanced Transportation

Autonomous Transportation Systems Enabled by Emerging Technologies

Lead Guest Editor: Linlin You
Guest Editors: Shao-wu Cheng, Rui Zhu, Zhicong Chen, and Hui Kong

Autonomous Transportation Systems Enabled by Emerging Technologies

Journal of Advanced Transportation

Autonomous Transportation Systems Enabled by Emerging Technologies

Lead Guest Editor: Linlin You
Guest Editors: Shao-wu Cheng, Rui Zhu, Zhicong Chen, and Hui Kong

Contents

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 6751908, 16 pages
https://doi.org/10.1155/2023/6751908

Research Article

Signal-Vehicle Coordination Control Modeling and Roadside Unit Deployment Evaluation under V2I Communication Environment

Wanting Chen,[1,2] **Zhaocheng He**,[1,2,3] **Yiting Zhu**,[1,2,3] **and Zhigang Wu**[1,2]

[1]*School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 518107, China*
[2]*Guangdong Provincial Key Laboratory of Intelligent Transportation System, Shenzhen 518107, China*
[3]*Pengcheng Laboratory, Shenzhen 518055, China*

Correspondence should be addressed to Yiting Zhu; zhuyt25@mail.sysu.edu.cn

Received 21 September 2022; Revised 12 September 2023; Accepted 25 September 2023; Published 17 October 2023

Academic Editor: Eneko Osaba

Signal-vehicle coordinated control holds substantial promise for enhancing urban transportation efficiency. However, its development faces notable challenges: (1) most existing studies have been conducted based on the assumption of perfect communication conditions. This assumption overlooks the significant impact of vehicle-to-infrastructure (V2I) communication quality on control performance, which leads to poor applicability in practice. (2) The evaluation of roadside unit (RSU) deployment for optimizing signal-vehicle control has not been well studied. Hence, the modeling of signal-vehicle coordination control and RSU deployment evaluation under V2I environment are studied in this paper. First, we introduce a communication model that characterizes the imperfections in communication between RSUs and connected vehicles (CVs). Second, we propose a model for signal-vehicle coordination control within this connected environment. This model integrates strategies from both signal control optimization and the speed optimization of CV platoons. Finally, to assess the impact of the RSU deployment parameters on the performance of signal-vehicle coordination control, we introduce a systematic evaluation method. The reduction in vehicle delays is introduced as the evaluation indicator for control performance. Six other indicators—the number of vehicles in the RSU communication domain, connectivity probability between the CV and RSU, number of vehicles whose speeds are successfully optimized, number of speed adjustments, green extension time, and overlap rate of the communication domains of multiple RSUs—are introduced as the observation indicators. The simulation experiments verify the effectiveness of the proposed model in implementing signal-vehicle coordination control under imperfect communication and environments in low-traffic, medium-traffic, and high-traffic scenarios. Furthermore, these experiments show the quantitative impact of RSU deployment parameters (communication distance, command transmission cycle, installation position, and number of RSUs) on control performance.

1. Introduction

Emerging vehicle-to-everything (V2X) technologies are a promising approach to addressing urban transportation network congestion. Owing to the limited computing capability of on-board units and the time-varying characteristics of vehicles [1], vehicle-to-vehicle (V2V) communication is insufficient for traffic control. To enhance the reliability of connected control, V2I communication technology has been incorporated, creating an extensive V2X communication domain through RSU deployment. RSUs receive real-time data from CVs and the signal controller and upload them to multiaccess edge computing (MEC) devices to generate control strategies. This facilitates signal-vehicle coordination control with a broader scope and more cooperative elements. Therefore, the scientific and efficient implementation of signal-vehicle coordination control has become a popular topic.

Current signal-vehicle coordination control strategies primarily concentrate on optimizing traffic indicators. Their goals include minimizing traffic pollution [2–4], enhancing traffic flow safety performance [5–7], and maximizing transportation efficiency [8–11]. These strategies compensate for the limited data resources and control range of

traditional signal control methods [12, 13]. In particular, the vehicle platooning is an effective way to improve traffic capacity and energy consumption [14, 15]. Platoons are employed as control objects in studies to develop control strategies. To minimize the total travel time of connected autonomous vehicles (CAVs), Ding et al. [16–18] introduced an optimal phase allocation and trajectory control method for CAV platoons, assuming instant interaction between all CAVs and the control center. Based on real-time and lossless communication conditions, Feng et al. [19, 20] proposed a signal-platoon coordinated control strategy to enhance intersection throughput. Chen et al. [21] developed a multiplatoon trajectory optimization approach to achieve safe, fuel-efficient, and efficient operations. In scenarios with low CV penetration rates, Li et al. [22–24] proposed a real-time predictive coordination method based on vehicle-triggered platoon dispersion to enhance control performance. However, these studies were conducted based on the assumption of perfect communication conditions between RSUs and CVs. The impacts of V2I communication conditions on signal-vehicle control performance have been ignored. For instance, factors such as the communication distance of RSUs and the connectivity probability between CVs and RSUs can affect the number of CVs that can participate in signal-vehicle control. Finkelberg et al. [25] experimentally demonstrated that realistic communication distortion conditions worsen the performance of traffic control. Neglecting the influence of imperfect communication conditions in these studies may limit their practical applicability.

Modeling an imperfect V2I communication environment is essential for signal-vehicle coordination control. In the V2I environment, RSUs play a pivotal role in gathering essential information from CVs and signal controllers, calculating control strategies, and delivering control commands [26]. Thus, further exploring the impacts of RSU deployment on signal-vehicle control performance is a meaningful topic.

To date, several studies have discussed the impact of RSU deployment on V2I communication quality, such as communication coverage [27], communication connectivity [28, 29], and communication delays [30, 31]. Liang et al. [32] presented an RSU deployment method to guarantee that V2I communication delays would remain below a threshold with the lowest cost. However, this method lacks applicability to emerging technologies that include 5G. Sankaranarayanan et al. [26] proposed a fusion algorithm-based optimal RSU distribution planner (ORDP). The ORDP is capable of constructing an objective function by selecting parameters such as RSU deployment cost, RSU transmission capacity, and communication coverage to meet the requirements of the scenario. However, it disregards the communication performance parameters including connectivity probability and transmission delay. Consequently, these studies fail to thoroughly analyze the effects of communication quality resulting from specific RSU deployments on control performance. Thus, these studies still cannot determine the impact mechanism of RSU deployment on signal-vehicle control performance.

Most RSU deployment studies have optimized deployment parameters for better perception of urban traffic flow. Barrachina et al. [33] formulated a density-based RSU deployment approach. Li et al. [34] established a model to investigate the relationship between RSU deployment, vehicle travel time, and traffic capacity. Olia et al. [35] proposed an RSU deployment optimization model to minimize RSU estimation errors for vehicle travel time. Salari et al. [36] developed a mathematical model to identify optimal RSU deployment for path flow reconstruction. However, few RSU deployment studies have explored the impact of deployment parameters on signal-vehicle control performance. Fang et al. [37] and Du et al. [38] noted that changes in RSU deployment can lead to communication disturbances affecting control performance, but a comprehensive theoretical and experimental analysis of these mechanisms is lacking. To address this gap, our study delves into the impact mechanism of RSU deployment parameters on signal-vehicle control performance.

In conclusion, the development of signal-vehicle coordination control encounters several challenges: (1) most existing studies assume ideal communication conditions between RSUs and CVs. The impacts of V2I communication conditions on signal-vehicle control performance are ignored in these studies, which limits practical applicability. (2) The evaluation of RSU deployment for signal-vehicle control optimization has not been well studied. Hence, this paper focuses on modeling signal-vehicle coordination control and evaluating RSU deployment under V2I environment.

The main contributions are summarized as follows:

(1) This article proposes a model of signal-vehicle coordination control under V2I communication environment. The model integrates strategies for both signal control optimization and speed optimization of the CV platoon, providing a comprehensive control strategy for multiple vehicles. Furthermore, it incorporates communication conditions such as connectivity probability and command transmission cycle to characterize the imperfect communication environment between RSUs and CVs. This approach challenges the prevailing assumption of perfect communication in most related studies.

(2) This article presents a method of evaluating the impacts of RSU deployment parameters on the performance of signal-vehicle coordination control. Comprehensive experiments quantitatively reveal the impact mechanism of the deployment parameters (communication distance, command transmission cycle, installation position, and number of RSUs) on the control performance.

2. Methodology

2.1. Overview. As shown in Figure 1, the research framework includes three modules: the communication model of RSU and CV platoons, the signal-vehicle coordination control strategy under a connected environment, and the method to

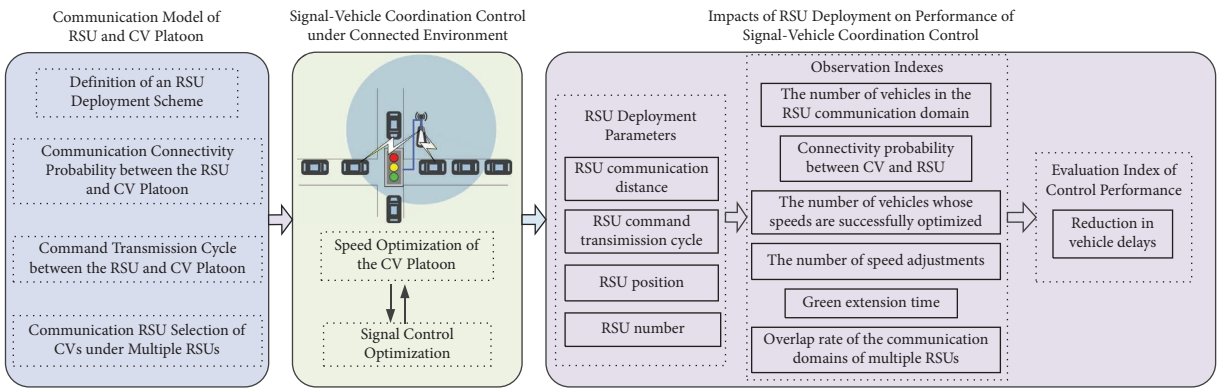


Figure 1: Overview.

evaluate the impacts of RSU deployment parameters on the performance of signal-vehicle coordination control. The function of the communication model is to develop an RSU deployment scheme, realize communication connection between the RSU and CV, and determine the command transmission cycle and the communication RSU selection of CVs under multiple RSUs. Based on the communication interactions of RSUs and CVs, a signal-vehicle coordination control strategy is formulated. The strategy includes the speed optimization of the CV platoon and the signal control optimization. In the third module, a method of evaluating the impacts of RSU deployment parameters on the performance of signal-vehicle coordination control is designed. The reduction in vehicle delays is used as the evaluation index of the control performance. Six other indicators—the number of vehicles in the RSU communication domain, connectivity probability between the CV and RSU, number of vehicles whose speeds are successfully optimized, number of speed adjustments, green extension time, and overlap rate of the communication domains of multiple RSUs—are employed as observation indicators. Simulation experiments under various RSU deployment parameters are conducted to demonstrate the impact mechanism of RSU deployment on the control performance.

The study scenario at an isolated signal intersection under V2I environment is shown in Figure 2.

Three assumptions are made in this paper.

(1) The communication between the RSU, the MEC server, and the signal controller is considered to be without transmission loss and delay. In addition, the signal controller executes the optimization strategy received from an RSU.

(2) All the vehicles are CVs, and all the vehicle members of a CV platoon are assumed to be within the communication range of the leading vehicle.

(3) Considering the CV platoon as a control unit, the complex behaviors between CV vehicles, such as decomposition and reorganization in the CV platoon, are not described. The RSU transmits information to the platoon's leading CV, and the following CVs always follow the commands from their preceding cars.

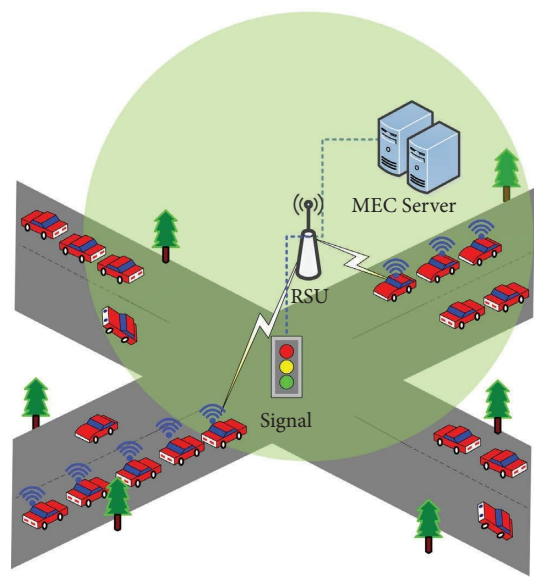


Figure 2: The study scenario at an isolated signal intersection under V2I environment.

2.2. Communication Model of the RSU and CV Platoon

2.2.1. Definition of an RSU Deployment Scheme. An RSU deployment scheme is defined as a quadruple $U = (D_{rsu}, T_{com}, X_{rsu}, N_{rsu})$, where D_{rsu} is the RSU communication distance, T_{com} is the RSU command transmission cycle, X_{rsu} is the RSU deployment position, and N_{rsu} is the RSU number.

The communication domain of an RSU is determined by its position and communication distance. Within the communication domain, an RSU can exchange information with CVs with a certain connectivity probability. The MEC server connected to the RSU subsequently generates a control strategy in response to the traffic flow and transmits it to the CVs in a regular cycle. Then, the communication topology between the RSU and the CV is updated.

2.2.2. Communication Connectivity Probability between the RSU and CV Platoon. The number of CVs and the number of RSUs are p and q, respectively. To establish a plane

rectangular coordinate system, this paper sets the center of the intersection as the origin, east as the positive direction of the x-axis, and north as the positive direction of the y-axis. Let X^j_{rsu} represent the position of the RSU. D^j_{rsu} denotes the communication distance of RSU j, and X^i_{veh} represents the position of CV i. CVs interact with RSUs via shortwave communication.

$$d^{\text{ij}}_{\text{vr}} = \sqrt{X^{i^2}_{\text{veh}} + X^{j^2}_{\text{rsu}}}\,(i = 1, 2, \ldots p;\ j = 1, 2, \ldots q). \tag{1}$$

In equation (1), $d^{\text{ij}}_{\text{vr}}$ represents the distance between RSU j and CV i. If $d^{\text{ij}}_{\text{vr}} \le D^j_{\text{rsu}}$, RSU j has the opportunity to establish a communication connection with CV i.

During information transmission, the information will undergo signal attenuation due to factors such as obstacle occlusion. An RSU and a CV are directly connected only when the received power P_r exceeds a predefined threshold P_{th}. The received power P_r and the communication probability $P(d^{\text{ij}}_{\text{vr}})$ between RSU j and CV i is given by the following equations [39]:

$$P_r = P_t - PL(d_0) - 10\alpha \lg\left(\frac{d^{\text{ij}}_{\text{vr}}}{d_0}\right) + Z_\sigma, \tag{2}$$

$$P\left(d^{\text{ij}}_{\text{vr}}\right) = P\{P_r \ge P_{\text{th}}\} = \frac{1}{2} + \frac{1}{2}\text{er}\,f\left(\frac{10\alpha \lg\left(r/d^{\text{ij}}_{\text{vr}}\right)}{\sqrt{2}\,\sigma}\right), \tag{3}$$

$$r = d_0 \cdot 10^{\left(P_t - \text{PL}\left(d_0\right) - P_{\text{th}}\right)/10\alpha}, \tag{4}$$

where P_t is the transmitted power, $\text{PL}(d_0)$ is the reference path loss at a critical distance d_0, α is the path loss exponent, and Z_σ is a Gaussian random variable with zero mean and standard deviation σ.

2.2.3. Command Transmission Cycle between the RSU and CV Platoon. Based on successful communication behaviors, the signal phase and timing (SPAT) message generated in the RSU, together with the real-time CVs' basic safety message (RSM), are sent back to the MEC, where the traffic control strategies are calculated. The strategies are sent to the CVs via the downlink according to the transmission cycle T_{com}. As the guidance command is distributed periodically, the successful transmission probability of the guidance information at time t is described by the following equation:

$$P^{\text{ij}}_c = \begin{cases} P\left(d^{\text{ij}}_{\text{vr}}\right), & \text{if } t \mid T_{\text{com}}, \\ 0, & \text{if } t \nmid T_{\text{com}}. \end{cases} \tag{5}$$

The RSU regularly sends guidance commands to CVs. In such a case, the probability that a CV receives command and updates its motion at any time t is P^{ij}_c.

2.2.4. Communication RSU Selection of CVs under Multiple RSUs. Due to the constraints of channel resources, a CV interacts exclusively with one RSU upon entering the overlapping region of multiple RSU communication domains. To guarantee communication quality of communication, the CV selects its communication RSU based on the principle of maximizing connectivity probability. The communication connectivity probability between RSU k and CV i is given by the following equation:

$$P\left(d^{\text{ik}}_{\text{vr}}\right) = \max\left\{P\left(d^{i1}_{\text{vr}}\right), P\left(d^{i2}_{\text{vr}}\right), P\left(d^{i3}_{\text{vr}}\right), \ldots, P\left(d^{\text{iq}}_{\text{vr}}\right)\right\}. \tag{6}$$

2.3. Signal-Vehicle Coordination Control under Connected Environment. Figure 3 illustrates the workflow of signal-vehicle coordination control under connected environment. The objective of signal-vehicle control is to minimize the total travel time of CVs.

The first step is to initialize the traffic state and the signal timing scheme. Subsequently, the RSU communicates with the CV platoon within its communication domain and estimates the time required by the CV platoon to pass the intersection. Combined with the SPAT, the RSU determines whether to transmit a suggested speed to the platoon. If the suggested speed is transmitted, the CV platoon will implement the speed adjustment command. Otherwise, the platoon will stop at the stop line to wait for the next green time.

2.3.1. Speed Optimization of the CV Platoon. The RSU offers a suggested speed to the target CV platoon in the communication, with the aim of minimizing the total travel time of the CV platoon and enabling the platoon to cross the intersection within the green time (including the green extension time).

When an RSU communicates with the CV platoon, the remaining passing time for the platoon is T_l. To calculate the suggested speed, two scenarios are considered in this paper. The first scenario is that no vehicle is ahead of the target CV platoon. The objective function can be formulated as in the following equation:

$$f_1 = \min\left[\max\left(0, T^i_a\right)\right], \tag{7}$$

$$T^i_a = \rho\left[T^i_{\text{constant}} + T^i_{\text{acc}(1)}\right] + (1-\rho)T^i_{\text{acc}(2)}, \tag{8}$$

$$\rho = \begin{cases} 1, & L^i \ge \left|\dfrac{v^2 - v^{i^2}_p}{2a^i_p}\right|, \\ 0, & L^i \le \left|\dfrac{v^2 - v^{i^2}_p}{2a^i_p}\right|, \end{cases} \tag{9}$$

$$L^i = L^i_1 + m^i L_{\text{car}} + \left(m^i - 1\right)L_{\text{in}}, \tag{10}$$

$$T^i_{\text{constant}} = \frac{L^i - \left|\left(v^2 - v^{i^2}_p/2a^i_p\right)\right|}{v}, \tag{11}$$

$$T^i_{\text{acc}(1)} = \left|\frac{v - v^i_p}{a^i_p}\right|, \tag{12}$$

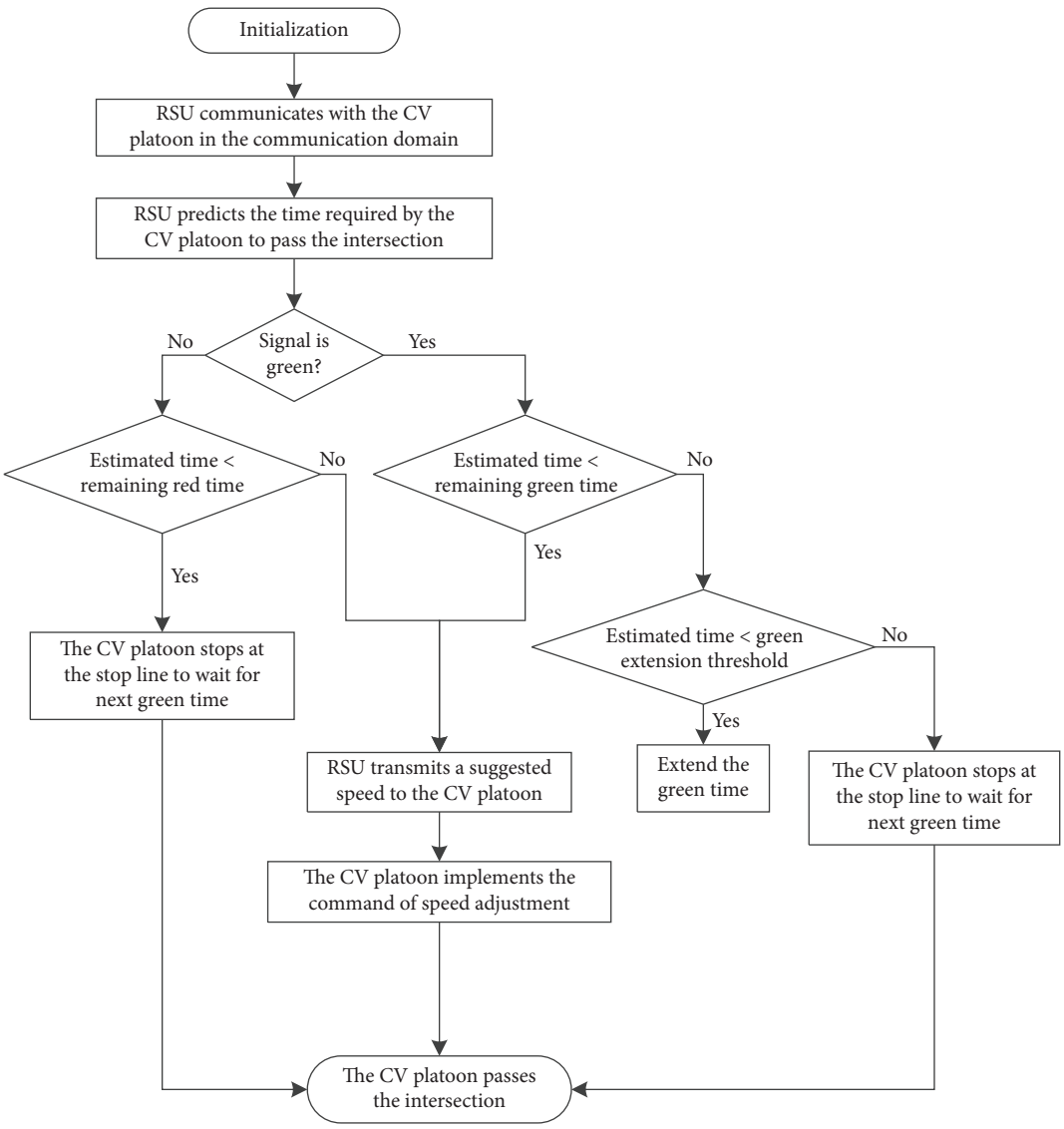


Figure 3: Workflow of signal-vehicle coordination control under connected environment.

$$T^i_{\mathrm{acc}(2)} = \frac{-v^i_p + \sqrt{v^{i^2}_p + 2a^i_p L^i}}{a^i_p}. \tag{13}$$

Let T^i_a denote the travel time that the platoon i takes to pass the intersection from its current position. In equations (8) and (9), the parameter ρ indicates whether the platoon i is capable of crossing the intersection by accelerating from the current speed v^i_p to the suggested speed v. $T^i_{\mathrm{acc}(1)}$ represents the time that platoon i takes to accelerate to the suggested speed. T^i_{constant} represents the time that platoon i takes to pass the intersection in a uniform velocity after the acceleration to the suggested speed. $T^i_{\mathrm{acc}(2)}$ represents the time that platoon i takes to pass the intersection at the suggested speed. L^i represents the distance from the last vehicle in the platoon i to the stop line. In equation (10), L^i_1 denotes the distance from the leading vehicle in the platoon i to the stop line. L_{car} denotes the vehicle length. L_{in} is the distance between CVs within a platoon. m^i and a^i_p are the vehicle number and the acceleration of the platoon i, respectively.

In the second scenario, other CVs travel in front of the platoon i. To mitigate the interference of the vehicles ahead of platoon i, in addition to the target platoon, the RSU controls the CVs ahead of platoon i. If all vehicles ahead of the CV platoon i are in the RSU communication domain, the RSU incorporates the front vehicles into the target platoon i. The size of the platoon is expanded, and the first vehicle of the updated platoon becomes the leading vehicle. The RSU transmits the suggested speed to the leading vehicle of the updated platoon. The guidance strategy is calculated by the equations (7)–(13).

According to the speed v_l of the preceding platoon, RSU transmits the following speed v_g to the leading vehicle of the platoon i. The following model is utilized to calculate the speed v_g [40].

$$v_g = \min\left\{v_g + \frac{2a_{\max}(d_s + d_{\mathrm{lf}})}{(v_l + v_{\mathrm{free}})}, \frac{2a_{\max}L_1^i}{v_{\mathrm{free}}}\right\}. \tag{14}$$

In equation (14), v_{free} denotes the free-flow speed. $a_{\max}$ represents the maximum acceleration of the vehicle. d_s and d_{lf} are namely the safe distance and the distance between the follower and the leader.

Since a CV outside the RSU communication domain fails to receive guidance information, the motion of the target platoon i will be influenced. Therefore, the RSU steers all CVs within the communication range as a platoon, which follows the last CV inside the RSU communication domain. The speed data of the CVs outside the RSU communication domain is transmitted by a holographic detection device to the RSU. The RSU calculates the suggested speed according to equation (14). The travel time T_b of platoon i is given by the following equation:

$$T_b = \begin{cases} T_{\mathrm{lr}} + T_q + \dfrac{Q+M}{T_s}, & v_l = 0, \\ \dfrac{Q+M}{T_s}, & v_l \neq 0, \end{cases} \tag{15}$$

where Q and M denote the number of CVs outside and inside the RSU communication domain, respectively. T_{lr} represents the remaining red time, T_s represents the saturation flow rate, and T_q represents the vehicle start-up loss time.

The mentioned objective function is supposed to meet two constraints: (i) the suggested speed is within the range composed by the minimum speed $v_{\min}$ and the maximum speed $v_{\max}$ and (ii) the suggested speed is sufficient for the target platoon to pass the intersection within the remaining passing time. Hence, the constraints can be denoted as equation (16). $T \in \{T_a, T_b\}$ represents the time that the platoon takes to pass the intersection.

$$\text{s.t.} \begin{cases} v_{\min} \leq v \leq v_{\max}, \\ T < T_l. \end{cases} \tag{16}$$

Based on the estimated time of the platoon and the SPAT, the RSU evaluates the platoon's current motion status. When the platoon is capable of passing the intersection within the remaining travel time, the RSU transmits a suggested speed to the target CV platoon.

2.3.2. Signal Control Optimization. If the time required by the platoon to pass the intersection is greater than the current remaining green time T_{lg}, the RSU evaluates whether to execute a green extension strategy according to the following equation:

$$y = T - T_{\mathrm{lg}} - G_{\max}, \tag{17}$$

$$G_{\max} = T_{\mathrm{og}} - \frac{Q_c}{T_s}, \tag{18}$$

where $G_{\max}$ is the maximum green extension threshold. Q_c and T_{og} are the average vehicle queue length and green time of uncontrolled phases, respectively. If $y > 0$, the green extension time required by the platoon to pass the intersection exceeds the threshold. Thus, the platoon drives based on the car-following model and the signal timing scheme remains unchanged. If $y \leq 0$, the green extension time required by the platoon to pass the intersection lies within the threshold. In such case, the RSU transmits a command to the signal controller for the green extension. The green extension time T_{len} is presented as follows:

$$T_{\mathrm{len}} = T - T_{\mathrm{lg}}. \tag{19}$$

2.4. Impacts of RSU Deployment on the Performance of Signal-Vehicle Coordination Control. The RSU deployment scheme influences the implementation performance of the signal-vehicle coordination control strategy via communication interaction with CVs and signal controllers. Traffic delay is an essential index for evaluating the service level of an intersection [41]. Consequently, the reduction in vehicle delays is employed to evaluate the signal-vehicle coordination control performance, and six other indicators are selected as observation indexes to construct the evaluation index system for the RSU deployment scheme (see Table 1).

CVs are categorized into three types in the proposed scenario. (1) CVs geographically located in the RSU communication domain, whose average number per second is defined as C; (2) CVs in the RSU communication domain that may pass through the intersection without stopping via RSU control; and (3) CVs in the RSU communication domain that successfully execute the speed adjustment command and pass through the intersection without stopping during the corresponding signal cycle (including the green extension time), whose number is defined as K.

The number H of speed adjustments and the number K of vehicles whose speeds are successfully optimized are two indicators that quantify the changes in vehicle operation under the coordinated control. The green extension time G_{ext} is an indicator that quantifies the changes in signal parameters under coordinated control. Moreover, multiple RSUs may have an overlapping area of communication. The lower the overlap rate Z is, the higher the number C of vehicles in the communication domain is.

3. Simulation and Analysis

3.1. Scenario Description. We use simulation of urban mobility (SUMO) to build a four-approach signalized intersection, and the length of each approach is 1 km. The simulation time is 3600 s. The experimental parameter

Table 1: Relative indicators for impact analysis of RSU deployment on control performance.

Indicators	Types	Definitions
Reduction in vehicle delays R	Evaluation index of control performance	The difference between the total delays with and without the signal-vehicle coordinated control under CV-RSU communication
The number of vehicles in the RSU communication domain C	Observation index of CV-RSU communication	The average number of vehicles in the RSU communication domain per second
Connectivity probability between CV and RSU P_c	Observation index of CV-RSU communication	The average connectivity probability between CV and RSU
The number of vehicles whose speeds are successfully optimized K	Observation index of signal-vehicle coordinated control	The number of CVs whose speeds are successfully optimized and pass through the intersection without stopping during the signal cycle (including the green extension time)
The number of speed adjustments H	Observation index of signal-vehicle coordinated control	The number that CVs adjust their speed toward the suggested speed received from RSU
Green extension time G_{ext}	Observation index of signal-vehicle coordinated control	The total green extension time that the signal controller executes
Overlap rate of the communication domains of multiple RSUs Z	Observation index of CV-RSU communication	The ratio of the overlapping communication area of multiple RSUs to their total communication area

settings are shown in Table 2. For the threshold of received power and average distance between CVs within a platoon given in Table 2, refer to [42, 43], respectively.

We consider three different traffic flow modes: low, medium, and high traffic flows, represented by traffic flows of $400\,\text{veh}\cdot h^{-1}\cdot\text{lane}^{-1}$, $800\,\text{veh}\cdot h^{-1}\cdot\text{lane}^{-1}$, and $1200\,\text{veh}\cdot h^{-1}\cdot\text{lane}^{-1}$, respectively. Under these three modes, the intersection saturation values are approximately 0.3, 0.6, and 0.9, respectively. The ratio of north-south traffic flow to east-west traffic flow is 4 : 3, and the north-south traffic flow is the major traffic flow. In this paper, we focus on the north-south traffic flow as the RSU control target. In the single-RSU deployment scenario, the RSU is positioned at the south approach of the intersection. In the multiple-RSU deployment scenarios, RSUs are installed at the south and north approaches of the intersection. The parameters of the RSU deployment schemes are presented in Table 3.

In order to investigate the impacts of RSU deployment parameters on performance of signal-vehicle coordination control, RSU deployment scheme for different analysis is shown in Table 4.

As shown in Figure 4, the RSU is designed on SUMO, and the blue area is the communication range of the RSU.

Table 2: Experimental parameters.

Parameters	Values
Threshold of received power (dB)	50
Low traffic flow ($\text{veh}\cdot\text{h}^{-1}\cdot\text{lane}^{-1}$)	400
Medium traffic flow ($\text{veh}\cdot\text{h}^{-1}\cdot\text{lane}^{-1}$)	800
High traffic flow ($\text{veh}\cdot\text{h}^{-1}\cdot\text{lane}^{-1}$)	1200
Average velocity ($\text{km}\cdot\text{h}^{-1}$)	40
Maximum velocity ($\text{km}\cdot\text{h}^{-1}$)	60
Maximum acceleration ($\text{m}\cdot\text{s}^{-2}$)	1.5
Minimum acceleration ($\text{m}\cdot\text{s}^{-2}$)	−4
Platoon size (veh)	10
Average distance between CVs within a platoon (m)	5
Minimum distance between CVs within a platoon (m)	1.5
Vehicle length (m)	4
Phase number	2
Signal cycle (s)	90
Green time (s)	42
Yellow time (s)	3

3.2. Impact Analysis of a Single-RSU Deployment

3.2.1. Impacts of RSU Communication Distance on Control Performance. As illustrated in Table 4, this section utilizes the RSU deployment schemes of U1–U5 for low, medium, and high traffic flows, where the RSU number, position, and command transmission cycle are unchanged. A single RSU with a communication distance of $D_{\text{rsu}} = \{100,200, 300,400,500\}$ is deployed at the roadside location of stop line on the south approach.

Figure 5(d) presents the impacts of RSU communication distance D_{rsu} on the reduction in vehicle delays R in the north-south direction. Under different traffic flows, R tends to increase and then decrease, as D_{rsu} increases. From Figure 5(a), it is clear that as D_{rsu} increases, the control range of the RSU expands, and C increases. Figure 5(b) shows that, given the characteristics of wireless transmission, P_c drops as D_{rsu} increases. This reduces the number of CVs that can successfully receive RSU commands. As shown in Figure 6(c), combining the above two effects, K tends to increase and then decrease as D_{rsu} increases, and R tends to increase and then decrease.

The peak of vehicle delay reduction is observed at an RSU communication distance of 200 m, because the number of vehicles in the RSU communication domain C has the largest growth, and the connectivity probability between a CV and RSU P_c remains at a high level. C increases slowly and P_c decreases dramatically when D_{rsu} is larger than 200 m. The analysis results for the two indicators show that the best choice of RSU communication distance is 200 m.

Overall, the RSU communication distance D_{rsu} affects both the number of vehicles in the RSU communication domain C and connectivity probability between CV and RSU P_c. The combination of two indexes influences the number of vehicles whose speeds are successfully optimized K, which ultimately changes the reduction in vehicle delays R.

3.2.2. Impacts of the RSU Command Transmission Cycle on the Control Performance. As shown in Table 4, this section utilizes RSU deployment schemes of U2, U6–U9 for low, medium, and high traffic flows, where RSU number, position, and communication distance are unchanged. Considering that the travel distance of CV within the cycle needs to be less than the minimum communication distance of RSU, the command transmission cycle is selected as $T_{\text{com}} = \{1, 3, 5, 7, 9\}$.

From Figure 6(c), under different traffic flows, the reduction in vehicle delays R for the north-south traffic flow follows a decreasing trend as the RSU command transmission cycle T_{com} increases. Further insight into the impacts of the RSU command transmission cycle is provided below. As shown in Figures 6(a) and 6(b), as T_{com} increases, H and G_{ext} decrease. Therefore, R decreases. As T_{com} increases, the number of issued control strategies decreases rapidly. This influences the optimization of the signal-vehicle control performance.

To summarize, the RSU command transmission cycle T_{com} affects both the number of speed adjustments H and green extension time G_{ext}, which eventually influences the reduction in vehicle delays. As T_{com} expands, H and G_{ext} both display a downward trend, which leads to a decrease in R.

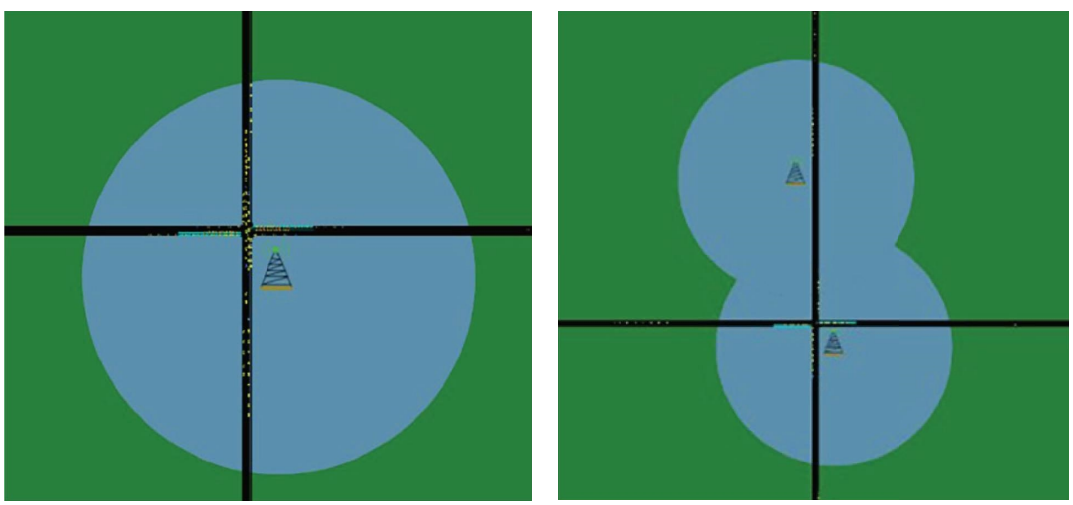
3.2.3. Impacts of the RSU Position on the Control Performance. As illustrated in Table 4, this section utilizes RSU deployment schemes of U6, U10–U12 for low, medium, and high traffic flows, where RSU number, communication distance, and command transmission cycle are unchanged. A single RSU under different positions is deployed to explore the impacts of the RSU deployment position on the control performance.

Table 3: Parameters of RSU deployment schemes.

Schemes	D_{rsu} (m)	T_{com} (s)	N_{rsu}	X_{rsu} (m)
U1	100	1	1	0 m from the stop line of the south approach
U2	200	1	1	0 m from the stop line of the south approach
U3	300	1	1	0 m from the stop line of the south approach
U4	400	1	1	0 m from the stop line of the south approach
U5	500	1	1	0 m from the stop line of the south approach
U6	200	3	1	0 m from the stop line of the south approach
U7	200	5	1	0 m from the stop line of the south approach
U8	200	7	1	0 m from the stop line of the south approach
U9	200	9	1	0 m from the stop line of the south approach
U10	200	1	1	100 m from the stop line of the south approach
U11	200	1	1	200 m from the stop line of the south approach
U12	200	1	1	300 m from the stop line of the south approach
U13	200	1	2	RSU 1 is 0 m from the stop line of the south approach, RSU 2 is 100 m from the stop line of the north approach
U14	200	1	2	RSU 1 is 0 m from the stop line of the south approach, RSU 2 is 200 m from the stop line of the north approach
U15	200	1	2	RSU 1 is 0 m from the stop line of the south approach, RSU 2 is 300 m from the stop line of the north approach
U16	200	1	2	RSU 1 is 0 m from the stop line of the south approach, RSU 2 is 400 m from the stop line of the north approach

Table 4: RSU deployment scheme for different analysis.

Analysis	Scheme
Impacts of RSU communication distance on the control performance	U1–U5
Impacts of RSU command transmission cycle on the control performance	U2, U6–U9
Impacts of single-RSU position on the control performance	U2, U10–U12
Impacts of dual-RSU position on the control performance	U13–U16
Impacts of RSU number on the control performance	U2, U12, and U16



(a) (b)

Figure 4: Simulation of signal-vehicle coordination control under CV-RSU communication. (a) The scene with a single-RSU located at the south approach, where the deployment position, communication distance, and command transmission cycle of the RSU are variable. (b) The scene with two RSUs located at the south approach and north approach, respectively, where the deployment positions of two RSUs are variable.

From Figure 7(c), under low traffic flow, the reduction in vehicle delays R for the north-south traffic flow decreases gradually as the RSU position X_{rsu} increases. Under medium and high traffic flows, the reduction in vehicle delays R for the north-south traffic flow increases as the RSU position X_{rsu} increases.

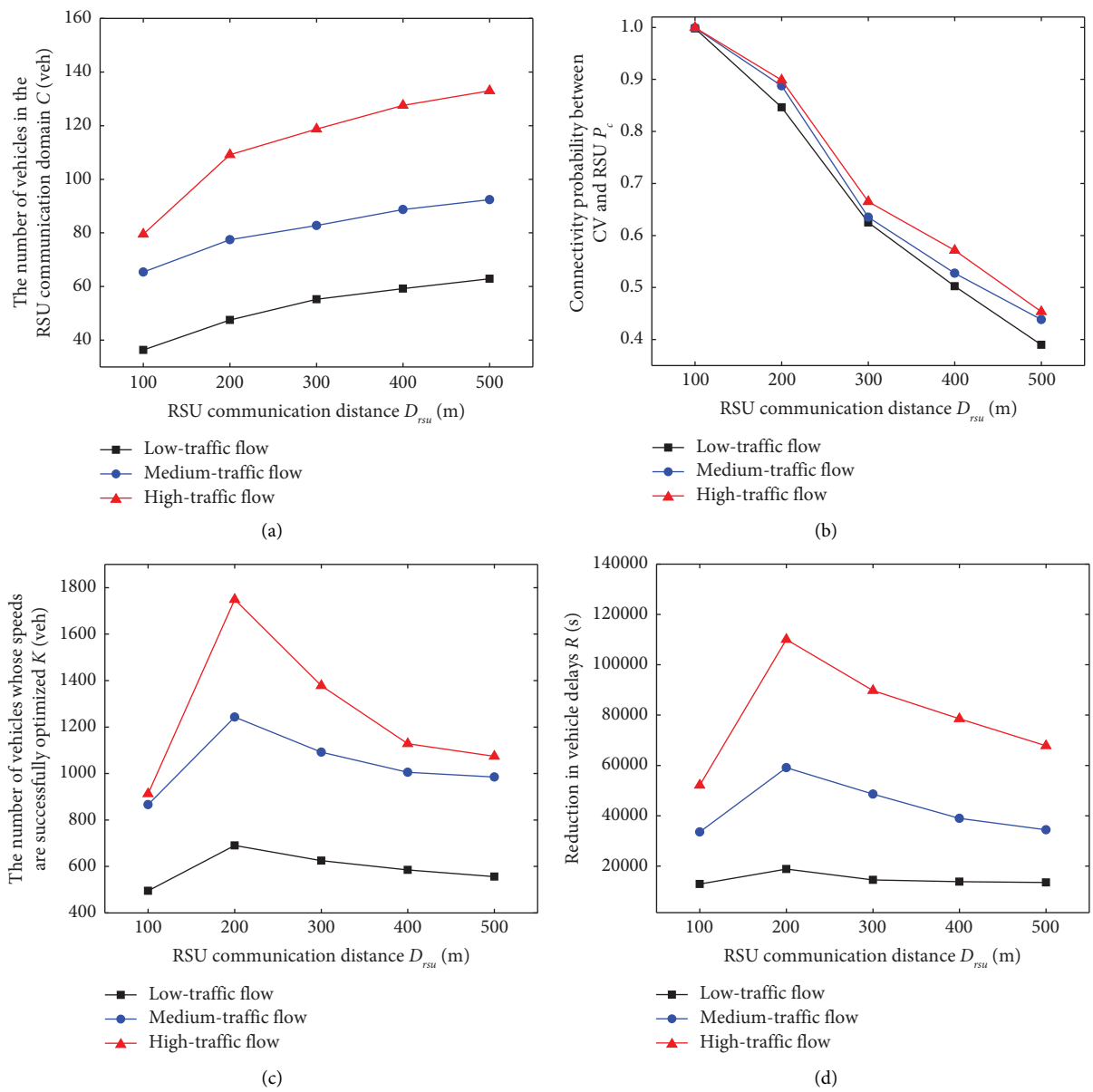


Figure 5: Impacts of RSU communication distance D_{rsu} on the analysis indicators.

Since the number of vehicles at the intersection is the largest, C tends to decrease as X_{rsu} increases in Figure 7(a). As shown in Figure 7(b), under low traffic flow, the vehicles are free to travel, the control space is large, and P_c remains unchanged. When the RSU is located closer to the intersection, C increases. The coordination control considers a broader range of traffic flows, which leads to an increase in K and R. Therefore, installing a single RSU close to the intersection is effective for reducing vehicle delays in low traffic flows.

Under medium and high traffic flows, queuing vehicles pass through the intersection with a saturation flow rate when the signal turns green. These vehicles lack guided space and value. As X_{rsu} increases, the proportion of queuing vehicles in the RSU communication domain is reduced. The number of vehicles with guided space grows. Consequently, K increases and eventually R displays an upward trend. Further analysis finds that considering only a static traffic flow is not sufficient for RSU deployment; the behavior and demands of dynamic traffic flow need to be addressed as well.

The position X_{rus} of an RSU influences the number of vehicles in the RSU communication domain C. This will influence the number of vehicles whose speeds are successfully optimized K. Consequently, the reduction in vehicle delays R is affected.

3.3. Impact Analysis of Dual-RSU Deployment

3.3.1. Impacts of the RSU Position on the Control Performance. As illustrated in Table 4, this section adopts the RSU deployment schemes of U13–U16 for low, medium,

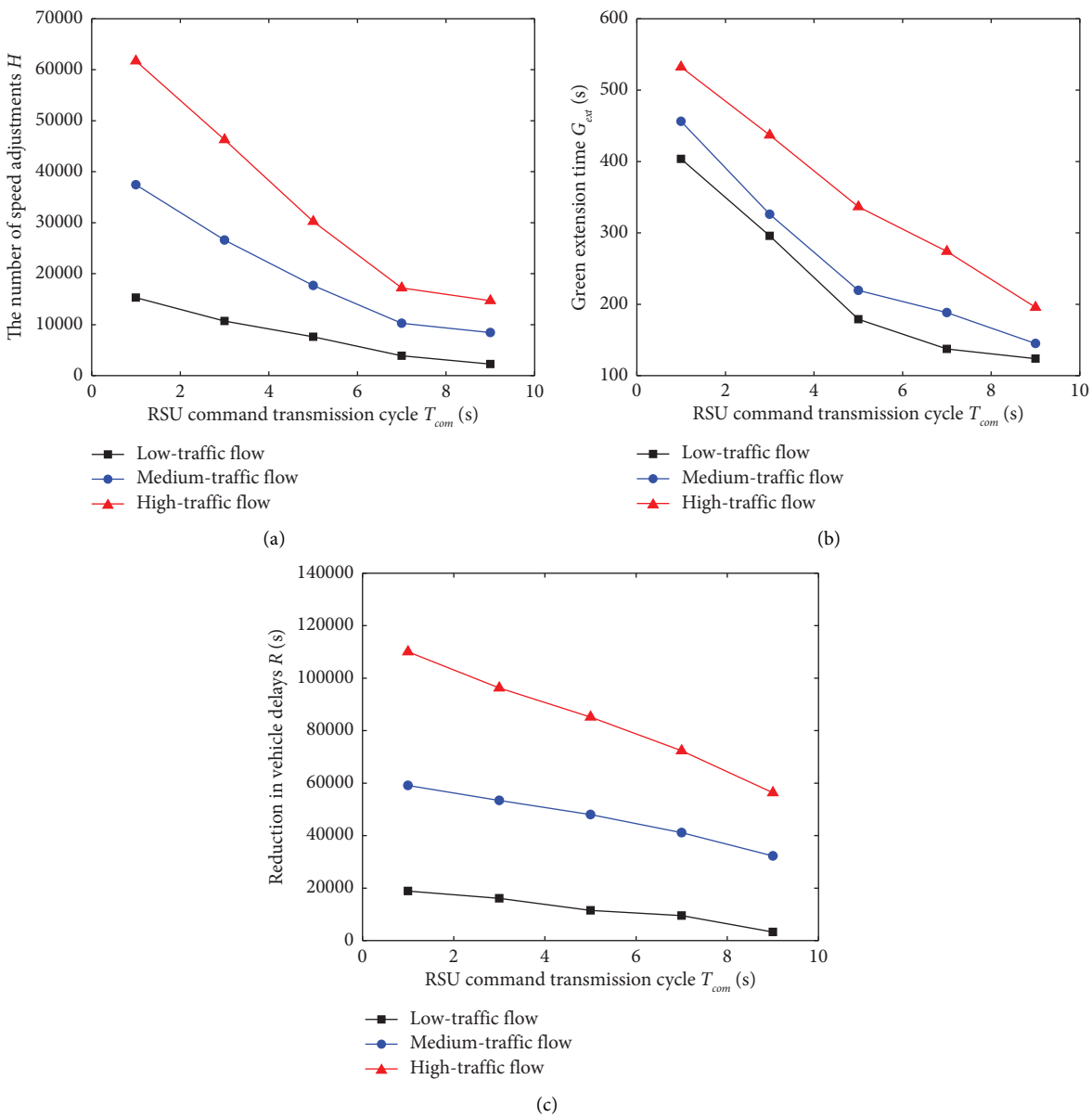


Figure 6: Impacts of the RSU command transmission cycle T_{com} on the analysis indicators.

and high traffic flows, where the RSU number, communication distance, and command transmission cycle are unchanged. Double RSUs are deployed at different positions on the north approach and south approach.

As shown in Figure 8(d), the reduction in vehicle delays R increases as the dual-RSU position X_{rsu} increases.

Under different traffic flows, as indicated in Figure 8(a), increasing the distance between the RSU located at the north approach and the intersection can reduce Z and expand the effective coverage of double RSUs. As a result, in Figure 8(b), C increases as X_{rsu} increases. Since the RSU-CV platoon connectivity probability P_c is unchanged, as presented in Figure 8(c), K increases and R increases accordingly.

The dual-RSU position X_{rsu} affects the overlap rate of the communication domains of multiple RSUs Z. This will influence the number of vehicles in the communication domain C, which in turn influences the number of vehicles whose speeds are successfully optimized K and ultimately changes reduction in vehicle delays R.

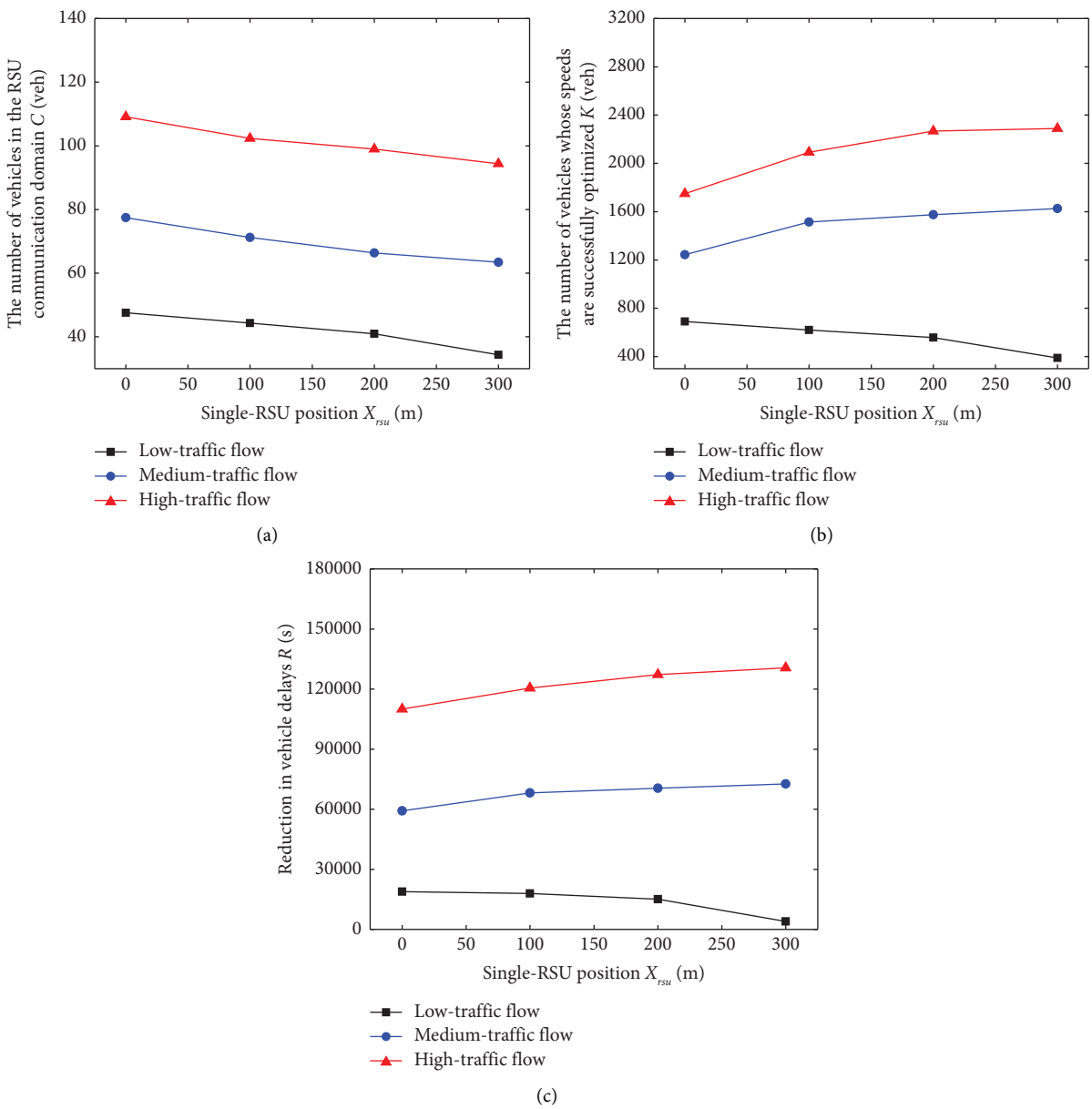


Figure 7: Impacts of the single-RSU position X_{rsu} on the analysis indicators.

3.3.2. Impacts of the RSU Number on the Control Performance. As illustrated in Table 5, to determine the impacts of the RSU number on the control performance, we adopt RSU deployment schemes U2 and U16 for low traffic flow and U12 and U16 for medium and high traffic flows. In Table 5, compared with the single-RSU deployment scheme, the deployment of two RSUs yields better performance in reducing vehicle delays. The experiments confirm that increasing the number of RSUs is crucial for enhancing the signal-vehicle coordination control performance.

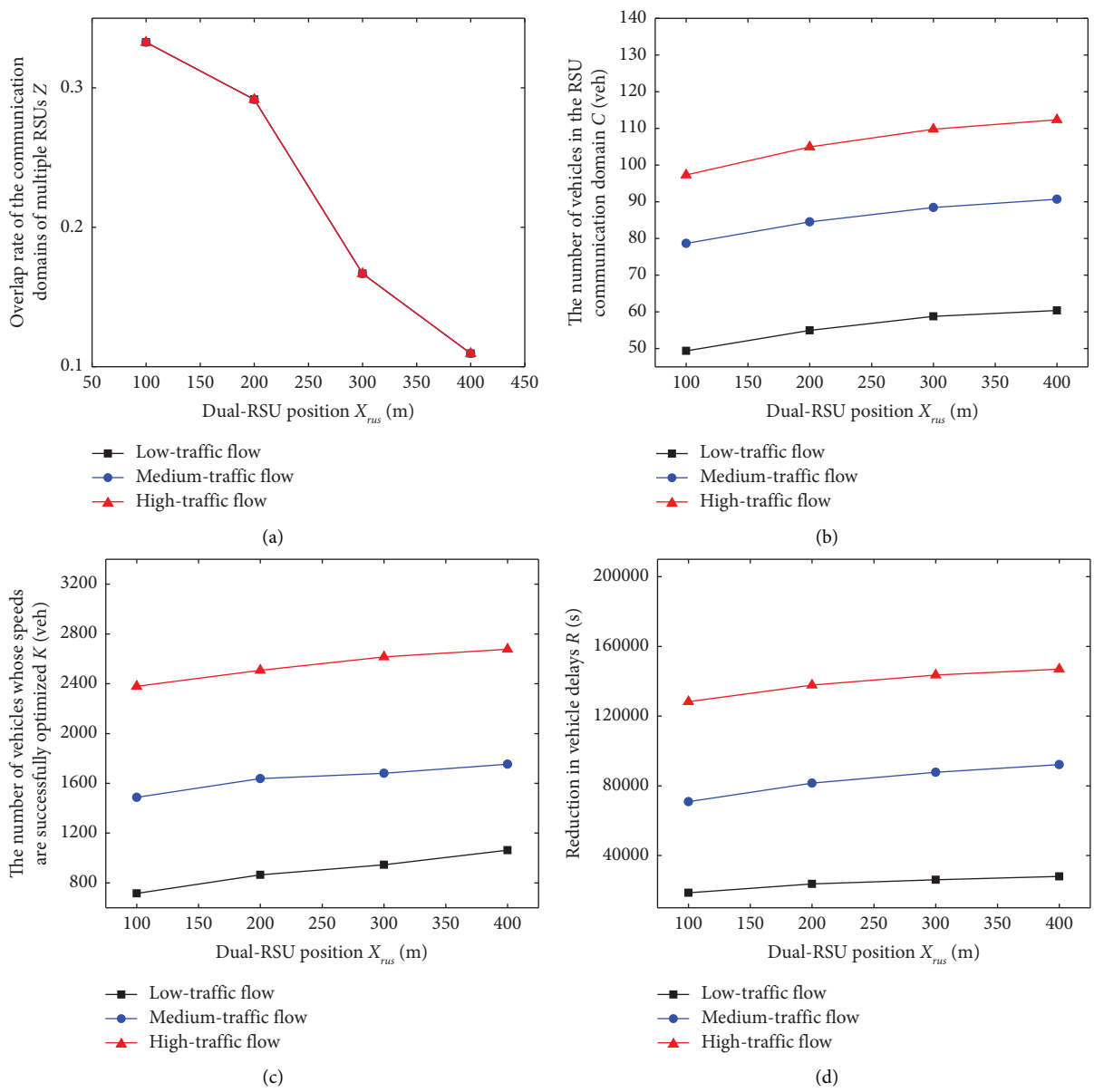


Figure 8: Impacts of the dual-RSU position X_{rsu} on the analysis indicators.

Table 5: Reduction in vehicle delays under different numbers of RSU.

Number of RSU	R under low traffic flow(s)	R under medium traffic flow(s)	R under high traffic flow(s)
1	18829.21	72609.29	130619.15
2	28012.40	92139.53	147001.35

Table 6: The influence of different RSU deployment parameters on the control performance.

Parameters	Observation indexes	Impact on the control performance
D_{rsu}	C, P_c, and K	The reduction in vehicle delays displays an upward trend and then a downward trend as the RSU communication distance increases
T_{com}	H and G_{ext}	The reduction in vehicle delays decreases as the RSU command transmission cycle increases

Table 6: Continued.

Parameters	Observation indexes	Impact on the control performance
X_{rsu} of a single RSU	C and K	Under low traffic flow, the reduction of vehicle delays decreases as the RSU position moves away from the intersection. Under medium traffic and high traffic flows, the reduction in vehicle delays increases as the RSU position moves away from the intersection
X_{rsu} of dual-RSU	Z, C, and K	The reduction in vehicle delays increases as the RSU position moves away from the intersection

3.3.3. Experiment Summary. In conclusion, Table 6 shows the influence of different RSU deployment parameters on the control performance. It reveals the impact mechanism of RSU deployment on the control performance. This provides theoretical support for the modeling and RSU deployment optimization algorithm under signal-vehicle coordination control.

4. Conclusions

In this paper, we investigate the modeling of signal-vehicle coordination control and RSU deployment evaluation under V2I environment. First, we develop a communication model of RSU and CV platoons under imperfect communication conditions. Second, a signal-vehicle coordination control strategy is designed to minimize the travel time of the CV platoon. Ultimately, we propose a method of evaluating the impacts of RSU deployment parameters on the performance of signal-vehicle coordination control. Through simulations, we reveal the impact mechanism of the RSU deployment parameters on the control performance.

The major conclusions are summarized as follows: (1) The reduction in vehicle delays tends to increase and then decrease as the RSU communication distance increases. (2) The reduction in vehicle delays decreases as the RSU command transmission cycle increases. (3) Under low traffic flow, the reduction in vehicle delays decreases as the distance between the intersection and RSU position increases. Under medium and high traffic flows, the reduction in vehicle delays increases as the distance between the intersection and RSU position increases. (4) The reduction in vehicle delays increases as the position of dual RSUs moves away from the intersection.

This article presents initial experimental research on the number of RSU deployments. Future research will integrate the installation cost and the number of RSUs to develop an RSU deployment optimization method. Moreover, future research aims to incorporate resource scheduling, communication delay, and other communication models to obtain a more realistic cosimulation model for both transportation and communication networks.

Data Availability

The data used in this study can be obtained from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (52202406 and U21B2090).

References

[1] H. Zhao, Y. Li, W. Hao, S. Peeta, and Y. Wang, "Evaluating the effects of switching period of communication topologies and delays on electric connected vehicles stream with car-following theory," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 12, pp. 7631–7641, 2021.

[2] H. Chen, H. A. Rakha, A. Loulizi, I. El-Shawarby, and M. H. Almannaa, "Development and preliminary field testing of an in-vehicle eco-speed control system in the vicinity of signalized intersections," *IFAC-PapersOnLine*, vol. 49, no. 3, pp. 249–254, 2016.

[3] H. Jiang, J. Hu, S. An, M. Wang, and B. B. Park, "Eco approaching at an isolated signalized intersection under partially connected and automated vehicles environment," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 290–307, 2017.

[4] Z. Wang, G. Wu, and M. J. Barth, "Cooperative eco-driving at signalized intersections in a partially connected and automated vehicle environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 2029–2038, 2020.

[5] Z. Zhang, F. Liu, B. Wolshon, and Y. Sheng, "Virtual traffic signals: safe, rapid, efficient and autonomous driving without traffic control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 6954–6966, 2021.

[6] T. Ghoul and T. Sayed, "Real-time signal-vehicle coupled control: an application of connected vehicle data to improve intersection safety," *Accident Analysis & Prevention*, vol. 162, Article ID 106389, 2021.

[7] H. Jiang, Z. Yao, Y. Jiang, and Z. He, "Is all-direction turn lane a good choice for autonomous intersections? a study of method development and comparisons," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 7, pp. 8510–8525, 2023.

[8] C. Ma, W. Hao, A. Wang, and H. Zhao, "Developing a coordinated signal control system for urban ring road under the vehicle-infrastructure connected environment," *IEEE Access*, vol. 6, pp. 52471–52478, 2018.

[9] Y. Guo and J. Ma, "Drl-tp3: a learning and control framework for signalized intersections with mixed connected automated

traffic," *Transportation Research Part C: Emerging Technologies*, vol. 132, Article ID 103416, 2021.

[10] M. Tajalli, M. Mehrabipour, and A. Hajbabaie, "Network-level coordinated speed optimization and traffic light control for connected and automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 6748–6759, 2021.

[11] J. Li and L. Peng, "Automated vehicle planning data-assisted method for estimating turning ratios and optimizing signals at connected arterials," in *Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 1057–1062, Indianapolis, IN, USA, September 2021.

[12] Y. Zhu, Z. He, and G. Li, "A bi-hierarchical game-theoretic approach for network-wide traffic signal control using trip-based data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15408–15419, 2022.

[13] Z. Yao, L. Shen, R. Liu, Y. Jiang, and X. Yang, "A dynamic predictive traffic signal control framework in a cross-sectional vehicle infrastructure integration environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1455–1466, 2020.

[14] B. Liu and A. El Kamel, "V2x-based decentralized cooperative adaptive cruise control in the vicinity of intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 644–658, 2016.

[15] S. Woo and A. Skabardonis, "Flow-aware platoon formation of connected automated vehicles in a mixed traffic with human-driven vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 133, Article ID 103, 2021.

[16] C. Ding, R. Dai, Y. Fan, Z. Zhang, and X. Wu, "Collaborative control of traffic signal and variable guiding lane for isolated intersection under connected and automated vehicle environment," *Computer-Aided Civil and Infrastructure Engineering*, vol. 37, no. 15, pp. 2052–2069, 2021.

[17] Z. Yao, L. Shen, R. Liu, Y. Jiang, and X. Yang, "A dynamic predictive traffic signal control framework in a cross-sectional vehicle infrastructure integration environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1455–1466, 2020.

[18] Y. Jiang, B. Zhao, M. Liu, and Z. Yao, "A two-level model for traffic signal timing and trajectories planning of multiple cavs in a random environment," *Journal of Advanced Transportation*, vol. 2021, Article ID 9945398, 13 pages, 2021.

[19] Y. Feng, D. He, and Y. Guan, "Composite platoon trajectory planning strategy for intersection throughput maximization," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6305–6319, 2019.

[20] Z. Yao, H. Jiang, Y. Cheng, Y. Jiang, and B. Ran, "Integrated schedule and trajectory optimization for connected automated vehicles in a conflict zone," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 1841–1851, 2022.

[21] C. Chen, J. Wang, Q. Xu, J. Wang, and K. Li, "Mixed platoon control of automated and human-driven vehicles at a signalized intersection: dynamical analysis and optimal control," *Transportation Research Part C: Emerging Technologies*, vol. 127, Article ID 103138, 2021.

[22] J. Li, L. Peng, and T. Z. Qiu, "Real-time predictive coordination based on vehicle-triggered platoon dispersion in a low penetration connected vehicle environment," *IET Intelligent Transport Systems*, vol. 15, no. 12, pp. 1548–1561, 2021.

[23] Z. Yao, Y. Jiang, B. Zhao, X. Luo, and B. Peng, "A dynamic optimization method for adaptive signal control in a connected vehicle environment," *Journal of Intelligent Transportation Systems*, vol. 24, no. 2, pp. 184–200, 2020.

[24] Z. Yao, B. Zhao, T. Yuan, H. Jiang, and Y. Jiang, "Reducing gasoline consumption in mixed connected automated vehicles environment: a joint optimization framework for traffic signals and vehicle trajectory," *Journal of Cleaner Production*, vol. 265, Article ID 121836, 2020.

[25] I. Finkelberg, T. Petrov, A. Gal-Tzur et al., "The effects of vehicle-to-infrastructure communication reliability on performance of signalized intersection traffic control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15450–15461, 2022.

[26] M. Sankaranarayanan, M. Chelliah, and S. Mathew, "A feasible rsu deployment planner using fusion algorithm," *Wireless Personal Communications*, vol. 116, no. 3, pp. 1849–1866, 2021.

[27] C. M Silva, A. Ll Aquino, and W. Meira, "Deployment of roadside units based on partial mobility information," *Computer Communications*, vol. 60, pp. 28–39, 2015.

[28] C. Ghorai and I. Banerjee, "A constrained delaunay triangulation based rsus deployment strategy to cover a convex region with obstacles for maximizing communications probability between v2i," *Vehicular Communications*, vol. 13, pp. 89–103, 2018.

[29] K. M. A. Alheeti, A. K. A. N. Alaloosy, H. Khalaf, A. Alzahrani, and D. Al_Dosary, "An optimal distribution of rsu for improving self-driving vehicle connectivity," *Computers, Materials & Continua*, vol. 70, no. 2, pp. 3311–3319, 2022.

[30] Z. Ahmed, S. Naz, and J. Ahmed, "Minimizing transmission delays in vehicular ad hoc networks by optimized placement of road-side unit," *Wireless Networks*, vol. 26, no. 4, pp. 2905–2914, 2020.

[31] Y. Wang, J. Zheng, and N. Mitton, "Delivery delay analysis for roadside unit deployment in vehicular ad hoc networks with intermittent connectivity," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 8591–8602, 2016.

[32] Y. Liang, S. Zhang, and Y. Wang, "Data-driven road side unit location optimization for connected-autonomousvehicle-based intersection control," *Transportation Research Part C: Emerging Technologies*, vol. 128, Article ID 103169, 2021.

[33] J. Barrachina, P. Garrido, M. Fogue et al., "Road side unit deployment: a density-based approach," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 3, pp. 30–39, 2013.

[34] Y. Li, Z. Chen, Y. Yin, and S. Peeta, "Deployment of roadside units to overcome connectivity gap in transportation networks with mixed traffic," *Transportation Research Part C: Emerging Technologies*, vol. 111, pp. 496–512, 2020.

[35] A. Olia, H. Abdelgawad, B. Abdulhai, and S. Razavi, "Optimizing the number and locations of freeway roadside equipment units for travel time estimation in a connected vehicle environment," *Journal of Intelligent Transportation Systems*, vol. 21, no. 4, pp. 296–309, 2017.

[36] M. Salari, L. Kattan, and M. Gentili, "Optimal roadside units location for path flow reconstruction in a connected vehicle environment," *Transportation Research Part C: Emerging Technologies*, vol. 138, Article ID 103625, 2022.

[37] Y. Fang, H. Min, X. Wu, W. Wang, X. Zhao, and G. Mao, "On-ramp merging strategies of connected and automated vehicles considering communication delay," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15298–15312, 2022.

[38] M. Du, S. Yang, and Q. Chen, "Impacts of vehicle-to-infrastructure communication on traffic flows with mixed connected vehicles and human-driven vehicles," *International Journal of Modern Physics B*, vol. 35, no. 6, Article ID 2150091, 2021.

[39] M. Rios, V. Marianov, and M. Pérez, "Locating fixed roadside units in a bus transport network for maximum communications probability," *Transportation Research Part C: Emerging Technologies*, vol. 53, pp. 35–47, 2015.

[40] S. Krauss, P. Wagner, and C. Gawron, "Metastable states in a microscopic model of traffic flow," *Physical Review E*, vol. 55, pp. 5597–5602, 1997.

[41] Y. Zhu, Z. He, and W. Sun, "Network-wide link travel time inference using trip-based data from automatic vehicle identification detectors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2485–2495, 2020.

[42] C. Bettstetter and C. Hartmann, "Connectivity of wireless multihop networks in a shadow fading environment," *Wireless Networks*, vol. 11, no. 5, pp. 571–579, Jan 2005.

[43] Y. Li, C. Tang, S. Peeta, and Y. Wang, "Nonlinear consensus based connected vehicle platoon control incorporating car-following interactions and heterogeneous time delays," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2209–2219, 2019.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 2570824, 13 pages
https://doi.org/10.1155/2023/2570824

Research Article

A Function Area Division Approach for Autonomous Transportation System Based on Text Similarity

Ke Huang,[1] **Caiting Chen**,[2] **Yao Xiao**,[1] **and Ming Cai**[1]

[1]*School of Intelligent Systems Engineering, Shenzhen Campus of Sun Yat-sen University, Shenzhen 518107, China*
[2]*Urban Mobility Institute, Tongji University, Shanghai 201804, China*

Correspondence should be addressed to Ming Cai; caiming@mail.sysu.edu.cn

Received 19 August 2022; Revised 27 October 2022; Accepted 21 March 2023; Published 2 June 2023

Academic Editor: Yanyong Guo

Along with emerging technologies and increasing demands, autonomation has become a significant trend in current transportation systems. Within this context, the autonomous transportation system (ATS) framework hinges on functions that serve as fundamental units to support its operation. Recognizing the divisions among these function areas can enhance our understanding of their meanings and interrelationships. This study introduces a method for dividing function areas within the ATS framework, grounded in text similarity, to mitigate reliance on subjective experience. Precisely, this method quantifies the similarity between functions based on their textual descriptions, and implements hierarchical clustering to delineate them into distinct function areas. To validate the effectiveness of this proposed method, a case study analyzing a vehicle automatic driving scenario was conducted. The results demonstrate that our approach can efficiently divide function areas, producing clustering outcomes that possess superior accuracy and purity when juxtaposed with reference classifications. Consequently, this method has the potential to facilitate the formulation of function areas within ATS, thereby supporting the autonomous operation and construction of ATS. Moreover, its applicability extends beyond ATS, showing promise for other clustering problems that involve multiple texts, such as in text classification.

1. Introduction

Along with emerging technologies and increasing demands, autonomation has become a significant trend in current transportation systems. Against this background, the concept of autonomous transportation system (ATS) emerged [1], aiming to realize autonomous perception, autonomous learning, autonomous decision-making, and autonomous action for transportation systems.

The construction of the ATS depends on the guidance of the architecture framework, and function area division is an essential part of the ATS framework's research. Related concepts for the task can be explained as follows. The services are the applications and values that the system can provide for users. For example, a transportation system can provide users with services of "parking space management," "freight administration," "vehicle emergency response," etc. The functions are the processes and activities used to support services. For example, the realization of the "parking space management" service relies on functions such as "get personal driver request," "process vehicle location data," "determine dynamic parking lot state," and "output parking lot information to drivers." The function areas are the sets of functions with common data processing characteristics and application scope. The function area division contributes to organizing the functions of transportation systems and sorting out intra-area correlation and interarea coordination. Furthermore, it benefits in determining key modules of ATS development.

As the basis of the ATS framework's research, the traditional intelligent transportation system (ITS) frameworks have more than 20 years of history, including typical frameworks of the United States, the European Union, and China. The research on the ITS framework of the United States started first and has been continuously improved since 1993. The latest version 9.0 [2] was released in

2020 to adapt to the transportation reform for automatic driving. The ITS framework of the European Union has been studied since the 1990s, and it was updated to version 4.1 in 2011 [3]. The ITS framework of China was studied in the early 21st century. It has not been updated and developed since the completion of version 2.0 in 2005 [4]. The three ITS frameworks have affected the development of other countries' and cities' ITS frameworks [5–8], and the ITS frameworks are evaluated by some researchers [9–11]. Table 1 lists the function areas of ITS frameworks in the United States, the Europe Union, and China, and the contents of the three are similar. However, the three ITS frameworks do not explain and demonstrate the division logic of function areas, which often relies on expert experience. Jiang et al. [12] once tried to use rough sets to identify new function areas of the ITS framework, which was a preliminary study of function area division methods. Still, there have been no further or similar research and applications.

Based on the traditional ITS frameworks, many explorations of the new ITS frameworks and improved ITS subsystems have been carried out recently. Especially, with the rapid development of Internet of Vehicles technology, some related ITS frameworks appeared, such as the vehicle-to-vehicle-to-infrastructure framework [13], the vehicular ad hoc network (VANET) communication architecture [14], and the VANET architecture assisted by unmanned aerial vehicles [15]. In addition, some studies were devoted to improving subsystems of the ITS frameworks, including public transportation systems [16, 17], vehicle tracking systems [18, 19], ITS security systems [20–22], ITS information systems [23, 24], and ITS communication systems [25, 26]. However, most current research on the ITS frameworks combines limited emerging technologies or only focuses on the ITS subsystems. There are few in-depth and detailed types of studies similar to the three ITS frameworks mentioned previously.

Regarding the function area division, the new ITS frameworks' research has updated the functions' content. Still, they mostly rely on subjective construction methods and have not yet formed a clear and complete methodology. As the transportation systems become more complex and the functions become more abundant, an adaptive function area division method is urgent to adapt to the dynamic evolution of the transportation systems. Actually, each function has some short texts that embody function characteristics, which can be used to cluster functions with commonality into the same function areas. Therefore, dividing function area can be regarded as clustering the function texts here, and this task is generally called text clustering. Text clustering is to group similar texts from a set of texts [27] and has many useful algorithms, such as hierarchical clustering [28], k-means clustering [29], eigenspace-based fuzzy c-means clustering [30], and deep embedding [31]. In most text clustering algorithms, text similarity is a necessary step. The text similarity approach can measure the commonality between two texts, which is often used in text clustering [32], text classification [33], information retrieval [34], and document matching [35]. In product-service systems and web service discovery, text similarity is applied in service clustering research [36, 37], which is similar to our research problem. Inspired by their works, text similarity is also adopted in establishing the ATS function area division method.

In this paper, an ATS function area division method is proposed based on text similarity, transforming the function area division task into a short text clustering task. Based on the method, the function similarity is measured by texts, and the functions are clustered by hierarchical clustering. Therefore, the method can adaptively form function areas, helping reduce the dependence on subjective experience and improve the efficiency of function area division work.

This paper consists of five sections. Section 2 describes the steps of the function area division methods, including function text processing, function similarity calculating, function area dividing, and method performance evaluating. In Section 3, a case analysis for a vehicle automatic driving scenario is provided, and the method performance is evaluated. Finally, Section 4 concludes all the work and discusses possible future improvements.

2. Methods of Function Area Division

The function area division will cluster similar functions based on considering various aspects of the transportation system operation. At last, the functions within the same function area have high similarity degrees, and the functions among different function areas have low similarity degrees, as shown in Figure 1.

The technical route for function area division is shown in Figure 2.

The steps are as follows:

(1) Function text processing: The text of the function name is converted into a phrase set by phrase segmenting and stops word removal. The texts of three function attributes, including "function provider," "process object," and "service object," are converted into word sets directly.

(2) Function similarity calculating: The function name similarity and function attribute similarity are calculated based on the Jaccard coefficient. Then, the comprehensive similarity matrix of functions is obtained with a weighted average of the two similarities.

(3) Function area dividing: The final function area result is obtained by hierarchical clustering and the silhouette coefficient. The function areas are named based on analyzing the keywords of the function texts. In addition, the functions in every function area are reclassified with the "operation stage" attribute, and a three-level function list is finally formed.

(4) Method performance evaluating: A function set of a vehicle automatic driving scenario is constructed,

Table 1: Function areas of ITS frameworks in the United States, the Europe Union, and China.

Countries	Function areas
United States	(1) Manage traffic (2) Manage commercial vehicles (3) Provide vehicle monitoring and control (4) Manage transit (5) Manage emergency services (6) Provide driver and traveler services (7) Provide electronic payment services (8) Manage archived data (9) Manage maintenance and construction (10) Support secure transportation services
European Union	(1) Provide electronic payment facilities (2) Provide safety and emergency facilities (3) Manage traffic (4) Manage public transport operations (5) Provide support for host vehicle services (6) Provide traveler journey assistance (7) Provide support for law enforcement (8) Manage freight and fleet operations (9) Provide support for cooperative systems
China	(1) Manage traffic (2) Provide electronic payment services (3) Provide traffic information services (4) Manage emergency rescue (5) Manage passenger transportation (6) Manage freight (7) Manage urban public transport (8) Support intelligent highway and safety driving assist (9) Manage transportation infrastructures (10) Manage ITS data

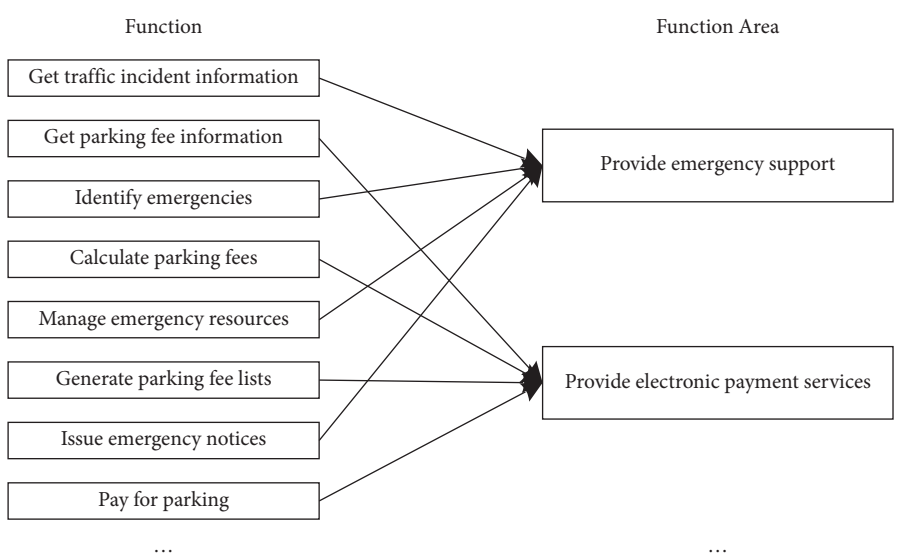


Figure 1: Schematic diagram of ATS function area division.

and the function areas are divided manually as reference classification. Based on the reference classification, the method performance is evaluated with accuracy and purity.

2.1. Function Text Processing. The text formats and processing of the ATS functions are introduced. In the ATS framework, a function is described by the function name and four attributes. The texts of the function name and three

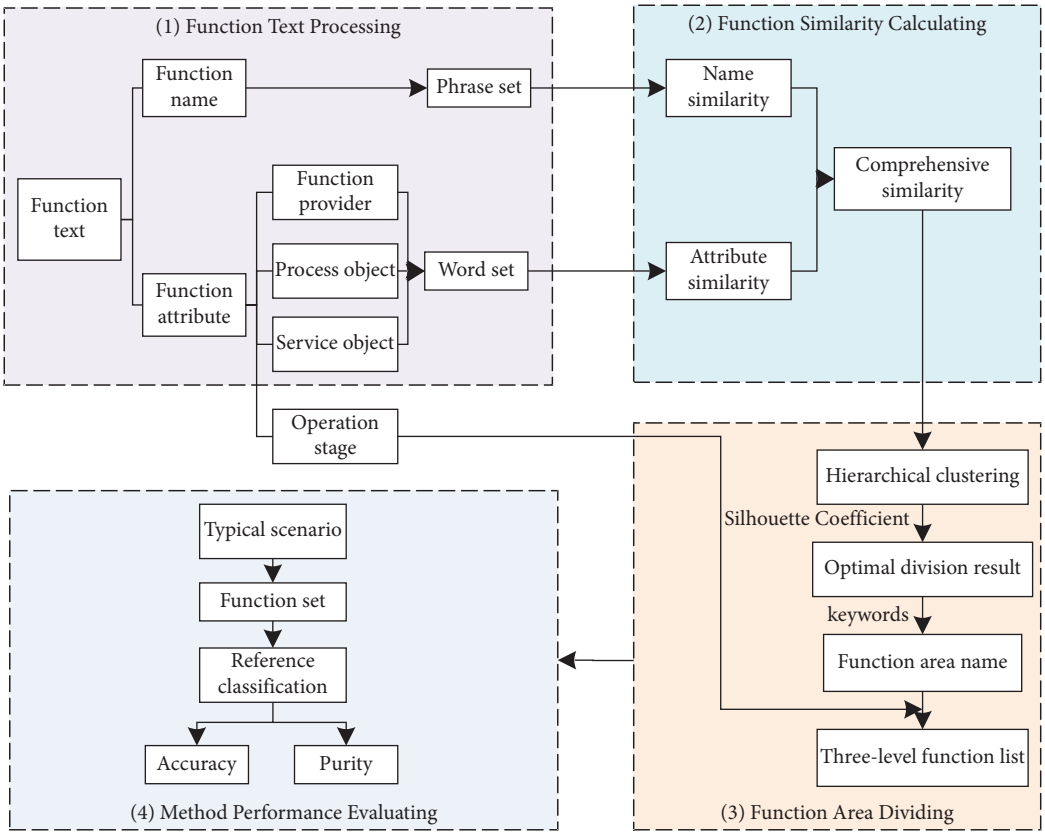


Figure 2: Technical route for function area division.

function attributes, including "function provider," "process object," and "service object," are processed for clustering functions. It should be noted that the function attribute "operation stage" is not used for clustering functions.

2.1.1. Function Name. The function name can summarize the function content and is a short Chinese text composed of "verb + noun or noun phrase," such as "dispatch emergency vehicles." Besides, extra information can be supplemented by parentheses, such as "provide traffic information query (traveler interface)."

Some auxiliary words, such as empty words and conjunctions, may exist in the function name text. Therefore, meaningful phrases must be extracted from the function name text for subsequent similarity calculation. The process of the function name text can be divided into the following two steps:

(1) Phrase segmenting: This work uses jieba, a mainstream Chinese phrase segmentation tool, to segment the function name texts into independent phrases, such as "monitor/passenger/anomaly/behavior," and each phrase is no more than three words.

(2) Stop word removal: In this work, stop words refer to meaningless symbols and redundant words. There are mainly two types: one is empty words, conjunctions, or other independent words after segmentation, and symbols, such as parentheses. Another is the verbs at the beginning of the text, such as "collect" and "process," which provide little help in distinguishing the function areas.

2.1.2. Function Attribute. The function attributes can describe the functional characteristics and embody the autonomous operation logic. Every function has four attributes:

(1) "Function provider": physical objects that provide the functions, including "user body," "system module," and "integration platform," such as vehicle-mounted equipment, infrastructures, and information platforms.

(2) "Process object": information objects used in the process of function realization, such as road network information and emergency events.

(3) "Service object": "user body" that can directly use functions, or "system module" and "integration platform" that directly use output results of functions, such as travelers, vehicle-mounted equipment, and information platforms.

(4) "Operation stage": system operation stages that can reflect the autonomous operation logic in the functions, including four stages of "perception," "learning," "decision-making," and "action".

The function attribute values are the specific contents of four function attributes, and the definition of all function attribute values can be found on the ATS website [38]. This research only studies functions whose function attributes have a single value.

The three function attributes, "function provider," "process object," and "service object," are used for clustering functions. The texts of their function attribute values are relatively simple, composed of short noun phrases without redundant components. Therefore, they can be directly converted to a word set for similarity calculation.

2.2. Function Similarity Calculating. The similarities of function names and function attributes are calculated first. Then, the comprehensive similarities between functions are calculated based on the two kinds of similarities.

The Jaccard coefficient can be used to calculate text similarity by measuring the overlap degree of phrases or words of two texts [37, 39]. Thus, the similarities of function names and attributes are measured based on the Jaccard coefficient.

2.2.1. Function Name Similarity. The function name similarity is calculated based on the phrase sets of two functions' names with the Jaccard coefficient:

$$r_{ij}^{(s)} = \frac{\left|W(s_i) \cap W(s_j)\right|}{\left|W(s_i) \cup W(s_j)\right|}, \tag{1}$$

where $r_{ij}^{(s)}$ is the similarity between the function i and the function j about the function name, and $W(s_i)$ is the phrase set of the function i about the function name.

2.2.2. Function Attribute Similarity. The function attribute similarity is calculated based on the word sets of two functions' attributes with the Jaccard coefficient:

$$r_{ij}^{(x_n)} = \frac{\left|C\left(t_i^{(x_n)}\right) \cap C\left(t_j^{(x_n)}\right)\right|}{\left|C\left(t_i^{(x_n)}\right) \cup C\left(t_j^{(x_n)}\right)\right|}, \tag{2}$$

where $r_{ij}^{(x_n)}$ is the similarity between the function i and the function j about function attribute x_n, and $C(t_i^{(x_n)})$ is the word set of the function i about function attribute x_n.

2.2.3. Comprehensive Similarity between Functions. The comprehensive similarity between two functions is obtained with a weighted average of the similarities of function names and function attributes:

$$r_{ij} = w_1 r_{ij}^{(s)} + w_2 r_{ij}^{(x_1)} + w_3 r_{ij}^{(x_2)} + w_4 r_{ij}^{(x_3)}, \tag{3}$$

where r_{ij} is the comprehensive similarity between the function i and the function j, $r_{ij}^{(x_1)}$, $r_{ij}^{(x_2)}$, and $r_{ij}^{(x_3)}$ are similarities of three function attributes: "function provider," "process object," and "service object." w_k is the weight of the k_{th} similarity, and $w_1 + w_2 + w_3 + w_4 = 1$.

The comprehensive similarity matrix of functions is

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{bmatrix}, \tag{4}$$

where $r_{ii} = 1$, $r_{ij} = r_{ji}$, and $0 \le r_{ij} \le 1$.

2.3. Function Area Dividing. All clustering results are obtained by aggregative hierarchical clustering, and the optimal clustering result is found by silhouette coefficient, which is the final function area result. Then, the function areas are named by analyzing the keywords of the function texts. At last, the functions in every function area are reclassified by the "operation stage" attribute, and a three-level function list is formed.

2.3.1. Function Clustering. Hierarchical clustering is a simple and effective unsupervised clustering method, which only needs the distances between samples. Specifically, this research adopts an agglomerative hierarchical clustering algorithm to cluster functions into function areas. The basic idea is to regard each function as an initial function area at the beginning and then continuously merge the two closest function areas until all functions are merged into one function area. The flow of function clustering by utilizing the agglomerative hierarchical clustering algorithm is shown in Figure 3.

In function clustering, the comprehensive similarity matrix should first be transformed into the comprehensive distance matrix, which is used as the initial distance between function areas. Then, during iteration, the distance between function areas is updated using the average sample distance between function areas:

$$d_{\text{avg}}(E_s, E_t) = \frac{1}{|E_s||E_t|} \sum_{i \in E_s} \sum_{j \in E_t} \text{dist}(i, j), \tag{5}$$

where $d_{\text{avg}}(E_s, E_t)$ is the average sample distance between function area E_s and function area E_t, and $\text{dist}(i, j)$ is the distance between function i and function j.

2.3.2. Optimal Division Result Determining. The hierarchical clustering can obtain all possible clustering results, whereas the optimal division result is what we need. Therefore, clustering performance should be evaluated to determine the optimal division result as the final function area division results.

The clustering performance is usually evaluated with external and internal indicators [40]. The external indicator compares the clustering result to the manual division result, whereas the internal indicator evaluates the clustering result directly. Compared with the external indicator, the internal

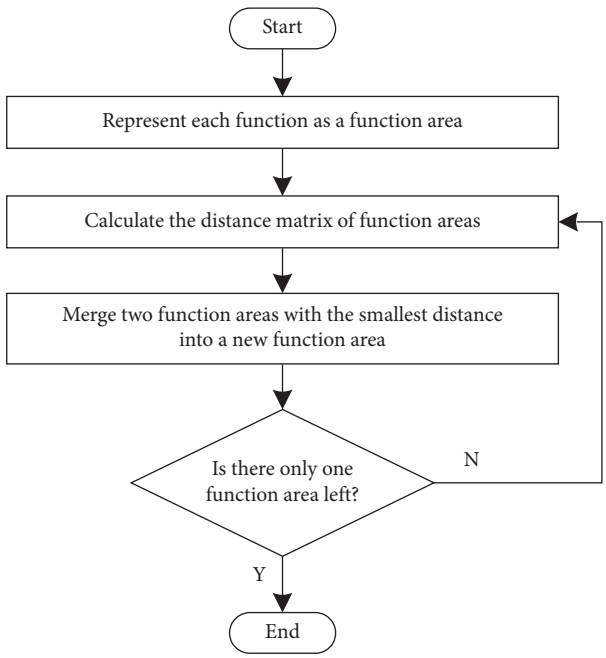


Figure 3: Flow of function clustering by the agglomerative hierarchical clustering algorithm.

indicator is more easily obtained and more adapted to the function area division work in different scenarios. Therefore, the internal indicator is selected to determine the optimal division result.

Silhouette coefficient [41] is a common internal indicator that considers both cluster cohesion and separation. Besides, it does not need to calculate the cluster center coordinates, so it is suitable for the clustering performance evaluation of our work with only sample similarity. The value of the silhouette coefficient ranges from −1 to 1, and a high value indicates that the intra-area similarity is high and the interarea similarity is low, representing a good clustering performance. Therefore, the clustering result with the maximum silhouette coefficient is the optimal division result.

Based on the function area division in the ITS framework, the maximum number of function areas can be set to 15. In addition, one function area that includes all functions is meaningless, so the minimum number of function areas is set to 2. The optimal division result is obtained by evaluating the silhouette coefficients of cluster numbers ranging from 2 to 15.

2.3.3. Function Area Naming. After obtaining the optimal division results, the function areas are named based on manual work because the number of the function area is no more than 15. For each function area, the three phrases with the highest frequency in the function name and the attribute value with the highest frequency in each function attribute are extracted as the keywords to provide references for naming manually.

The keywords and candidate names of function areas are concluded based on the function area research for the ITS framework, as shown in Table 2. The appropriate names can be directly selected according to the keywords.

2.3.4. Three-Level Function List Generating. Each function area has many functions, which is poor to display functions clearly, so the functions in all function areas are further categorized.

In each function area, the "operation stage" function attribute is used to reclassify the functions into "perception function," "learning function," "decision-making function," and "action function." As a result, a three-level function list is generated, whose structure is shown in Figure 4.

2.4. Method Performance Evaluating. The method performance of function area division needs to be evaluated. Thus, we construct a vehicle automatic driving scenario and obtain a function set of this scenario. Then, the function areas are divided manually as reference classification, and the method performance is evaluated by comparing the clustering result with the reference classification.

Two external indicators are used to assess the clustering performance compared with the reference classification: accuracy and purity [42].

2.4.1. Accuracy. The accuracy is the ratio of functions divided correctly. It indicates that the coherence between the clustering result and reference classification, and a value close to 1 represents the clustering performance is good. The accuracy is calculated with

$$\mathrm{ACC} = \frac{|F_{\mathrm{succ}}|}{|F|}, \tag{6}$$

where ACC is the accuracy, $|F_{\mathrm{succ}}|$ is the number of functions divided correctly, and $|F|$ is the total number of functions.

2.4.2. Purity. Since function distribution is not uniform in specific scenes, only the accuracy cannot reflect the clustering performance well. Therefore, the purity is introduced to help evaluate the clustering performance. It indicates the precision of clustering, and a value close to 1 represents the clustering performance is good.

The purity of each function area is calculated with

$$\mathrm{PUR}(C_k) = \frac{1}{|C_k|}\max\left(|C_k^s|\right), \tag{7}$$

where $\mathrm{PUR}(C_k)$ is the purity of the k_{th} function area, $|C_k|$ is the number of functions in the k_{th} function area, and $|C_k^s|$ is the number of functions belonging to the s_{th} function area of the reference classification.

The purity of the clustering result is calculated by the weighted average of the purities of all function areas

$$\mathrm{PUR} = \sum_{k\epsilon C} \frac{|C_k|}{|F|}\mathrm{PUR}(C_k), \tag{8}$$

Table 2: Keywords and candidate names of function areas.

Keywords	Candidate names
Vehicle	Provide vehicle control and safety Provide vehicle safety and aided driving
Traveler	Provide travel services
Nonmotorized traffic	Manage pedestrian and nonmotor vehicle Manage pedestrian and nonmotor vehicle safety
Weather environment	Monitor and manage environment
Road traffic	Manage traffic network Provide traffic management and planning
Freight	Manage traffic and freight
Bus operation, parking lot	Manage traffic operation Provide parking services Manage service facilities
Fee	Provide electronic payment services Provide electronic charging services
Infrastructure	Manage traffic facilities Manage infrastructures
Communication	Support communication
Emergency event	Provide emergency support Provide emergency rescue Provide emergency management/ response
Safety	Provide traffic safety Provide vehicle safety Provide pedestrian and nonmotor vehicle safety Provide public safety
Maintenance	Manage maintenance and construction
Data	Manage data Manage information service

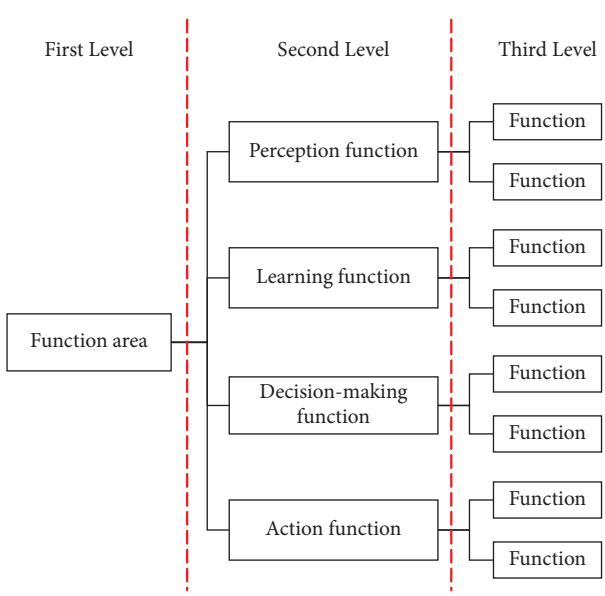


Figure 4: Structure of three-level function list.

where PUR is the purity of the clustering result, and $\mathrm{PUR}(C_k)$ is the purity of the k_{th} function area.

3. Scenario Verification and Result Discussion

A typical autonomous transportation scenario is constructed to verify the performance of the function area division method. First, a function set supporting this scenario is constructed. Then, the function areas are obtained by using the method. At last, the performance of the method is verified by comparing the clustering result with the reference classification through two external indicators: accuracy and purity.

3.1. Scenario Hypothesis and Function Set Construction. Vehicle automatic driving is a typical autonomous transportation scenario. In this scenario, we set an event that users want to drive from their homes to their workplaces on city roads. Before the travel, the users need to obtain travel information and plan routes. During travel, users need to drive cars and have safety requirements. Besides, they need to park their cars and pay traffic costs when arriving at the destination. Finally, after the travel, the system needs to provide travel evaluation and feedback to improve travel quality.

The completion of this event requires that the ATS framework has corresponding functions. As a result, a function set containing 174 ATS functions is constructed based on the autonomous operation logic (see Appendix I in Supplementary Materials (available here)), and the detailed contents of each function can be found in [38]. By referencing the traditional ITS framework, the 174 ATS functions are manually divided into 9 function areas as the reference classification (see Appendix I in Supplementary Materials (available here)). The function distribution of these function areas is shown in Table 3.

3.2. Function Area Division Result. Based on the method of Section 2, the functions can be clustered into function areas. The weights in (3) are set to $w_1 = 0.6$, $w_2 = 0.05$, $w_3 = 0.2$, and $w_4 = 0.15$, which are the optimal values by experiments.

The silhouette coefficients of cluster numbers ranging from 2 to 15 are calculated to judge the optimal cluster number, as shown in Figure 5. With the cluster number increasing, the silhouette coefficient increases first and then decreases. It reaches a maximum of 0.1989 when the cluster number is nine. Thus, the optimal cluster number is nine, the same as the function area number of the reference classification.

When the cluster number is nine, the accuracy and purity are both 0.8966, indicating that the function area division method can work well for function clustering.

Then, the keywords of the function name and three function attributes are extracted according to word frequency and combined with the candidate names in Table 2 to

Table 3: Function distribution of reference classification.

Function area	Autonomous operation logic				Function number
	Perception	Learning	Decision-making	Action	
Provide vehicle control and safety	14	14	13	13	54
Provide travel services	3	5	3	7	18
Provide electronic payment services	2	3	3	5	13
Monitor and manage environment	1	3	0	1	5
Manage traffic network	9	12	5	7	33
Manage traffic facilities	0	0	2	2	4
Support communication	0	0	0	2	2
Provide emergency support	7	7	6	9	29
Provide parking services	6	4	3	3	16
Total	42	48	35	49	174

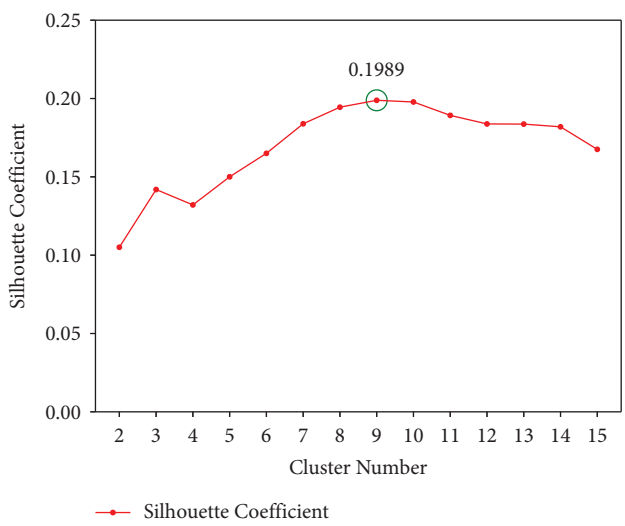


Figure 5: Silhouette coefficients of different cluster numbers.

name each function area manually. The keywords and recommended names of nine function areas obtained by clustering are shown in Table 4. Finally, the three-level function list is obtained based on the function attribute "operation stage" (see Appendix I in Supplementary Materials (available here)).

3.3. Analysis of Experiment Results. The effect of different text similarity weights and the clustering performance of optimal division results will be analyzed.

3.3.1. Effect Analysis of Text Similarity Weights. In (3), four weights need to be set when calculating the comprehensive similarity of functions. To explore the influence of weight values on clustering performance, we conduct the clustering experiments by changing weight values with a step size of 0.05 every time. The weight values and clustering performance of the top ten experiments by ranking the accuracy are shown in Table 5.

The accuracy and purity of the first seven experiments are the same, and most of the seven have nine function areas consistent with the reference classification. The last three experiments have more function areas than the reference classification, showing that the functions are divided into more details, resulting in higher purity and lower accuracy. The clustering result with high accuracy should be selected first because it is closer to the reference classification.

The relative relation of the four weights in Table 5 is $w_1 \geq w_3 > w_4 \geq w_2$, that is, the weights of the function name and "process object" are relatively large, while the weights of "function provider" and "service object" are relatively small. Specifically, the function name has the largest weight and $w_1 \geq 0.3$. It indicates that the semantic text information of the function name is rich to help most for function clustering. Among the three function attributes, the "process object" is the most important, the "function provider" is the least significant, and the "service object" plays a complementary role.

In Table 5, some experiments have different weights but the same clustering performance. In particular, experiment 1 and experiment 2 obtain different numbers of function areas, but their accuracies and purities are the same. The function distributions of the two experiments are shown in Figure 6, and the clustering results are close to the reference classification. However, the result of experiment 2 lacks the function area of "manage traffic facilities," and functions of this function area are mistakenly divided into the function area of "manage traffic network." In addition, some functions are incorrectly divided into the "monitor and manage environment" function area in experiment 1, while these functions are divided into correct function areas in experiment 2. Although experiment 1 and experiment 2 have different focuses on function clustering results, their correct function numbers are the same, resulting in their same accuracy and purity. In this scenario, both "manage traffic facilities" and "monitor and manage environment" have few functions. Because of the limited functions, it is difficult to perform an accurate performance comparison, and the two experiments can be considered similar clustering performances.

The clustering performance of the top 10 optimal experiments is not significantly different, especially since the index difference of the first four experiments is only 0.58%. When considering the number of function areas consistent with the reference classification preferentially, the clustering performance of experiment 1 is the best, followed by experiment 3 and experiment 4. Additionally, the two weights of experiment 3 are 0, which can simplify the calculation well without losing precision.

Table 4: Keywords and recommended names of nine function areas.

Function areas	Keywords				Recommended name
	Function name	Function provider	Process object	Service object	
C_1	Emergency, information, get	Emergency event management platform	Emergency event	Emergency event management platform	Provide emergency support
C_2	Vehicle, collect, information	Vehicle-mounted processing equipment	Vehicle operation control	On-board processing equipment	Provide vehicle control and safety
C_3	Provide, communication, interface	Vehicle-mounted interactive equipment	Communication information	Vehicle-mounted interactive equipment	Support communication
C_4	Road, electronic, indicator	Traffic facility management platform	Infrastructure information	Traffic facility management platform	Manage traffic facilities
C_5	Data, traffic, information	Road network management platform	Road network information	Road network management platform	Manage traffic network
C_6	Travel, generate, data	Traveler management platform	Travel demand information	Traveler management platform	Provide travel services
C_7	Information, weather, traveler	Vehicle-mounted interactive equipment	Weather environment information	Environmental climate information platform	Monitor and manage environment
C_8	Parking space, parking lot, information	Traffic facility management platform	Service location information	Traffic facility management platform	Provide parking services
C_9	Payment, parking fee, get	Charging equipment	Cost information	Vehicle-mounted payment equipment	Provide electronic payment services

Table 5: Weight values and clustering performance of the top ten experiments.

Rank	Function name (w_1)	Function provider (w_2)	Process object (w_3)	Service object (w_4)	Accuracy	Purity	Function area number
1	0.6	0.05	0.2	0.15	0.8966	0.8966	9
2	0.55	0.1	0.2	0.15	0.8966	0.8966	8
3	0.8	0	0.2	0	0.8908	0.8908	9
4	0.65	0.05	0.2	0.1	0.8908	0.8908	9
5	0.6	0.1	0.2	0.1	0.8793	0.8793	9
6	0.7	0.05	0.2	0.05	0.8736	0.8736	9
7	0.75	0.05	0.2	0	0.8678	0.8678	9
8	0.3	0.15	0.3	0.25	0.8506	0.9253	10
9	0.4	0.1	0.3	0.2	0.8448	0.9253	10
10	0.35	0.1	0.35	0.2	0.8448	0.9253	10

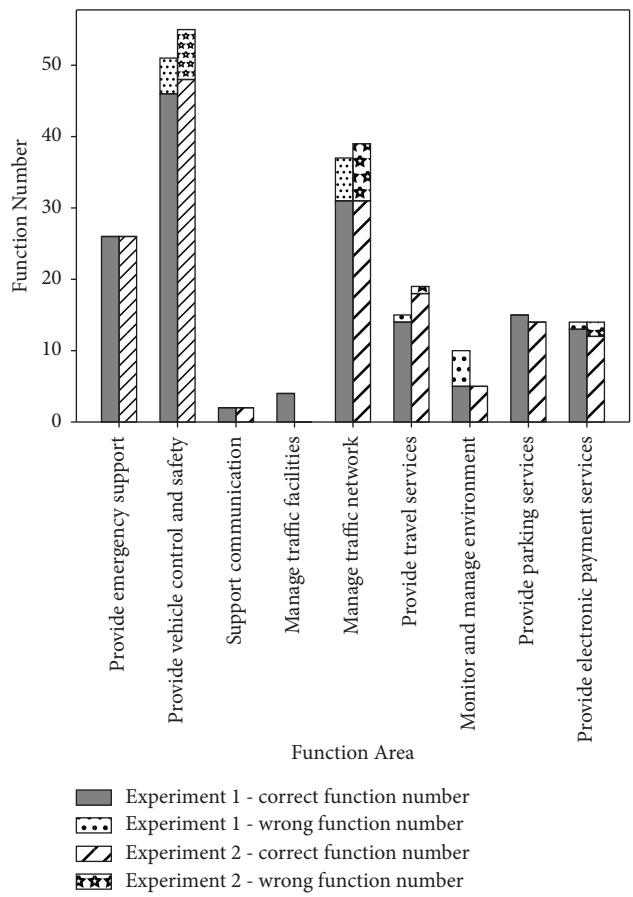


Figure 6: Clustering result distributions of experiment 1 and experiment 2.

In most experiments, the "function provider" weight is the smallest and even close to 0. Thus, we set this weight to 0 to explore the influence of the weights of the other two function attributes on the clustering performance.

Figure 7 shows the values of the accuracy and purity changing with the weight w_3 and w_4. On the whole, the accuracy and purity increase with w_3 increasing, and they can maintain a relatively stable high value in the range of $w_4 < 0.3$ and $w_3 \geq 0.2$. Besides, when $w_4 = 0$ and $w_3 = 0.2$, the purity reaches the highest value of 0.8908, which is the result of experiment 3 in Table 5. When $w_4 > w_3$, the accuracy and the purity both drastically decrease, whereas the purity is generally more significant than the accuracy, indicating that each function area is still relatively accurate. In short, the "process object" is the most critical influence factor and has the most weight.

3.3.2. Clustering Performance Analysis of Optimal Division Result. The accuracy and purity of the optimal division result (experiment 1) are both 0.8966, indicating that the function clustering precision is high and close to the results of the reference classification.

In the optimal division result, the purity of each function area relative to the reference classification is given in Table 6. Most of the function areas are divided relatively correctly; especially, the purities of C_1 (provide

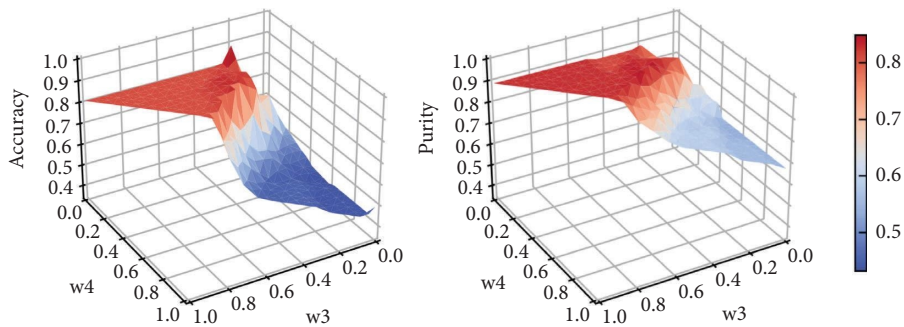


Figure 7: Influence of weights w_3 and w_4 on accuracy and purity.

Table 6: Purity of every function area.

Function area	Function number	Correct function number	Purity
C_1	26	26	1
C_2	51	46	0.9020
C_3	2	2	1
C_4	4	4	1
C_5	37	31	0.8378
C_6	15	14	0.9333
C_7	10	5	0.5000
C_8	15	15	1
C_9	14	13	0.9286
Total	174	156	0.8966

emergency support), C_3 (support communication), C_4 (manage traffic facilities), C_8 (provide parking services) are all equal to 1.

However, the clustering performance of C_7 (monitor and manage environment) is relatively poor. It is because the functions of the weather environment in this scenario are very few, and most of them are for travelers, causing it to confuse easily with the functions of C_6 (manage traveler services). If functions about the weather environment are added, the distinction between these two function areas might be improved.

In short, each function area is relatively independent, and the function similarity within the same function area is high, meeting the requirements of the ATS function area division. In addition, the keywords in the function text can show the characteristics of the function area to some extent, and the naming result can be consistent with the reference classification combined with the candidate name list. Therefore, the function clustering and naming methods proposed in this research are helpful to practical function area division.

4. Conclusion

A division approach that can adaptively divide the function areas is proposed based on text similarity for ATS framework research. Based on it, the function set of the vehicle automatic driving scenario is constructed, and the functions are clustered into function areas using the method. The experiment results show that the proposed method can effectively divide function areas, and the clustering results are relatively more accurate. Also, the text keywords can help to name function areas while reducing the dependence on subjective experience.

Even so, the ATS function area division research still has some limitations. First, the evaluation of the method performance depends on comparing the clustering result with the reference classification. Hence, the quality of the reference classification is vital, which still lacks inspection. Second, more realistic and complicated scenarios are necessary for the verification of the approach in the future. Last, the function areas are manually named here, while automatically generating function area names is worth studying, making it easy to verify a large number of scenarios. In the future, more efficient evaluation methods, more extensive scenario verification, and more appropriate function area name generation will be further studied.

The function area division method helps reduce the dependence on subjective experience and increases the efficiency of function area division work. In addition, the method is suitable for other ATS scenarios and clustering problems in other areas, such as text classification, whereas the clustering objects need to have multiple texts.

Data Availability

The data supporting the current study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work was funded by the National Key R&D Program of China (Grant no. 2020YFB1600400) and Shenzhen Science and Technology Program (Grant nos. 202206193000001, 20220817201129001).

Supplementary Materials

Appendix I. Table S1: Three-level function list. (*Supplementary Materials*)

References

[1] ATS Research Group, "Architecture of autonomous transportation system," 2021, http://auto-trans-sys.com/.

[2] United States Department of Transportation, *The National ITS Reference Architecture*, 2020,https://local.iteris.com/arc-it/index.html.

[3] FRAME Forum, "The FRAME architecture," 2020, https://frame-online.eu/.

[4] K. Zhang, T. Qi, D. Liu, C. Wang, R. He, and H. Liu, "The latest achievements of Chinese national ITS architecture (in Chinese)," *Journal of Transportation Systems Engineering and Information Technology*, vol. 5, no. 05, pp. 10–15, 2005.

[5] H. Borges, G. Knapp, and B. Eisenhart, "Development of Canadian architecture for intelligent transportation systems," *Artificial Intelligence and Intelligent Transportation Systems: Planning and Administration Transportation research record*, vol. 1774, pp. 80–89, 2001.

[6] X. M. Chen, Y. Lei, G. Jifu, Q. Yongshen, and G. Yanbin, "Development of beijing regional intelligent transportation system architecture," in *Proceedings of the 6th IEEE International Conference on Intelligent Transportation Systems*, pp. 560–565, Shanghai, China, October 2003.

[7] A. W. Sadek, R. Chamberlin, and P. R. Keating, "Use of the national architecture to develop an intelligent transportation systems strategic plan: case study for a medium-sized area," *Transportation Research Record*, vol. 1774, no. 1, pp. 71–79, 2001.

[8] R. Salazar-Cabrera and A. Pachon De La Cruz, "Design of urban mobility services for an intermediate city in a developing country, based on an intelligent transportation system architecture," in *Applied Computer Sciences in Engineering*, pp. 183–195, Springer International Publishing, Berlin, Germany, 2018.

[9] H. Vahidi and T. Sayed, "Using the Canadian ITS architecture for evaluating the safety benefits of intelligent transportation systems," *Canadian Journal of Civil Engineering*, vol. 30, no. 6, pp. 970–980, 2003.

[10] J. M. Golob, C. C. Stecher, and C. Felkins, "California statewide intelligent transportation systems plan evaluation: case study of conformity with national intelligent transportation systems architecture," *Transportation Research Record*, vol. 1826, no. 1, pp. 1–6, 2003.

[11] Y. Liu, J. Shi, and M. Jian, "Understanding visitors'responses to intelligent transportation system in a tourist city with a mixed ranked logit model," *Journal of Advanced Transportation*, vol. 2017, Article ID 8652053, 16 pages, 2017.

[12] C. Jiang, Q. Peng, K. Shi, X. Long, and F. Xu, "Rough set knowledge identification on the logic architecture of the national ITS of China (in Chinese)," *Presented at the 2007 Cross-Strait Symposium on ITS*, Tianjin University Press, Tianjin China, 2007.

[13] J. Miller, "Vehicle-to-vehicle-to-infrastructure (V2V2I) intelligent transportation system architecture," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 1062–1067, Eindhoven, Netherlands, June 2008.

[14] S. Sousa, "A new approach on communications architectures for intelligent transportation systems," *Procedia Computer Science*, vol. 110, pp. 320–327, 2017.

[15] A. Raza, S. H. R. Bukhari, F. Aadil, and Z. Iqbal, "An UAV-assisted VANET architecture for intelligent transportation system in smart cities," *International Journal of Distributed Sensor Networks*, vol. 17, no. 7, 2021.

[16] J. Z. Wang and Z. J. Wang, "Architecture design of urban intelligent transportation using cloud computing," in *Proceedings of the 2nd International Conference on Materials and Products Manufacturing Technology (ICMPMT 2012)*, pp. 2549–2552, Guangzhou, China, December 2013.

[17] M. Barth and M. Todd, "Intelligent transportation system architecture for a multi-station shared vehicle system," in *Proceedings of the 3rd IEEE Intelligent Transportation Systems Conference (ITSC-2000)*, pp. 240–245, Dearborn, Michigan, USA, October 2000.

[18] G. Chen, H. Cao, M. Aafaque, and J. Chen, "Neuromorphic vision based multivehicle detection and tracking for intelligent transportation system," *Journal of Advanced Transportation*, vol. 2018, Article ID 4815383, 13 pages, 2018.

[19] R. Salazar-Cabrera, A. Pachon De La Cruz, and J. M. M. Molina, "Design of a public vehicle tracking service using long-range (LoRa) and intelligent transportation system architecture," *Journal of Information Technology Research*, vol. 14, no. 1, pp. 147–166, 2021.

[20] M. Mallegowda, V. Nete, and A. Kanavalli, "Intelligent transportation system based on the principles of service-oriented architecture," in *Proceedings of the 12th IEEE and IFIP International Conference on Wireless and Optical Communications Networks (WOCN)*, Bangalore, India, September 2015.

[21] Q. Chen, A. K. Sowan, and S. H. Xu, "Assoc Comp, "A safety and security architecture for reducing accidents in intelligent transportation systems," in *Proceedings of the 37th IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, California, CA, USA, November 2018.

[22] Y. R. B. Al-Mayouf, O. A. Mahdi, N. A. Taha, N. F. Abdullah, S. Khan, and M. Alam, "Accident management system based on vehicular network for an intelligent transportation system in urban environments," *Journal of Advanced Transportation*, vol. 2018, Article ID 6168981, 11 pages, 2018.

[23] M. K. Natvig and H. Westerheim, "National multimodal travel information - a strategy based on stakeholder involvement and intelligent transportation system architecture," *IET Intelligent Transport Systems*, vol. 1, no. 2, pp. 102–109, 2007.

[24] W. G. Li, Y. Yamashita, M. W. Koendjbiharie, R. C. D. Juca, and A. MacIver, "The development and implementation of the operation system and data bank for the intelligent transportation system - sitcuo," *Journal of Advanced Transportation*, vol. 38, no. 2, pp. 163–186, 2004.

[25] S. li, Y. Cheng, T. Y. Zhou, and P. Y. Liang, "The improved precoding method in the VLC-based intelligent transportation system," *Journal of Advanced Transportation*, vol. 2022, Article ID 5951389, 9 pages, 2022.

[26] S. Din, A. Paul, and A. Rehman, "5G-enabled hierarchical architecture for software-defined intelligent transportation system," *Computer Networks*, vol. 150, pp. 81–89, 2019.

[27] A. Subakti, H. Murfi, and N. Hariadi, "The performance of bert as data representation of text clustering," *Journal of Big Data*, vol. 9, no. 1, 2022.

[28] L. S. Lomakina, V. B. Rodionov, and A. S. Surkova, "Hierarchical clustering of text documents," *Automation and Remote Control*, vol. 75, no. 7, pp. 1309–1315, 2014.

[29] C. Xiong, Z. Hua, K. Lv, and X. Li, "An improved k-means text clustering algorithm by optimizing initial cluster centers," in

Proceedings of the 2016 7th International Conference on Cloud Computing and Big Data (CCBD), pp. 265–268, Macau, China, November 2016.

[30] H. Murfi, "The accuracy of fuzzy c-means in lower-dimensional space for topic detection," in *Proceedings of the Third International Conference on Smart Computing and Communications SmartCom*, pp. 321–334, Springer International Publishing, Tokyo, Japan, December 2018.

[31] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proceedings of the Presented at the 33rd International Conference on Machine Learning*, Proceedings of Machine Learning Research, New York, NY, USA, June 2016.

[32] S. Liu, X. Wu, and J. Chai, "A dynamic clustering method of hot topics based on user interaction and text similarity," in *Proceedings of the 2021 14th International Congress on Image and Signal Processing BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1–5, Shanghai, China, October 2021.

[33] G. S. Reddy and T. V. Rajinikanth, "A text similarity measure for document classification," *IADIS International Journal on Computer Science and Information Systems*, vol. 12, no. 1, pp. 14–25, 2017.

[34] H. Li and J. Xu, "Semantic matching in search," *Foundations and Trends® in Information Retrieval*, vol. 7, no. 5, pp. 343–469, 2014.

[35] H. Pham, M. T. Luong, and C. D. Manning, "Learning distributed representations for multilingual text sequences," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pp. 88–94, Denver, CO, USA, June 2015.

[36] Z. Guo, "Research on knowledge service matching based on attribute similarity and clustering (in Chinese)," *Modular Machine Tool & Automatic Manufacturing Technique*, vol. 09, pp. 171–174, 2020.

[37] B. Jiang, L. Ye, W. Pan, and J. Wang, "Service clustering based on the functional semantics of requirements (in Chinese)," *Chinese Journal of Computers*, vol. 41, no. 06, pp. 1035–1046, 2018.

[38] ATS Research Group, "ATS component - function," 2021, http://auto-trans-sys.com/func.

[39] M. A. Alksher, A. Azman, R. Yaakob, R. A. Kadir, A. Mohamed, and E. M. Alshari, "A review of methods for mining idea from text," in *Proceedings of the Third International Conference on Information Retrieval and Knowledge Management (CAMP)*, Malacca, Malaysia, August 2016.

[40] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.

[41] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[42] J. G. Conrad, K. A. Al-Kofahi, Y. Zhao, and G. Karypis, "Effective document clustering for large heterogeneous law firm collections," in *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, Bologna, Italy, June 2005.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 9390468, 10 pages
https://doi.org/10.1155/2023/9390468

Research Article

Physical Architecture Simulation Based on System Dynamics Modelling for an Autonomous Transportation System Scenario

Chen Liang, Zhenwu Chen, Liang Yang, Zihan Guo, Xianglong Feng, Zhijun Huang, Xiang Liu, Yuqing Qian, Yanqing Xu, Xiaochun Zhang, and Ji Zhang

Shenzhen Urban Transport Planning Center Co., Ltd., 10/F Tower B, Building No. 9, Shenzhen Bay Science and Technology Ecological Park, Nanshan District, Shenzhen 518057, Guangdong, China

Correspondence should be addressed to Xiaochun Zhang; lijunyuan@sutpc.com and Ji Zhang; zhangji@sutpc.com

Received 19 September 2022; Revised 2 February 2023; Accepted 5 April 2023; Published 9 May 2023

Academic Editor: Rui Zhu

With the rapid growth of traffic demand and the development of intelligent technology, autonomous transportation system (ATS) has been considered as future transportation system. The evaluation of a designed physical architecture of ATS is necessary for understanding whether and how ATS operates and evolves automatically without manual works, and architecture simulation is a method for solving such research problem. Therefore, in this study, architecture simulation based on system dynamics modelling has been employed for physical architecture research. Under this methodology, a simulation case for scenario "Autonomous Vehicle on a Crossing in an Autonomous Transportation System" has been studied for understanding the information flow in ATS to evaluate and optimise its physical architecture of ATS. In conclusion, the system dynamic model could help researchers understand and evaluate the physical architecture's operation of ATS by scenario analysis.

1. Introduction

With further increase of traffic demand and the development of intelligent technology, it is obvious that intelligent transportation system (ITS) has been gradually taking place of the traditional transportation system in recent years. However, with further increase of traffic demand and the development of intelligent technology, autonomous transportation system (ATS) will take the lead. When automobiles were invented and ran alongside carriages, the traditional transportation system has already been established. However, this system was simple in comparison with ATS, and due to the backward computing technologies in 1850s, there were few studies in architecture designation by simulation or other designing theories.

In recent years, with the rapid development of ITS, technologies of transportation management (including information, computing, communication, and artificial intelligence) are increasingly used for improving the efficiency of transportation services. Compared to the traditional transportation system, there are less manual works in ITS; though manual work is still necessary in areas of transportation management, service, and so on. The following activities of ITS designation and optimisation reveal some development trends of transportation system: (1) results of traffic flow simulation demonstrate the information of traffic status, congestion prediction, vehicle guidance schemes, and so on; (2) the four-step model and activity-based model show the macroscopic traffic demand to assist administrators in designing traffic infrastructure, optimising public transportation schemes, and so on; and (3) the system dynamic model has been applied in public policy analysis, transportation resource supply-demand gaming analysis, and so on for the optimisation of system architecture and the allocation of resources. Furthermore, these tendencies would be upgraded in ATS, because the transportation system in ATS could operate and supply services automatically without manual work. This system will evolve automatically by the changing of transportation service demands. As a result, physical architecture of ATS would be more complex, but could still be studied for understanding the operation status, which makes it a starting point of

research to understand this status and evaluate the ATS architecture researchers designed.

Research on ATS is currently in the initial stages of exploration and early studies largely concentrated on theoretic or qualitative analyses of ATS. For instance, Crayton and Meier [1] examined the public health impact of autonomous reform of transportation and suggested a corresponding research agenda. Hancock et al. [2] analysed the effects of the autonomy of the transportation system from technological and ethical standpoints, with a preference for the drawbacks. Other research studies focused on autonomous services that rely on carriers and infrastructures to support specific scenarios, including emergency health aid [3], Internet of Vehicle [4], and public transportation [5, 6]. In general, the development of ATS still requires a steady base in theory and technology in which a general and personalised architecture is very essential [7].

The construction of the ATS architecture involves two key processes, namely, design and simulation. However, the majority of current work on the ATS architecture mainly focuses on the theoretical level. For example, Xu et al. [8] utilised complex network theory to examine and evaluate the reliability of the logical architecture for ATS, as well as to identify the crucial nodes. With the help of fuzzy theory, Tang et al. [9] analysed architectural information and established mapping associations based on the traits of complex systems. Zhang et al. [10] constructed the evolution model and detailed the evolution process to reveal the evolution mechanism of ATS, showing the development law of the ATS architecture. Moreover, when referring to the ITS architecture, most existing design methodologies have scenario or scale restrictions [11–13].

Additionally, being one of the primary techniques for contemporary engineering verification, microsimulation is extensively used in the transportation industry to analyse and resolve dynamic issues. Despite the fact that it performs well in specific application scenarios and offers great support for related research, e.g., public transportation [14] and urban freight transport [15] it has not yet had a complete set of tools for simulation and evaluation in transportation architecture [12, 16, 17]. To put it simply, ATS architecture design, such as other system architectures, has various requirements for reliability, adaptability, and versatility [8, 18, 19], and it also needs to establish a relationship with simulation tools to better match the simulation possibilities and system requirements [20]. According to those reviews, few studies on physical architecture analysis and evaluation of ATS currently reveal a significant gap in the ATS physical architecture simulation modelling.

After a comprehensive investigation, it was determined that modelling and simulation based on system dynamics (SD) is a feasible comprehensive method because it provides a systematic approach that can illustrate the value of entity feedback and delayed response [21]. In SD simulation, a "stock and flow diagram" has been employed for modelling (see Figure 1). Stock is defined as containers of transmitters, while transmitters could be transited to another stock through a flow. Level variable describes the volume of transmitters a stock saved, and speed variable describes the volume of transmitter flow transfers per time unit (e.g., second, hour, year, and so on.). Figure 2 shows the mechanism of system dynamics simulation.

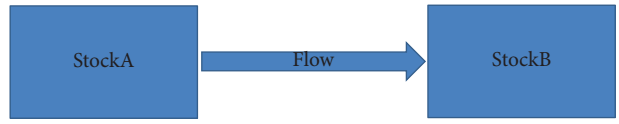


Figure 1: An example of stock and flow diagram.

Wen et al. [22] modelled the demand and supply of the designed public autonomous transportation system to analyse services. Sayyadi and Awasthi [23] integrated SD simulation and analytic network process to evaluate policies of the transportation system and to analyse their sustainability. Qu et al. [24] utilised SD to analyse logistics transportation systems for designing cost-effective IoT solutions. The transportation system is quite complicated since it is comprised of many different entities, which is particularly obvious in ATS. Additionally, there is currently no effective or appropriate way to design and simulate ATS architecture using SD, but it is highly promising for complicated and dynamic ATS architecture with customised demands.

In ATS architecture, information transfers through physical entities, which makes it critical to understand the status of this procedure for evaluating a designed architecture. SD modelling is a tool for simulating this procedure, and a methodology of utilising SD in ATS physical architecture has been established to solve such a problem. In this methodology, a simulation model based on SD has been introduced for understanding and evaluating an ATS physical architecture's operation by a scenario "Autonomous Vehicle on a Crossing in an Autonomous Transportation System," which shows that system dynamic model could be utilised in the designing of ATS physical architecture so that researchers and designers can test and evaluate the architecture they designed.

2. Architecture of ATS

According to previous studies, ATS includes components, function, service, physical entities, and sequential logics. For architecture research, those modules should be serviced to physical architecture, which contains a logic network established by physical entities composed by components and their functions, as well as sequential logics between physical entities composed by interoperability relationships.

2.1. Components. Components refer to the various parts that constitute the transportation system, which are the physical representations of the transportation system and serve as its foundation for both continued operation and maintenance. There are two types of components, one is transportation service demander, such as passengers and goods, which leads the formation of transportation system; the other one is supplier, such as transportation infrastructures (i.e., roads and traffic lights) and vehicles, which guarantees the

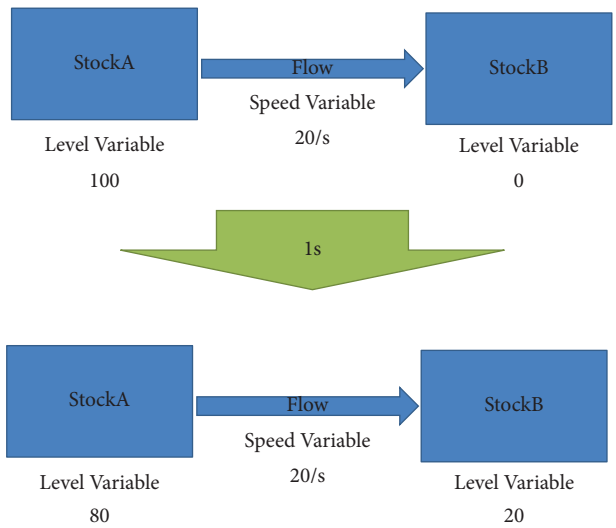


Figure 2: Mechanism of system dynamics simulation.

operation of the transportation system. Demanders and suppliers construct the transportation system, and they could not work standalone because according to the definitions, suppliers should work and adjust automatically for matching the demanders in ATS.

2.2. Function. Function is one of the basic elements in the architecture of ATS, which is driven by technology and utilised to implement numerous transportation-related services and guarantee the systematic and autonomous operation of the whole system, e.g., fetching the position of a vehicle, fetching the position of an obstacle, and recognising traffic signals.

2.3. Physical Entity. Physical entities are abstracted containers of real-world items made of components and functions with dynamic and measurable attribute information. By the rule of composing a physical entity, a component could only contribute to one entity, but one function can be realised by multiple entities. Physical entities should be categorised in accordance with the type of entity they transmit, including the type that develops the transportation system and the type that keeps it operating. From the theory of traffic flow, it should be separated into two categories, i.e., individuals and information. Level variables and capacity have distinct expressions as well. Although it is still feasible to divide up people and information, due to similar feature attributes, further subdivision is not possible from a procedural perspective.

The arrangement of physical entities follows principles of "sensing-learning-deciding-reacting" and "individual-module system." The information stream linking physical items generates the architecture. Physical entities can be divided into multiple hierarchies, while solitary physical entities should be avoided. The physical entity's structure is layered according to its level. The basic principle is that only one of the containment relationship and the sequential logical connection relationship can be selected. Meanwhile, rather than direct simulating, the operation result of the upper-layer physical entity depends on the lower-layer physical entity, which is the reaction of the lower-layer physical entity's simulation result. The simulation evaluation of the corresponding functional domains can be performed by simulating functions of various physical entities.

2.4. Service. Every function offers its service. To complete a service, the cooperation of different functions is necessary, and a structure of functions would be built. Function architecture is a base for constructing the physical architecture.

2.5. Sequential Logic. Corresponding to the flow component of the system dynamics model, the sequential logic is closely related to the data and the information required by certain specific functions, which is composed of information exchange pairs. An information exchange pair consists of a "source" physical entity, an information stream, and a "sink" physical entity. It indicates that a pair of physical entities completes an interaction through an information stream and forms process model through collaboration mechanism. The information stream corresponds to the interoperability relationship, which reflects the interaction relationship between physical entities, and consists of the two physical entities with information exchange in the architecture.

Sequential logic refers to the interconnection of the containers where the physical entities are located and forms the inherent logic of sequence. The information passing container parameters and the information receiving container parameters are the input and output of the sequential logic. There are several limitations to the sequential logic. For example, the sequential logic shall not directly connect physical entities of different levels with direct subordinate relationship, i.e., the lower-level physical entities shall not directly output data to the upper-level physical entities, while the upper-level physical entities shall not directly release data to the lower-level physical entities. Sequential logic, however, can be rationally explained even if there are no physical entities going in or coming out. Meanwhile, the process simulation simulates the sequential logic of the system dynamics architecture, and all simulation results are calculated from the bottom-level physical entities and sequential logic sets.

2.6. Physical Architecture. An essential component of the research on the system architecture of ATS is the connotation analysis and interaction mode of functional architecture, logical architecture, and physical architecture. The physical architecture primarily defines the numerous physical elements contained in the system and supports the realisation of logical architecture and functional architecture through these elements. For specific scenarios, physical architecture defines physical entities and information streams. It analyses the elements of ATS and builds system

infrastructure based on physical entities and sequential relationships.

The process of establishing the physical architecture starts with a single subservice. It is necessary to first identify a number of subfunctions that correspond to the service, and then the physical entities that are provided and undertaken by the subfunctions are determined in accordance with the logical sequence of these subfunctions. The source physical entity and subservice determine the information stream and data between two physical entities. A physical architecture is then generated by connecting these physical entities and information streams. The major focus of the physical architecture analysis is the structure that the ATS should have from the standpoint of the physical system, including multiple key contents such as components, scenarios, information exchange pairs, and interoperability relationships. Figure 3 shows the relationship among those concepts.

3. Concept Model of Scenario

It is a typical scenario that an autonomous vehicle drives in a typical crossing of a transportation network that includes many physical entities such as "driver," "walker," and "roadside equipment." Those physical entities are linked by interoperability relationships for information transmission. Different from ITS and traditional transportation systems, information and signal services should be supplied automatically. Therefore, a physical structure should be built for services including right-of-way allocation, collision alert, and giving way to pedestrians, and would simulate the information flow among those functions. A model has been built for modelling this scenario: when an autonomous vehicle drives to a crossing, information from other vehicles should be received. This procedure includes the following steps:

(1) Data recording other vehicles' unusual behaviour is transferred from vehicles to roadside equipment

(2) Roadside equipment treats that information into transport status and transfers to operation centre and transport information centre

(3) Transport information centre publishes transport status as broadcast information through information publish department to the vehicle's On-Board Unit (OBU), while operation centre shows regulation information by analysing this transport information to OBU

(4) OBU receives broadcast and regulation information as driving scheme and connects to Body Control Module (BCM) for asking avoidance feedback

(5) OBU sends collision information to interaction module for sending collision alert to vehicle controller

(6) Vehicle controller operates the vehicle by consuming collision alert information

According to the scenario, there are in all 9 physical entities and 10 sequential logics (the other flow is for information consumption) (see Tables 1 and 2).

Figure 4 shows the stock and flow diagram of this empirical model.

In the previous research, the model has been constructed by empirical methods, and the flow has been built already. However, this research shows a new method for quantitative evaluation by system dynamics modelling. First, parameter sets have been employed for setting the volume of information of each physical entity and the flow speed of each sequential logic; second, those parameters have been put as the stocks' initial level variables and flows' speed variables. Then, this system dynamic model was run, and finally the level variables of all physical entities in each simulation step would be collected.

4. Results and Discussion

4.1. Simulation Results. According to Section 3, a system dynamics model has been built. The settings of level variables of physical entities and max flow speeds of sequential logics are shown in Tables 3–4. In this research, the level variable shows the volume of information, which is a relative volume (Info. Unit, IU), and time unit of speed variable is set as "step."

The results of this simulation model in different parameter sets are shown in the following diagrams.

4.2. Discussion. According to the results of five parameter sets shown in Section 4.1, there is a model for analysing the scenario of information transmission when autonomous vehicles are driving on a crossing in an ATS.

According to Figures 5(a) and 5(b) (parameter set 1), when there are no enough spaces to save information from operation centre and information publish department in OBU and BCM, information will be stuck in operation centre and information publish department as information is full in OBU, and information solved by BCM could not be transferred back to OBU and transferred to next step (i.e., interaction module). At the same time, information saving space in BCM is full and could not receive further undersolving information from OBU. As a result, information stayed in this area, and the whole system failed.

When the information space of BCM raised (Figures 6(a) and 6(b), parameter set 2), this system performs better. However, it costs many steps to response to the information of unusual behaviours from other vehicles. When information from other vehicles occurs, level variables of BCM, OBU, information publish department, and operation centre raise to a certain level (up to more than 900 IU). This is a result of the raising of space in BCM because BCM could save nearly all the "unusual behaviours" information and such information could be solved in a long period. Meanwhile, due to the large space in BCM, OBU could send much more information to BCM. However, this system may fail if more and more information is received by OBU, which will eventually fill BCM to the full.

Another way may be faster (parameter sets 3 and 4). In these two sets, maximum information saving space of OBU raises, and BCM is 1000 IU or 10000 IU. When OBU's space

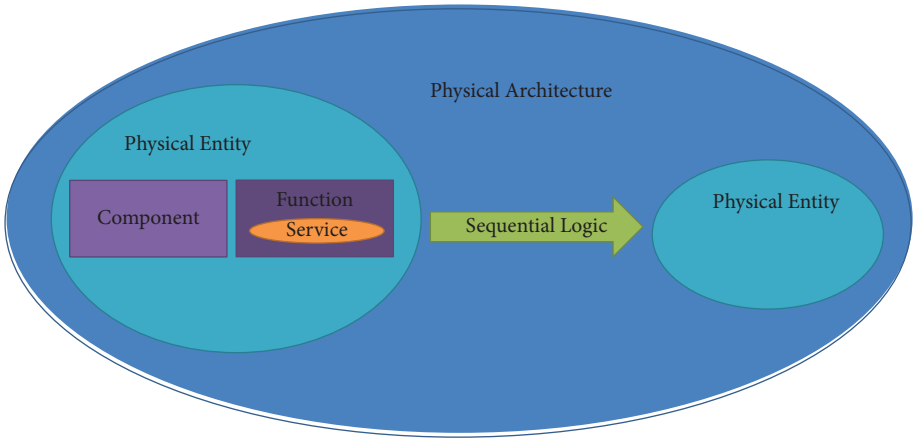


Figure 3: Relationship among concepts related to ATS.

Table 1: Physical entities of scenario.

#	Physical entity
1	Other vehicles
2	Roadside equipment
3	Operation centre
4	Transport info centre
5	Info publish department
6	OBU
7	BCM
8	Interaction module
9	Vehicle controller

Table 2: Sequential logics of scenario.

#	Sequential logic	From entity	To entity
1	Unusual behaviour	Other vehicles	Roadside equipment
2	Transport status a	Roadside equipment	Transport info centre
3	Transport status b	Roadside equipment	Operation centre
4	Broadcast info a	Transport info centre	Info publish department
5	Broadcast info b	Info publish department	OBU
6	Regulation info	Operation centre	OBU
7	Driving scheme	OBU	BCM
8	Avoidance feedback	BCM	OBU
9	Collision info	OBU	Interaction module
10	Collision alert	Interaction module	Vehicle controller

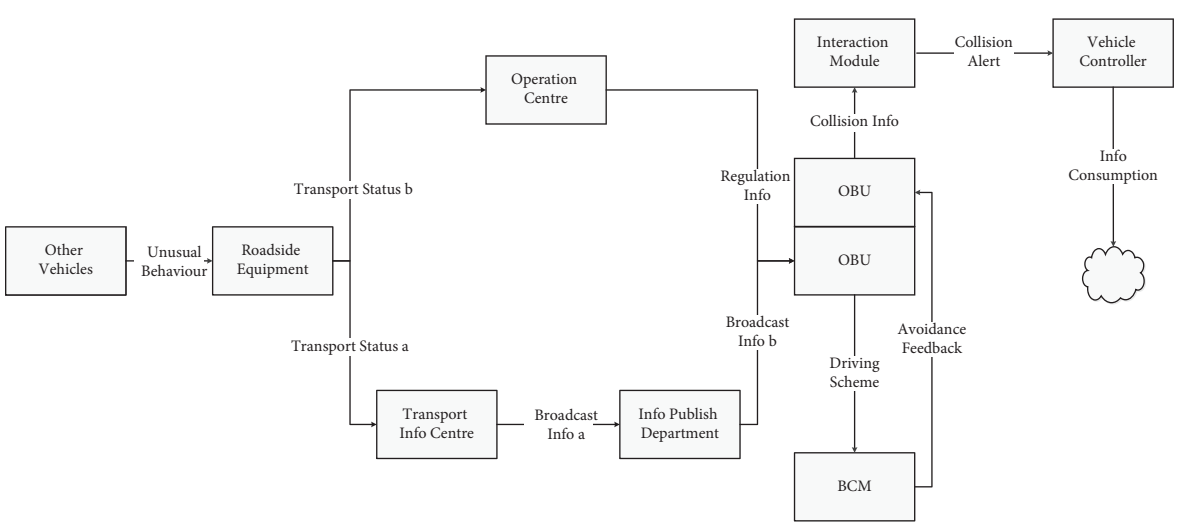


Figure 4: Stock and flow diagram of scenario "autonomous vehicle drives in a typical crossing of a transportation network."

Table 3: Minimum, maximum, and initial level variable (LV) of physical entities and maximum speed variable (SV) of sequential logics of scenario (parameter set 1).

Physical entity	Min. LV/IU	Max. LV/IU	Init. LV/IU	Sequential logic	Max. SV/(IU/s)
Other vehicles	0	Inf.	1000	Unusual behaviour	50 ± 5
Roadside equipment	0	10000	0	Transport status a	50 ± 5
Operation centre	0	100000	0	Transport status b	50 ± 5
Transport info centre	0	100000	0	Broadcast info a	50 ± 5
Info publish department	0	100000	0	Broadcast info b	50 ± 5
OBU	0	1000	0	Regulation info	50 ± 5
BCM	0	1000	0	Driving scheme	50 ± 5
Interaction module	0	10000	0	Avoidance feedback	50 ± 5
Vehicle controller	0	1000	0	Collision info	50 ± 5
				Collision alert	50 ± 5

Table 4: Parameter set 2 for research scenario.

Physical entity	Min. LV/IU	Max. LV/IU	Init. LV/IU	Sequential logic	Max. SV/(IU/s)
Other vehicles	0	Inf.	1000	Unusual behaviour	50 ± 5
Roadside equipment	0	10000	0	Transport status a	50 ± 5
Operation centre	0	100000	0	Transport status b	50 ± 5
Transport info centre	0	100000	0	Broadcast info a	50 ± 5
Info publish department	0	100000	0	Broadcast info b	50 ± 5
OBU	0	1000	0	Regulation info	50 ± 5
BCM	0	10000	0	Driving scheme	50 ± 5
Interaction module	0	10000	0	Avoidance feedback	50 ± 5
Vehicle controller	0	1000	0	Collision info	50 ± 5
				Collision alert	50 ± 5

Table 5: Parameter sets 3 and 4 for research scenario.

Physical entity	Min. LV/IU	Max. LV/IU	Init. LV/IU	Sequential logic	Max. SV/(IU/s)
Other vehicles	0	Inf.	1000	Unusual behaviour	50 ± 5
Roadside equipment	0	10000	0	Transport status a	50 ± 5
Operation centre	0	100000	0	Transport status b	50 ± 5
Transport info centre	0	100000	0	Broadcast info a	50 ± 5
Info publish department	0	100000	0	Broadcast info b	50 ± 5
OBU	0	10000	0	Regulation info	50 ± 5
BCM	0	1000/10000	0	Driving scheme	50 ± 5
Interaction module	0	10000	0	Avoidance feedback	50 ± 5
Vehicle controller	0	1000	0	Collision info	50 ± 5
				Collision alert	50 ± 5

Table 6: Level variables of other physical entities (parameter sets 3 and 4).

Physical entity	Max.	Min.	Average
Roadside equipment	0.45	0.00	0.13
Operation centre	0.16	0.00	0.02
Transport info centre	0.21	0.00	0.05
Info publish department	0.21	0.00	0.03
BCM	0.34	0.00	0.20
Interaction module	0.20	0.00	0.07
Vehicle controller	0.15	0.00	0.06

is high enough, there are no influences whether the BCM's space is large or not. The clearance time is shortened to 400 steps. These sets are better than those in set 2, but according to Figure 7 and Table 6, even though many physical entities are good running-condition, level variable in OBU is raised to the maximum information value (1000 IU). Therefore, these two sets could not solve the problem "more information" in parameter set 2.

According to Figure 8 and Table 8, the results of the final parameter set shows that during the running of simulation, information of "unusual behaviours" has been consumed, while all other physical entities run stably (level variable is between 0 IU and 1 IU). The last problem found in this scenario is caused by the information consumption speed. When raising the information consumption speed of sequential logics named "driving scheme," "avoidance feedback," "collision info," and "collision alert," which means when increasing the performance of autonomous vehicles, the system would run automatically and smoothly.

According to previous analysis, physical architecture can be only tested and verified by empirical analysis. However, in this research, as the quantitative analysis method "system dynamics simulation" has been utilised in the scenario "Autonomous Vehicle on a Crossing in an Autonomous

Table 7: Scenario parameter set 5.

Physical entity	Min. LV/IU	Max. LV/IU	Init. LV/IU	Sequential logic	Max. SV/(IU/s)
Other vehicles	0	Inf.	1000	Unusual behaviour	50 ± 5
Roadside equipment	0	10000	0	Transport status a	50 ± 5
Operation centre	0	100000	0	Transport status b	50 ± 5
Transport info centre	0	100000	0	Broadcast info a	50 ± 5
Info publish department	0	100000	0	Broadcast info b	50 ± 5
OBU	0	10000	0	Regulation info	50 ± 5
BCM	0	1000	0	Driving scheme	100 ± 5
Interaction module	0	10000	0	Avoidance feedback	100 ± 5
Vehicle controller	0	10000	0	Collision info	100 ± 5
				Collision alert	100 ± 5

Table 8: Level variables of physical entities except "other vehicles" (parameter set 5).

Physical entity	Max.	Min.	Average
Roadside equipment	0.52	0.00	0.31
Operation centre	0.09	0.00	0.03
Transport info centre	0.21	0.00	0.08
Info publish department	0.43	0.00	0.14
OBU	0.02	0.00	0.01
BCM	0.01	0.00	0.01
Interaction module	0.01	0.00	0.01
Vehicle controller	0.01	0.00	0.01

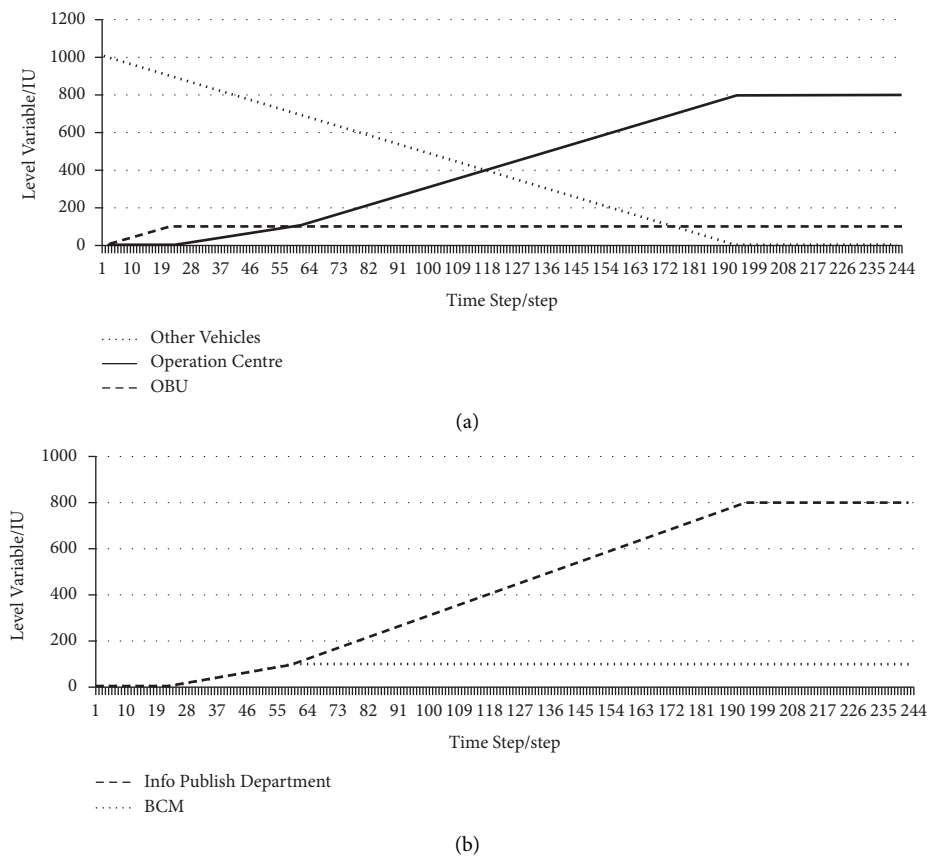
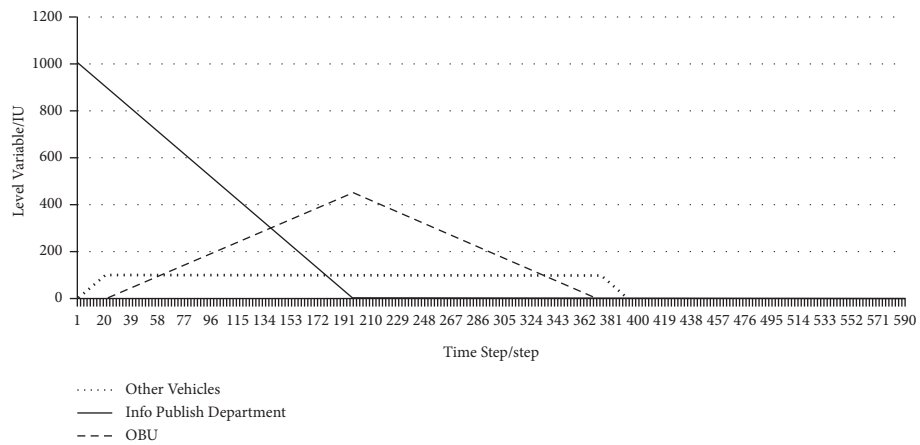
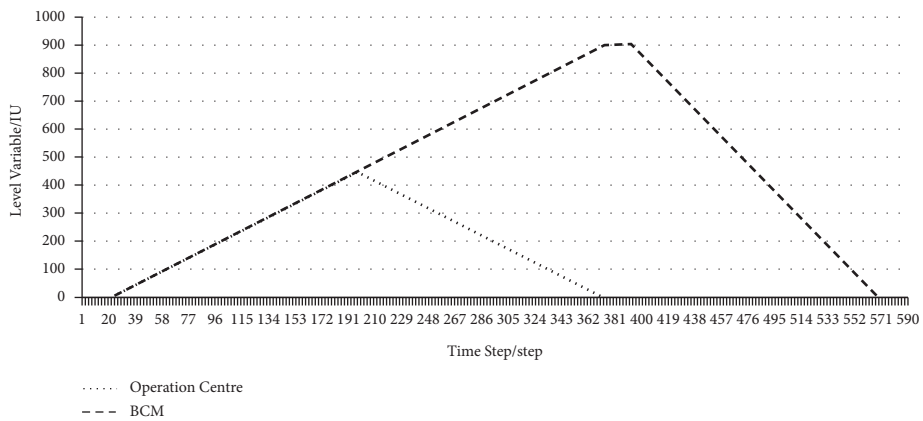


Figure 5: (a) Level Variables of Physical Entities (Parameter Set 1, OBU, etc.) and (b) Level Variables of Physical Entities (Parameter Set 1, BCM, etc.).



(a)



(b)

Figure 6: (a) Level Variables of Physical Entities (Parameter Set 2, OBU, etc.) and (b) Level Variables of Physical Entities (Parameter Set 2, BCM, etc.).

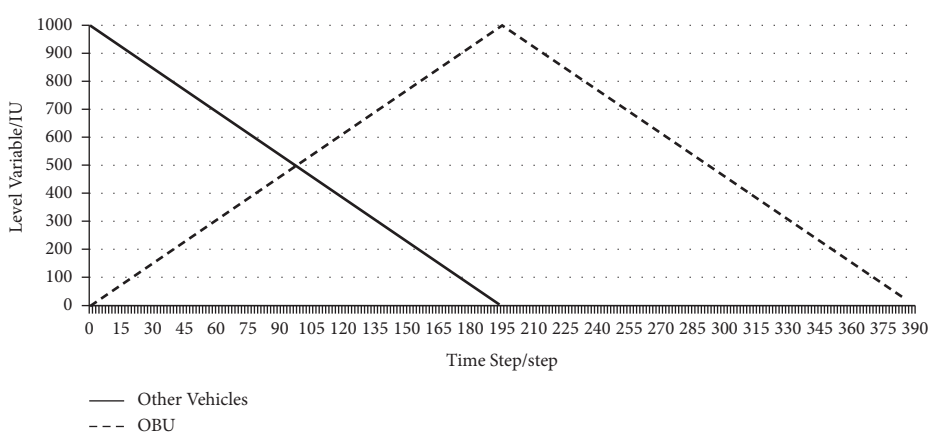


Figure 7: Level variables of "other vehicles" and "OBU" (parameter sets 3 and 4).

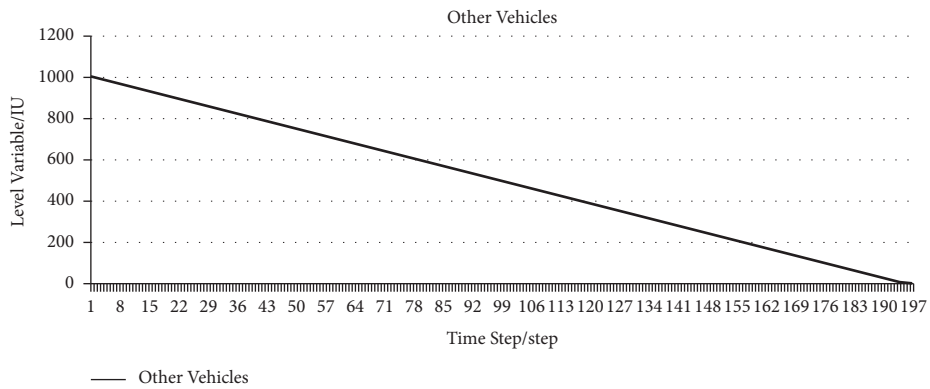


Figure 8: Level variables of physical entities (parameter set 5).

Transportation System," we could get the operation status of this physical architecture by the results of the SD model and also verify and optimise it conveniently.

5. Conclusion

In this research, system dynamics modelling has been utilised in the analysis of ATS' physical architecture. Estimating the necessary volume of information saving and the minimum speed of information transmitting are significant challenges in ATS designation. This research utilised system dynamics modelling and optimising a parameter set to evaluate this designed architecture. As a result, this research solved a problem of how physical architecture works in one generation of ATS. However, ATS has a character that it can evolve automatically, which includes several generations. In future research, there may be studies on the architecture simulation between generations of ATS.

Data Availability

The ATS physical architecture data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Key Research and Development Program of China (2020YFB1600400).

References

[1] T. J. Crayton and B. M. Meier, "Autonomous vehicles: developing a public health research agenda to frame the future of transportation policy," *Journal of Transport and Health*, vol. 6, pp. 245–252, 2017.

[2] P. A. Hancock, I. Nourbakhsh, and J. Stewart, "On the future of transportation in an era of automated and autonomous vehicles," *Proceedings of the National Academy of Sciences*, vol. 116, no. 16, pp. 7684–7691, 2019.

[3] M. Khalid, M. Awais, N. Singh et al., "Autonomous transportation in emergency healthcare services: framework, challenges, and future work," *IEEE Internet of Things Magazine*, vol. 4, no. 1, pp. 28–33, 2021.

[4] A. H. Sodhro, J. J. P. C. Rodrigues, S. Pirbhulal, N. Zahid, A. R. L. de Macedo, and V. H. C. de Albuquerque, "Link optimization in software defined IoV driven autonomous transportation system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3511–3520, 2021.

[5] M. Repoux, N. Geroliminis, and M. Kaspi, "Operational analysis of an innovative semi-autonomous on-demand transportation system," *Transportation Research Part C: Emerging Technologies*, vol. 132, Article ID 103373, 2021.

[6] M. Salazar, F. Rossi, M. Schiffer, and H. Christopher, "Onder, and Marco Pavone. "On the interaction between autonomous mobility-on-demand and public transportation systems," in *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2262–2269, Maui, HW, USA, November 2018.

[7] L. You, J. He, W. Wang, and M. Cai, "Autonomous transportation systems and services enabled by the next-generation network," *IEEE Network*, vol. 36, pp. 66–72, 2022.

[8] G. Xu, X. Liu, L. Zhong, Y. Xiao, and H. Huang, "Reliability analysis and evaluation of logical architecture for autonomous transportation system," *Journal of Railway Science and Engineering*, pp. 1–9, 2022.

[9] J. Tang, Y. Liu, Q. Fu, and H. Tuo, "Multi-dimensional mapping relationship modeling in autonomous transportation system based on fuzzy theory," *Forest Engineering*, vol. 38, no. 2, pp. 146–153, 2022.

[10] L. Zhang, Y. Xiao, S. Jiang, C. Xiong, S. Jin, and M. Cai, "A study on the evolution model of autonomous transportation system based on complex network," in *16th ITS China Congress (ITSAC 2021)*, pp. 141–148, China Machine Press, Beijing, China, 2021.

[11] M. Jakob and Z. Moler, "Modular framework for simulation modelling of interaction-rich transport systems," in *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 2152–2159, The Hague, Netherlands, October 2013.

[12] R. Salazar-Cabrera and A. Pachon, "Methodology for design of an intelligent transport system (ITS) architecture for intermediate colombian city," *Ingeniería y competitividad*, vol. 21, no. 1, pp. 49–62, 2019.

[13] G. Cavone, M. Dotoli, and C. Seatzu, "A survey on Petri net models for freight logistics and transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 6, pp. 1795–1813, 2018.

[14] P. Manser, H. Becker, S. Hörl, and K. W. Axhausen, "Designing a large-scale public transport network using agent-based microsimulation," *Transportation Research Part A: Policy and Practice*, vol. 137, pp. 1–15, 2020.

[15] V. Krivda, J. Petru, D. Macha, and J. Novak, "Use of microsimulation traffic models as means for ensuring public transport sustainability and accessibility," *Sustainability*, vol. 13, no. 5, p. 2709, 2021.

[16] C. G. Gomez-Marin, C. A. Serna-Uran, M. D. Arango-Serna, A. Comi, and A. Comi, "Microsimulation-based collaboration model for urban freight transport," *IEEE Access*, vol. 8, pp. 182853–182867, 2020.

[17] E. Cascetta, A. Cartenì, F. Pagliara, and M. Montanino, "A new look at planning and designing transportation systems: a decision-making model based on cognitive rationality, stakeholder engagement and quantitative methods," *Transport Policy*, vol. 38, pp. 27–39, 2015.

[18] N. A. Khan, V. Brujic-Okretic, and S. Khaddaj, "Intelligent integration framework for smart transport system," in *Proceedings of the 2016 12th International Conference on Intelligent Environments (IE)*, pp. 76–79, London, UK, September 2016.

[19] A. Aleti, B. Buhnova, L. Grunske, A. Koziolek, and I. Meedeniya, "Software architecture optimization methods: a systematic literature review," *IEEE Transactions on Software Engineering*, vol. 39, no. 5, pp. 658–683, 2013.

[20] J.-P. Brunet, H. Sohier, M. Yagoubi, M. Bisquay, P. Lamothe, and M. Pascal, "Simulation architecture definition for complex systems design: a tooled methodology," in *International Conference on Complex Systems Design and Management*, pp. 153–163, Springer, Berlin, Germany, 2019.

[21] S. P. Shepherd, "A review of system dynamics models applied in transportation," *Transportation Business: Transport Dynamics*, vol. 2, no. 2, pp. 83–105, 2014.

[22] J. Wen, Y. X. Chen, N. Nassir, and J. Zhao, "Transit-oriented autonomous vehicle operation with integrated demand-supply interaction," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 216–234, 2018.

[23] R. Sayyadi and A. Awasthi, "An integrated approach based on system dynamics and ANP for evaluating sustainable transportation policies," *International Journal of Systems Science: Operations and Logistics*, vol. 7, no. 2, pp. 182–191, 2020.

[24] T. Qu, M. Thürer, J. Wang et al., "System dynamics analysis for an Internet-of-Things-enabled production logistics system," *International Journal of Production Research*, vol. 55, no. 9, pp. 2622–2649, 2017.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 6909964, 15 pages
https://doi.org/10.1155/2023/6909964

Research Article

Information Exchange Pairs Simulation Method Based on Discrete Event Simulation for Autonomous Transportation System

Ji Zhang, Zhenwu Chen, Liang Yang, Zihan Guo, Xianglong Feng, Xiang Liu, Zhijun Huang, Puyang Deng, Xiaochun Zhang, and Chen Liang

Shenzhen Urban Transport Planning Center Co., Ltd., Shenzhen 518057, China

Correspondence should be addressed to Xiaochun Zhang; lijunyuan@sutpc.com and Chen Liang; liangc@sutpc.com

Received 12 October 2022; Revised 12 February 2023; Accepted 5 April 2023; Published 5 May 2023

Academic Editor: Zhicong Chen

Research on the complicated and indeterminate autonomous transportation system (ATS) has drawn increased interest in order to better address issues associated with contemporary transportation development. Hoping to make transportation more sustainable, economical, safe, convenient, and efficient, researchers try to address the expanding demands with the help of ATS characteristics, i.e., sensing, learning, deciding, and reacting autonomously. The majority of research is focused on the ATS architecture, which is currently in the design phase. To refine the relationship between entities in the architecture, this paper proposes DESIEP, a method based on discrete event simulation (DES) for information exchange pairs (IEPs) in ATS. This method can provide an intelligible description of the information stream while conveniently illustrating the control relationship of sequential logic amongst the entities in the architecture. It helps researchers better understand the running mechanisms driven by internal demands and external technologies of the system so as to diagnose, evaluate, verify, and optimize the designed architecture. Meanwhile, this method also supports further research on the system evolution and architecture analysis, including the functions, logic, and physical architectures of each generation, as well as their mapping relationships and parallel evolution mechanisms.

1. Introduction

With the rapid increase in volume of information, number of systems, and pace of iterative updates, the current transportation systems relying on people to command is difficult to meet the demand. It is vital to construct system architectures that adapt to the new generation of transportation system, i.e., autonomous transportation system (ATS). Additionally, the transportation system needs to transform from partial automation to full automation as a result of emerging technologies such as artificial intelligence, big data, mobile communication, and satellite positioning.

ATS stands for the trend in transportation development that is focused on the future. In addition to roads, vehicles, and pedestrians, ATS involves many elements, including roadside equipment and technical facilities, etc. In the design of the system, it is necessary to consider the problem of cooperative work in the multiautonomous individual network and the adaptive design of the system. Yue et al. [1] considered the joint design of UAV-USV-UUV networks for cooperative target hunting in the maritime battlefield when designing the intelligent control system composition and intelligent control structure of unmanned combat systems. Gan et al. [2] mainly consider how to choose the optimal communication network link to maximize energy efficiency in the process of UAV-assisted vehicle network communication. Xue et al. [3] considered the influence of the high-speed mobile environment on the new multi-access scheme of the 5G-Rlarge-scale Internet of Things and proposed the corresponding improvement scheme. It exhibits unique characteristics compared to other systems, including sensing, learning, deciding, and reacting autonomously. These characteristics determine that ATS is a dynamic and evolving system rather than a static one. Therefore, comprehensive principles are needed for the design of ATS

architecture. By establishing an ATS architecture, the system generation will be divided depending on the running mechanism that is driven by internal demands and external technologies of the system. The architecture is also examined in conjunction with the evolution method, including the functions, logic, and physical architectures of each generation, as well as their mapping relationships and parallel evolution mechanisms. The analysis method of system architecture can also refer to some algorithms from other fields. Chen et al. [4] proposed an efficient hybrid optimization algorithm based on mesh search and improved Nelder-Mead simplex (GS-INMS) for parameter identification of the PV model. Although the application field is different, the algorithm optimization ideas can be referenced in the optimization of architecture parameters.

In short, according to principles of ATS architecture, the standardized architecture design and implementation technology are eventually made possible, and the design reference scheme may be offered for typical transportation scenarios. In other words, it is first important to assure the accuracy of the existing system architecture or design principles in order to ensure that the research on ATS may be conducted properly.

Intelligent transportation system (ITS), which includes architecture reference for cooperative and intelligent transportation (ARC-IT) in the USA [5], ITS Canada [6], ITS in China [7], etc., are representative of the existing and reasonably mature research on transportation systems. For better research on the association and running mechanisms of various elements in the system architecture, many approaches to describing and modifying the logical and physical relationships between entities have been proposed. E.g., Michal and Moler [8] developed a modular framework called AgentPolis, which can support the implementation and simulation of transportation systems with a high level of interaction, but its abstraction of entities and agent behavior has to be perfected. Salazar-Cabrera and Pachon [9] proposed an ITS architecture design method allowing the integration and interoperability of new services with current technologies and service platforms, but its scenario has urban or regional restrictions.

However, ATS is more complicated, changeable, and understudied when compared to ITS. As mentioned by You et al. [10], component analysis, evolution prediction, and six other challenges must be addressed when creating ATS services, and current approaches have certain flaws and may be improved. One of these, component analysis, is closely tied to the internal attributes and relationships that characterize the ATS architecture and is specifically focused on the two entities that exchange information in the ATS, i.e., the information exchange pair (IEP).

The initial focus of information exchange research was on human relationships, but it has now evolved to include Internet data sharing situations. The research subject of the information exchange for the transportation system may be identified as the transportation entities that can be merged in pairs. Shen et al. [11] developed IoT-IDIVS, which incorporates data from transportation systems and can monitor vehicles in real-time with the assistance of roadside units. Gao et al. [12] proposed a vehicle-consensus scheme to realize traffic information retrieval and vehicle route management with information exchange. Jacobsson et al. [13] explored the automatic sharing of interoperable information on intermodal freight transportation systems. These studies utilize paired transportation entities, including travelers, carriers, infrastructures, etc., to discuss their data exchange process. However, IEPs in ATS also need to talk about the large-scale entity group and characterize the pair's particular impact connection, which is not covered in the previous study.

It is found that there is no corresponding research on IEP, and an appropriate method should satisfy the following criteria:

(1) The method can fulfill the demand of dynamic analysis in ATS architecture design.

(2) The method can be directly applied to various IEPs, and the application procedure is convenient and simple.

(3) The method can always meet the emerging application demands.

Simulation is a suitable strategy since there is a research demand for a dynamic analysis of the ATS design. Many researchers in the transportation field frequently use simulation to back up the viability of a proposal. The tools can be off-the-shelf simulation software [14–16] or a self-built simulation system platform [17–19], but the majority of them are based primarily on the defined scenarios to simulate and assess the functioning of certain entities. In the field of transportation, Kogler and Rauch [20] summarized the pertinent research on the analysis of multimodal and unimodal transportation supply chains using DES, pointing out that the existing research has an imprecise interpretation of the simulation model and scant details about the verification and validation process, and there are research gaps in many simulation details. Felde et al. [21] developed a transportation system simulator that enables parallel DESs on high-performance computers, but its current analysis of model validation and performance outcomes is insufficient. Sebastiani et al. [22] conduct DESs of urban BRT public transportation to optimize operations, but their scenarios are lightly constrained and concentrate more on energy consumption. Moreover, there is currently no method to support the description of the information exchange process or to apply it to the level of the transportation system architecture design.

To fill the current gaps, this work proposes a DES-based method for IEPs in ATS, named as DESIEP. The major contributions are summarized as follows:

(1) A modeling and simulation method is proposed to describe the IEPs in the ATS architecture. This method can also be applied to study diverse system structures.

(2) This method can visually describe the IEP and calculate the efficiency of information transfer. It will promptly provide feedback to the system for ATS

architecture design, which is convenient for users to design and modify.

(3) Through the building simulation model of the IEP for a concrete scenario, the results show that the model can be well compatible with the IEP in this scenario. The information retention of all links in each time step of the information exchange process can be counted, and the user can judge the rationality of the structure of the information interaction pairs by the value of the retention and the law of change.

The reminder in this paper is arranged as follows. In Section 2, some related work is presented to summarize the current challenges and related solutions in IEP. Section 3 introduces the related concepts and methods of IEP and DES. Section 4 describes the process and details of constructing the simulation. In Section 5, a scenario example is presented to demonstrate and analyze the simulation result. Finally, Section 6 concludes this study and discusses the future work.

2. Related Work

This section summarizes the challenges encountered in IEPs studies using simulation methods as well as related work to address them.

2.1. Emerging Challenges. In order to describe the interaction behavior of IEPs more accurately and make the research work efficient, three main challenges need to be faced, namely:

2.1.1. The Expression Ability of Exchange Behavior. The most basic function of IEPs is to express the logical control relationship between each module in the ATS architecture. Therefore, the simulation model of IEPs should have the ability to express the control relationship. The influence of this ability should include two aspects. One is that the information receiver changes with the information transmitted by the information producer, which is common in simulation scenarios for classical network communication [23] or data transmission [24]. The second is that the information generator includes a feedback mechanism based on the status of the information receiver, similar to how the state and control in a simulation of a traffic stream operate as a feedback loop to act constantly in real-time [25].

2.1.2. Description of the Amount of Information Transmitted. When the quantitative expression of ATS architecture is involved, the IEPs also need to express the amount of information transmitted per unit time. Therefore, the simulation model should be able to transmit information which is expressed discretely or describe the amount of information transmitted per unit time [26, 27].

2.1.3. Individual Differentiation. Different IEPs have different attributes, physical meaning, amount of received information and meaning, so the simulation model should be able to express the differences in the above points. Additionally, this would offer architecture-based customized configuration and have high levels of flexibility, stability, scalability, etc [28, 29].

2.2. Related Solutions. Analyzing the research methods that have been applied to IEPs, it is found that there are usually two research approaches: one is to build mathematical models, and the other is to build detailed ATS architectural models.

Build mathematical models: the simplest way is to directly set some basic parameter values. Most of the ATS architecture research is based on qualitative research, and IEPs mainly represent the logical control relationship between modules. As for the quantitative description, usually define some basic parameter values directly or according to the different levels set several parameters. But directly defining parameters can only extend the description of the results of qualitative analysis. It cannot accurately describe the information interaction on the specific interaction. So, a mathematical model related to the information receiver can be established, and it can be used to describe the amount of information transmitted by IEP [30, 31]. In this way, with the help of mathematical formulas, individual differences between different IEP can be expressed, and their sensitivity to receiving incoming information can be described. The mathematical model can express the individual differences of different IEPs, and the information amount of each IEP per unit time will change with the different information amount. Additionally, the model is optimized using data-driven concepts. For example, Hao et al. [32] optimize the decision equation based on rough set theory to facilitate car-following among cars with various features. The effect, which cannot be simply stated by a mathematical formula, will be significantly impacted if it is oriented to a complicated structure due to the implausible assumptions and laborious parameter calibration procedure. Therefore, some papers construct neural network models to describe IEPs [33–35]. However, none of these methods can effectively and intuitively obtain the physical meaning of IEP. In particular, the more complex the model is, the more difficult it is to understand its physical meaning.

Build detailed ATS architectural models: Research on the ATS architecture, there has been a lot of use system dynamics modeling for architecture. There is a split process in the modeling process. And IEP will be subdivided into multiple modules. Modules are connected by a fixed relation or a concrete equation, so IEPs can be viewed as a combination of several relationships.

Table 1 provides a thorough review of various strategies. In light of these methods, this paper investigates IEPS using the DES model. In Section 3, IEP and DES will be introduced in detail.

3. Concepts and Methods

3.1. Information Exchange Pair. Early on in the field of design, the idea of information exchange was employed to characterize design behavior [36]. It was progressively

Table 1: Challenges and representative solutions in the IEP (●: solved; ○: not-solved; ◎: partially).

Related solutions	The expression ability of exchange behavior	Description of the amount of information transmitted	Individual differentiation
Define some basic parameter	◎	◎	○
Establish a mathematical equation	●	◎	●
Construct the neural network model	◎	◎	○
Build detailed ATS architectural models	●	○	◎

described as the method of exchanging information amongst communicators, and today it is primarily used in computer science and Internet technology. The logical control relationship is represented by behaviors that are comparable in both physical objects and functional modules in ATS design. To exclusively express this control relationship in ATS architecture, researchers provide the idea of IEP.

IEP is different from the typical information exchange behavior. The behavior of information interaction is specifically directed at two individuals who have specific information interaction, which are particular objects with fine granularity. The physical granularity of objects in ATS architecture, however, is not just restricted to microscopic levels, but may also refer to a group with coarse granularity, as shown in Figure 1. And the items stated here pertain to more than just concrete physical things. Furthermore, the information exchange behavior typically only represents the logical relationship between the two described parties, while the IEP also needs to describe the specific influence relationship between the two, and the information transmitted is represented by the information stream contained in the IEP.

The relationship between the two must be fully outlined in the IEP. Information generator (IG), information receiver (IR), control relationship of sequential logic (CRSL), and information stream (IS) should all be included in the IEP as a whole. Some characteristics of the IEP have been described in Section 2, The following is a detailed description:

(1) *Independence*. All IS and CRSL are exclusive to the IG and IR of this IEP and have no connection to the other architecture components.

(2) *Closure*. There will be no external transmission of any IS or CRSL, which will only take place within this IEP.

(3) *Directionality*. IS is from IG to IR.

(4) *Information-Volume*. This is the data volume of IS delivered by IEP.

(5) *CRSL-Feedback-Mechanism*. In IEP, there is a CRSL whose direction is the direction of IS, i.e., usually from IG to IR. Due to its control relationship, the information transmission process will alter. Additionally, there will be information feedback effects, meaning that when IR gets information, it may respond by providing feedback to IG.

(6) *Physical-Meaning*. All processes of IEPs can be mapped in reality and find their realistic meaning.

IEP can describe the CRSL between various ATS architecture components, which is a further statement of the architecture relationship. The research on IEP enables users to design while also assisting scholars in their understanding of the interactions between components.

3.2. Discrete Event Simulation. A group of entities that interact and associate with certain rules in order to accomplish specific goals is referred to as a discrete event system. The discrete event system has discrete events, each of which takes place in a certain order or under specific circumstances [37].

After analyzing previous DES situations, the modeled items may often be split into different basic logical structural relationships, as shown in Figure 2 in the form of Petri nets.

In principle, DES may be used to replicate the IEPs of ATS since the internal sequential logic of IEPs in the ATS architecture can likewise be roughly separated into the aforementioned basic structures. Among DES models, the process model is based on specific process activities, and a comprehensive discrete event model is formed through the logical relationship between each process link. As shown in Figure 3, the bus's whole route, from departure to arrival, is simulated. Figure 3 shows the entire operation of the bus from the start to the end of the station. In Figure 3, These Modules ("BusSource," "LineToRoadnet," "Roadnet," "trafficmangement," el at.) respectively represent the factors affecting the efficiency of bus operation. Buses select specific routes according to the target stations and are affected by road structure and grade, traffic control facilities such as traffic lights, road conditions, possible emergencies, and queuing when entering the station. The whole process is complete when the bus completes its inbound behavior. Process simulation can describe the phases and processes of sequential flow. It is a technology based on process testing and quantitative analysis, whose essence is a numerical simulation technology based on running sequential processes. Process simulation may therefore effectively illustrate how IS is conveyed across IEPs.

This work constructs some fundamental process components in process simulation, and a process simulation model based on IEPs will be constructed through the disassembly and assembly of components. We can comprehend the functioning of IEPs through the statistics of information

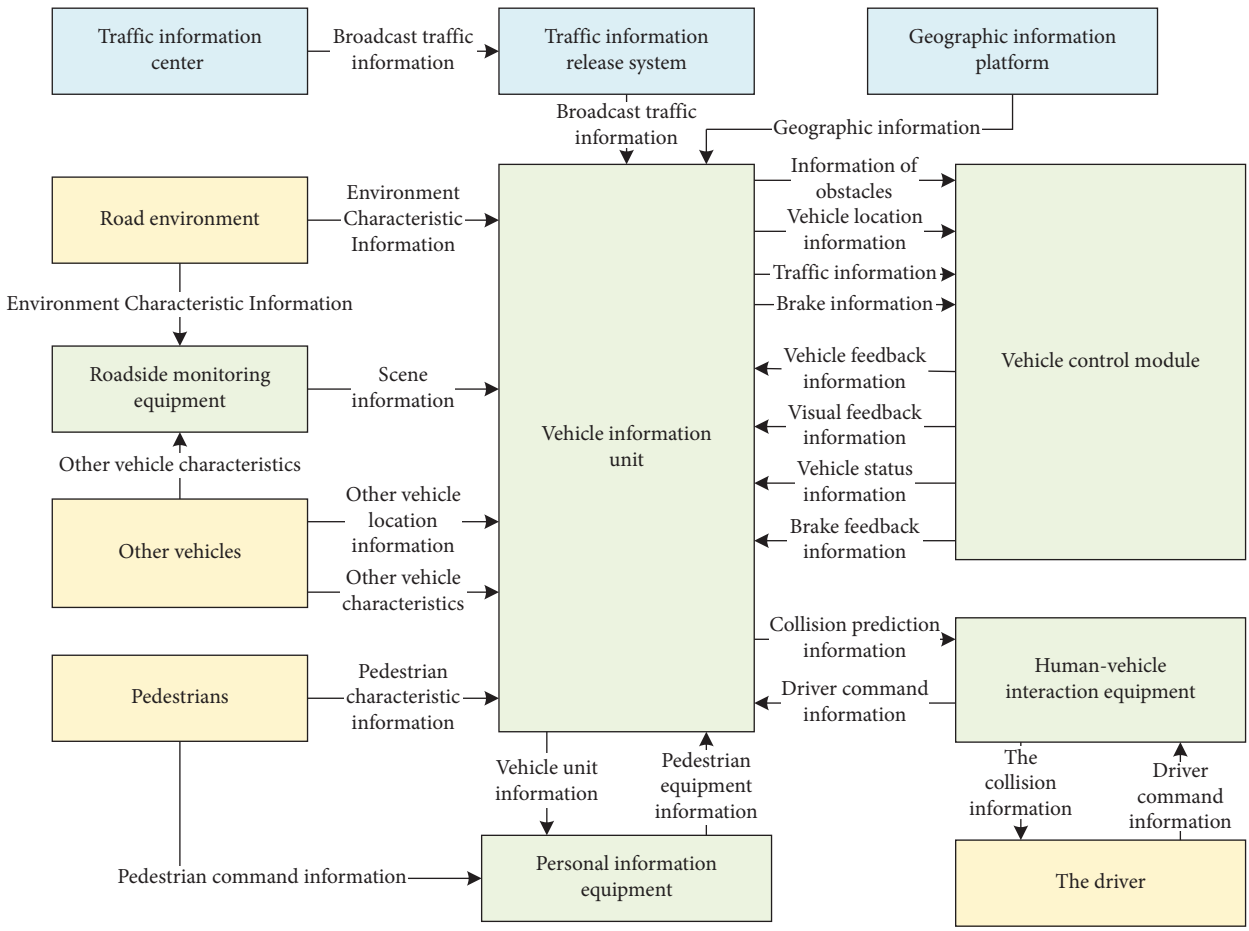


Figure 1: Example diagram of the ATS physical architecture.

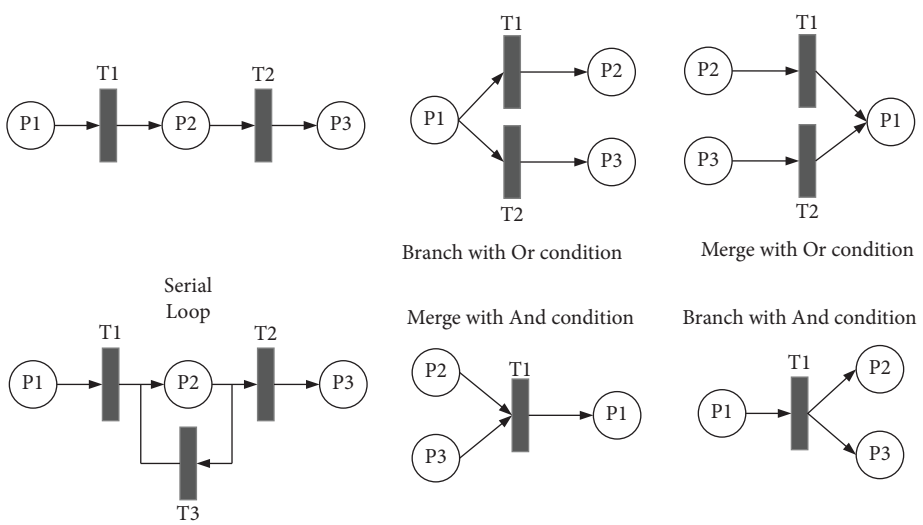


Figure 2: Basic logical relations.

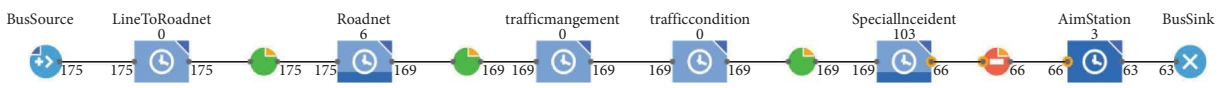


Figure 3: Discrete event simulation flow.

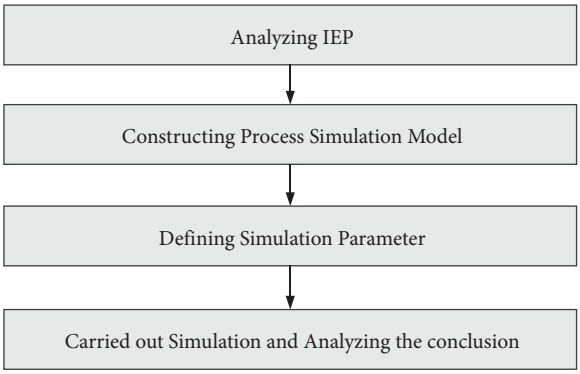


Figure 4: Overall workflow of simulation construction.

volume in each process module during simulation, and some calculation parameters can also be set to obtain the information desired by users.

4. Simulation Construction

The overall workflow of simulation construction is shown in Figure 4, which is divided into four parts, i.e., analyzing IEP, constructing process simulation model, defining simulation parameter, and drawing conclusion from simulation. The CRSL of IEP can be expressed intuitively through simulation. Regarding the information feedback mechanism that should also exist between IEPs, it is possible to track changes in the original link using the specific modifications to each link in the sequential process.

4.1. Analyzing IEP. IEPs in ATS are influenced by physical objects in the architecture and the IS between physical objects. The related process activity may be found by analyzing the physical object information of architecture that is provided in IEPs. The cornerstone for building process simulation models is process activities, and process activities may often be obtained through IEP in two different ways:

(1) Generally, in order to characterize IS transmitted by IEPs, architecture researchers would develop the unique process present in IEPs. Therefore, it is feasible to immediately locate relevant process activities utilizing the IG and IR of IEP.

(2) According to relevant data, users can query discrete activity events that are a part of the process. Then in accordance with the logical relationship (e.g., Or, And, Union, Serial, etc.), they can arrange and combine these activity events to obtain the process activities that are suitable for IEPs.

4.2. Constructing Process Simulation Model. The process activity model defines fundamental modules including Source, Sink, Delay, Split, Combine, Select, Hold, etc. Several modules can be merged to reflect the logical relationship mentioned above, and Figure 5 depicts their combined expressions.

The information entity is crucial to completing all process links in a process simulation. The full process simulation is finished when it is created from source and flows via each component module into sink in the direction of the connecting line. The process module can carry information entities and will specify particular functions upon initialization. The roles of the aforementioned modules and their corresponding construction features are as follows:

(1) *Source*. It produces information entities and is where all information entity processing begins. All processes begin with the generation of information flow, Source is the first component and has no upstream process activities.

(2) *Sink*. It acts as the end of the process for all information entities. All processes end with the collection of IS, Sink is the final component and has no downstream process activities.

(3) *Delay*. It is the basic component module. The module is delay if there are no exceptional events. The information entity will be stuck when passing through the module. Therefore, by giving its actual meaning, it is possible to characterize that an information entity is doing some activity when passing through the module. When modeling, delay is employed if there is just one process activity upstream and downstream of a process activity and the sequential information only remains in this location for a short while.

(4) *Split*. It divides a number of information entities. There will be several process activities downstream but only one upstream for the process activity that corresponds to this component. Examining if the two downstream processes are on the same logical timing level is crucial when modeling. Split will be utilized if the two processes are run simultaneously.

(5) *Combine*. It generates a new information entity by combining multiple existing information entities. Combine will be utilized if the process activity corresponding to the component contains more than one downstream process activity but only one upstream process activity, and there is a process activity corresponding to Split before that.

(6) *Select*. It provides multiple processing directions for information entities. This component's associated process activity is similar to Split, i.e., there are several downstream process activities. Only one of the branches, though, can be selected to carry out the further processing steps. For instance, while approaching a fork in the road, a person can only proceed in one direction. Select will be employed in this case.

(7) *Hold*. It prevents the information object from moving on to the following the module, forcing it to stay in the current one. The process activities corresponding to this component are subject to clear restrictions. E.g., when there are too many

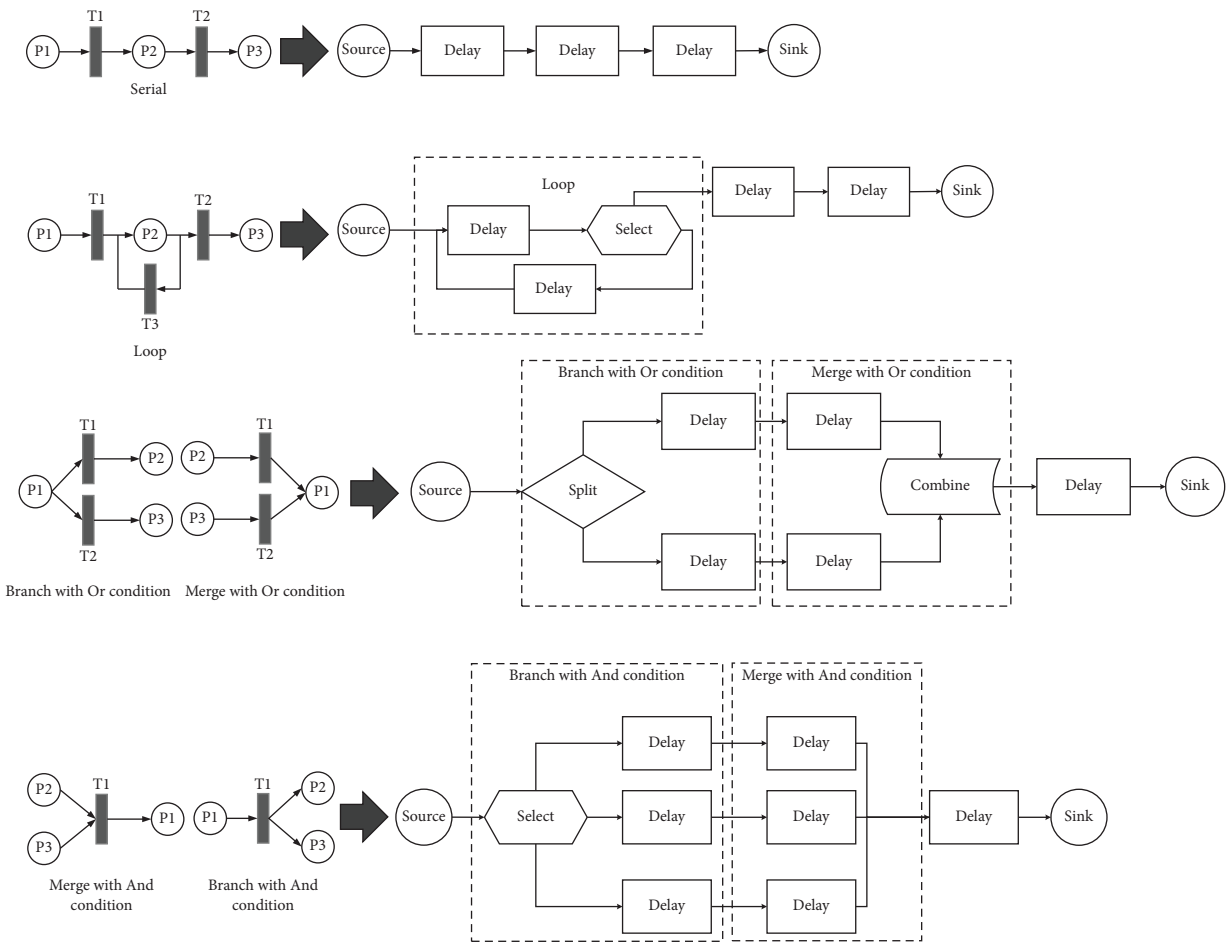


Figure 5: Expressing logical relationships in process simulation.

passengers in the subway station during peak hours, the staff would set up barriers to keep people out until the number of passengers lowered to a particular threshold. Hold can be used to illustrate this process.

Additionally, users may regulate how the aforementioned modules behave by establishing conditions. E.g., the number limit can be specified. The effectiveness of source producing information entities may decrease as the number of reaching sinks exceeds a particular threshold.

Following their extraction from IEPs, process activities are converted into process components based on the characteristics of each component. The aforementioned component modules are then joined by directed lines to create a process simulation model based on the logical relationships before and after the process activities.

4.3. Defining Simulation Parameter. There are two kinds of parameters for process simulation. The settings for the simulation's general running are one (e.g., simulation step length, simulation time, amount of information expressed by process entities, etc.). The second is the parameters set for the function of process modules, which should account for the actual basis and be tied to the physical meaning of the process activity module. Below is an explanation of the aforementioned ideas with an example.

Figure 6 depicts a process simulation representation of the aforementioned concept. The parameters for the simulation's running parameters are shown in Table 2. Several ones are set according to their physical meaning, such as the simulation step size of 10 ms for the process in 1 s and the IGs' recurring generation of 10 kb data.

Meanwhile, when setting the parameters of flow module functions, the actual situation is the reference, and the following are some examples: (1) Hard disk drives (HDDs) or solid-state drives (SSDs) are usually employed as backup storage. HDDs operate at a speed of roughly 120 M/s, whereas SSDs operate at a speed of 500 M/s to 3500 M/s. In this case, the storage speed of the simulation is set to 1000 M/s, and the time required for backup storage is 0.01 ms. (2) For video data, a two-stage target detection algorithm (detection speed is about 11 frames/s) or a single-stage target detection algorithm (detection speed ranges from 5.4 frames/S–65.8 frames/s) is usually adopted [38–40]. We selected 10 frames as the default speed for algorithm processing. Assuming that a frame of 1280 $*$ 720 video is about 10 kb, it takes about 0.4 s to process entity. (3) The speed of 5G data transmission is set to 1 Gb/s, and the time required for data transmission is 0.01 ms.

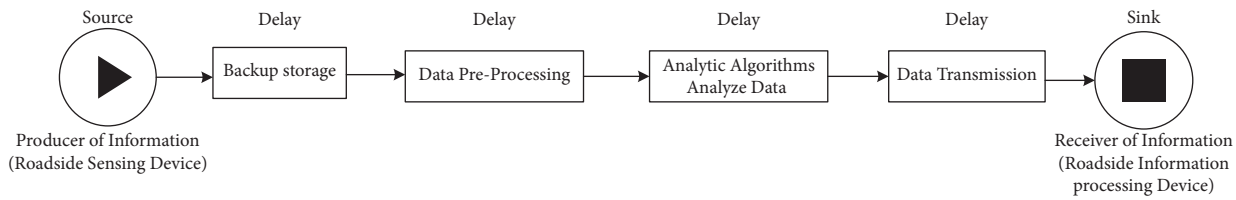


Figure 6: Diagram of process simulation (from "roadside sensing device" to "roadside information processing device").

Table 2: Table of simulation parameters.

Parameter names	Parameter values	Parameter units
Simulation time	1	Second (s)
Simulation step size	10	Millisecond (ms)
Process entity	10	Kilobyte (kb)
Backup storage (delay)	0.01	Millisecond (ms)
Data preprocessing (delay)	0.5	Millisecond (s)
Analytic algorithms analyze data (delay)	0.2	Millisecond (s)
Data transmission (delay)	0.01	Millisecond (ms)

5. Simulation Effect Analysis

The simulation scenario exemplified in this paper is shown in Figure 7 (see the Tables 3–5 for module meanings), in which information about environmental feature collected by the route is processed and transmitted before eventually entering the roadside sensing device. The information on environmental features collected from routes is divided into two categories, i.e., road pavement data and road structure data. They disassemble and model the format specification, transmission, and storage of data, respectively, and then the processed data is fused and collected into the roadside sensing device.

The structure of the simulation model has been determined not to change in the simulation process, and the set parameters are fixed without setting random variables. The simulation is conducted for the structure itself; the final simulation results will not change with the number of simulation. In this scenario, we assume that the data transmission rate over the line is constant. Therefore, the process of transmitting IS is considered to be a stable transmission process and can be divided into a certain time granularity. This time, the simulation time is 1s, and the simulation step is 10 ms. During transmitting IS of the environmental feature, the entities in all the processes represent a single data volume of 10 kb.

Different output parameters can be set to characterize the impact of user demands. E.g., we can directly count the number of information entities received by Sink throughout each simulation step if we want to find out the final response status to the information transmission of IR. And the period between modules is approximately 0.18, which is related to the simulation scenario.

As shown in Figure 8, the X-axis is simulation step, which records the duration of simulation, and the Y-axis is simulation count, which records the statistics of the transmitted process entities. The IS sent by IG goes through each link in the process, and the time when the IS reaches each module can be counted in simulation. We counted the time when IS arrived at the IR module and counted the number of IS arriving at IR at each moment according to the time series, so that the change in IS quantity received by IR with the simulation time could be obtained. Figure 8 shows the amount of IR received accumulation at each moment, and the relationship diagram shown in the figure is obtained in chronological order. So, Figure 8 reflects the change of the amount of IS received by IR with the simulation time. Firstly, we found that IR begins to receive information at 0.07 s. Then the volume of information keeps on growing over time, and a total of 63.7 Mb of data is received in a single second. When designing architecture, the information processing effectiveness of IEPs can be utilized as a guide to appropriately set the attribute values of the associated physical objects.

In the process, IG will set the information generation situation, and the process entities will flow through each module of the simulation. In this process, the number of process entities in each module of each simulation step is counted. The users can obtain the location of the information blockage from the output data. The location of the information jam is the bottleneck point of the simulation. By keeping track of the amount of information entities held in each Delay, the running bottleneck of the process can be determined. Figure 9 shows the number of information entities counted by each delay at the first second. The result indicates that while the majority of the information in Delay is dynamically balanced, some of it exhibits abnormalities. In a normal net, there are upstream and downstream modules except IG and IR. If the information received by the module from the upstream module cannot be consumed in time, the information will accumulate continuously. In a normal system structure, each component has an upper limit for storing information. If the information accumulates continuously, it will reach the storage capacity of the module, resulting in information blockage in the system. In reality, there is no actual object that can hold information indefinitely. Therefore, abnormal links should be dealt with first in optimization to avoid information blockage in system

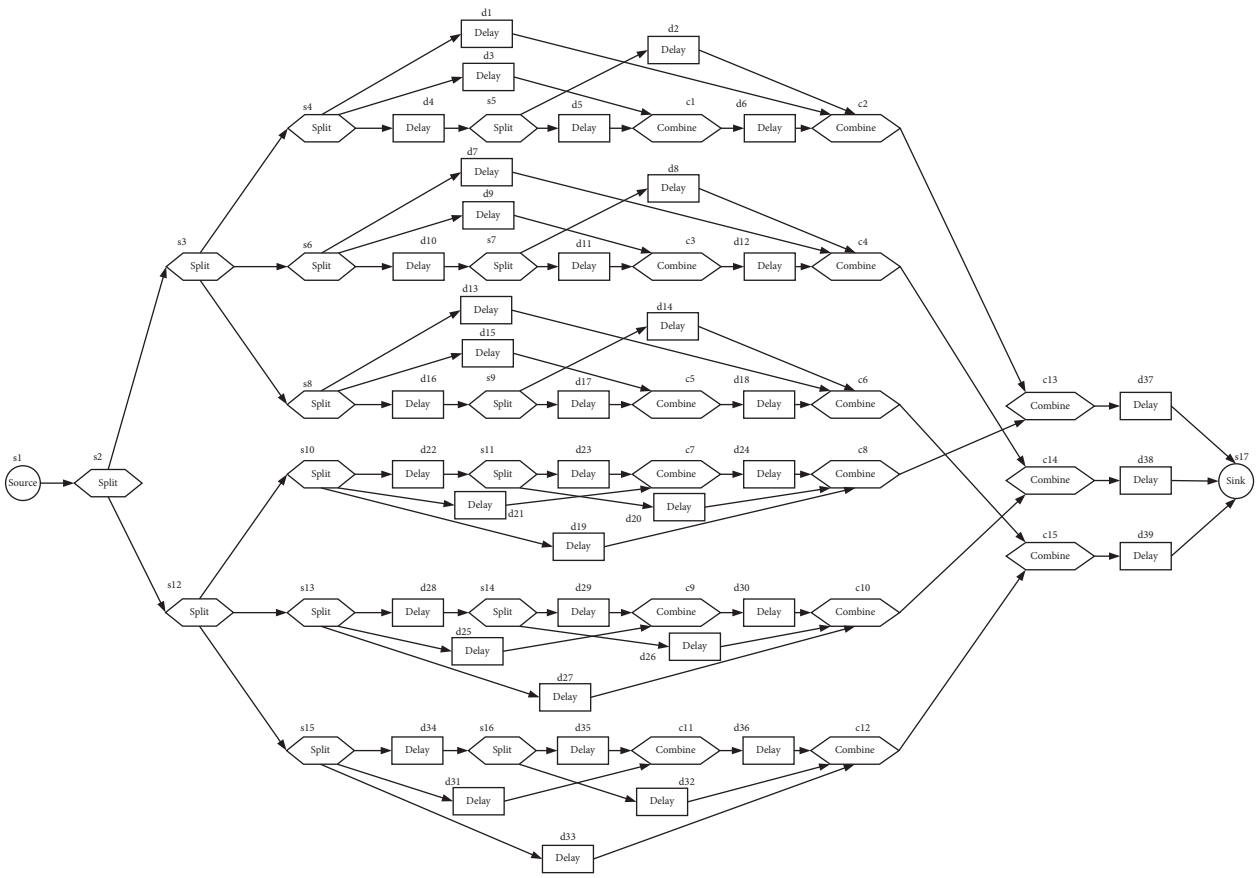


Figure 7: IEPs simulation (from "route" to "roadside sensing device").

architecture. Such abnormal links in the process might be prioritized for optimization when the ATS architecture is optimized based on the sequential logic in the architecture.

Furthermore, the eight special modules in Figure 9 are selected, and Figure 10 displays the temporal relationship between the quantity of internal storage entities. For each individual module, the change in the quantity of internal storage has an obvious periodicity. The period among modules is approximately 0.18, which is related to the simulation scenario. Meanwhile, it can be found that the modules located upstream of the same Combine module have basically the same value and trend, e.g., d7 and d12, or d13 and d18. Notably, d37 showed a delayed change trend with respect to d38 and d39, which may have been a result of the scenario setup or may have served as the point of penetration for the anomaly investigation.

Scenario experiments show how DESIEP may be used to characterize the architecture and its elements in an intuitive and effective manner. On the one hand, the process simulation model constructed based on IEPs analysis clearly describes the architecture structure and the transfer of IS. On the other hand, the execution of the simulation visualizes the internal operation of the architecture and the CRSL, and the simulation results can efficiently and accurately assist researchers in parsing, understanding, and improving the system designs in ATS.

In this case, we can be clearly observed that different technologies have direct influences on simulation parameter settings, such as different storage technologies affecting data storage time and different data transmission technologies affecting data transmission time. All these are directly related to the efficiency of the transmission of information flow. When designing the architecture, researchers can use this method to find the specific impact of different technologies on the efficiency of information transfer between architectural elements. Researchers can also adjust the architecture according to the degree of impact. For the already designed architecture, it can also evaluate and find the position of the architecture to be improved, so as to facilitate the researchers to adjust and optimize the already designed architecture. In addition, the results obtained by simulation statistics can also confirm the correctness of the structure of the IEP. The accessibility of the structure can be guaranteed if there is an information flow that completes the whole process. Recording each module can provide feedback on the change in sensitivity of the overall architecture to the transmitted information, which is convenient for users to observe the boundedness of the IEP.

Table 3: Table of simulation module (source/split/sink).

Module IDs	Module roles	Module names	Upstream modules	Downstream modules
s1	Source	Route	—	s2
s2	Split	Splitting data stream	s1	s3, s12
s3	Split	Road raw data format division	s2	s4, s6, s8
s4	Split	Standardized disassembly of road video data	s3	d1, d3, d4
s6	Split	Standardized disassembly of road millimeter wave radar data	s3	d7, d9, d10
s8	Split	Standardized disassembly of road lidar data	s3	d13, d15, d16
s12	Split	Road structure raw data format distinction	s2	s10, s13, s15
s10	Split	Standardized dismantling of road structure video data	s12	d22, d21, d19
s13	Split	Road structure millimeter wave radar data standardized disassembly	s12	d28, d25, d27
s15	Split	Standardized disassembly of lidar original data of road structure	s12	d34, d31, d33
s5	Split	Distinguish the road surface video data which has be stored	d4	d2, d5
s7	Split	Distinguish the road surface millimeter microwave radar data which has be stored	d10	d11
s9	Split	Distinguish the road surface lidar data which has be stored	d16	d14, d17
s11	Split	Distinguish the road structure video data which has be stored	d22	d23, d20
s14	Split	Distinguish the road structure millimeter microwave radar data which has be stored	d28	d29
s16	Split	Distinguish the road structure lidar data which has be stored	d34	d32, d35
s17	Sink	Roadside sensing device	d37, d38, d39	—

Table 4: Table of simulation module (delay).

Module IDs	Module roles	Module names	Upstream modules	Downstream modules
d1	Delay	Road surface video data transfer without specification and storage	s4	c2
d3	Delay	Road surface video data transfer without storage	s4	c1
d4	Delay	Road surface video data format normalization	s4	s5
d7	Delay	Road surface millimeter wave radar data transfer without specification and storage	s6	c4
d9	Delay	No specification of road surface millimeter wave radar data storage	s6	c3
d10	Delay	Road surface millimeter wave radar data format normalization	s6	s7
d13	Delay	Road surface lidar data transfer without specification and storage	s8	c6
d15	Delay	No standardized road surface lidar data storage	s8	c5
d16	Delay	Road surface lidar data format normalization	s8	s9
d22	Delay	Road structure video data format normalization	s10	s11
d21	Delay	Road structure video data storage without specification	s10	c7
d19	Delay	Road structure video data transfer without specification and storage	s10	c8
d28	Delay	Road structure millimeter wave radar data format normalization	s13	s14
d25	Delay	Mm-wave radar data storage without specification of road structure	s13	c9
d27	Delay	Mm-wave radar data transfer of road structures without specification and storage	s13	c10
d34	Delay	Road structure lidar data format normalization	s15	s16
d31	Delay	Road structure lidar data storage without specification	s15	c11
d33	Delay	Road structure lidar data transfer without specification and storage	s15	c12
d2	Delay	Road surface video data transfer without storage	s5	c2
d5	Delay	Standardized road surface video data storage	s5	c1
d6	Delay	All stored road surface video data transfer	c1	c2
d8	Delay	Road surface millimeter microwave radar data transfer without storage	s7	c4
d11	Delay	Standardized road surface millimeter microwave radar data storage	s7	c3
d12	Delay	All stored road surface lidar data transfer	c3	c4
d14	Delay	Road surface lidar data transfer without storage	s9	c6
d17	Delay	Standardized road surface lidar data storage	s9	c5
d18	Delay	All stored road surface lidar data transfer	c5	c6
d20	Delay	Road structure video data transfer without storage	s11	c8
d23	Delay	Standardized road structure video data storage	s11	c7
d24	Delay	All stored road structure video data transfer	c7	c8
d26	Delay	Road structure millimeter microwave radar data transfer without storage	s14	c10
d29	Delay	Standardized road structure millimeter microwave radar data storage	s14	c9
d30	Delay	All stored road structure lidar data transfer	c9	c10
d32	Delay	Road structure lidar data transfer without storage	s16	c12
d35	Delay	Standardized road structure lidar data storage	s16	c11
d36	Delay	All stored road structure lidar data transfer	c11	c12
d37	Delay	Multi-device video data processing	c13	s17
d38	Delay	Multi-device millimeter wave radar data processing	c14	s17
d39	Delay	Multi-device lidar data processing	c15	s17

Table 5: Table of simulation module (combine).

Module ID	Module role	Module name	Upstream modules	Downstream modules
c1	Combine	Collect all stored road surface video data	d3, d5	d6
c2	Combine	All road surface video data collection	d1, d2, d6	c13
c3	Combine	Collect all stored road surface millimeter microwave radar data	d9, d11	d12
c4	Combine	All road surface millimeter microwave radar data collection	d7, d8, d12	c14
c5	Combine	Collect all stored road surface lidar data	d15, d17	d18
c6	Combine	All road surface lidar data collection	d13, d14, d18	c15
c7	Combine	Collect all stored road structure video data	d21, d23	d24
c8	Combine	All road structure video data collection	d19, d20, d24	c13
c9	Combine	Collect all stored road structure millimeter microwave radar data	d25, d29	d30
c10	Combine	All road structure millimeter microwave radar data collection	d26, d27, d30	c14
c11	Combine	Collect all stored road structure lidar data	d31, d35	d36
c12	Combine	All road structure lidar data collection	d32, d33, d36	c15
c13	Combine	video data fusion	c2, c8	d37
c14	Combine	Millimeter wave radar data fusion	c4, c10	d38
c15	Combine	Lidar data fusion	c6, c12	d39

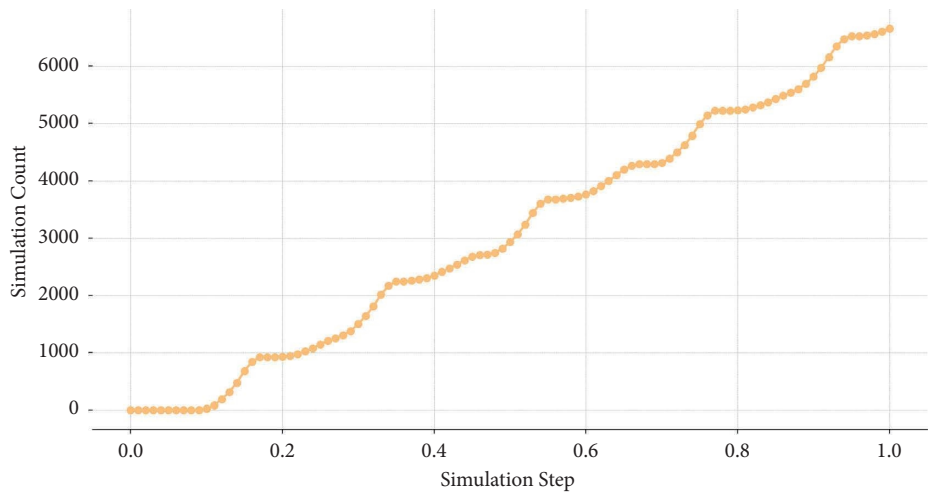


Figure 8: Number of entities having completed the process as a function of simulation time.

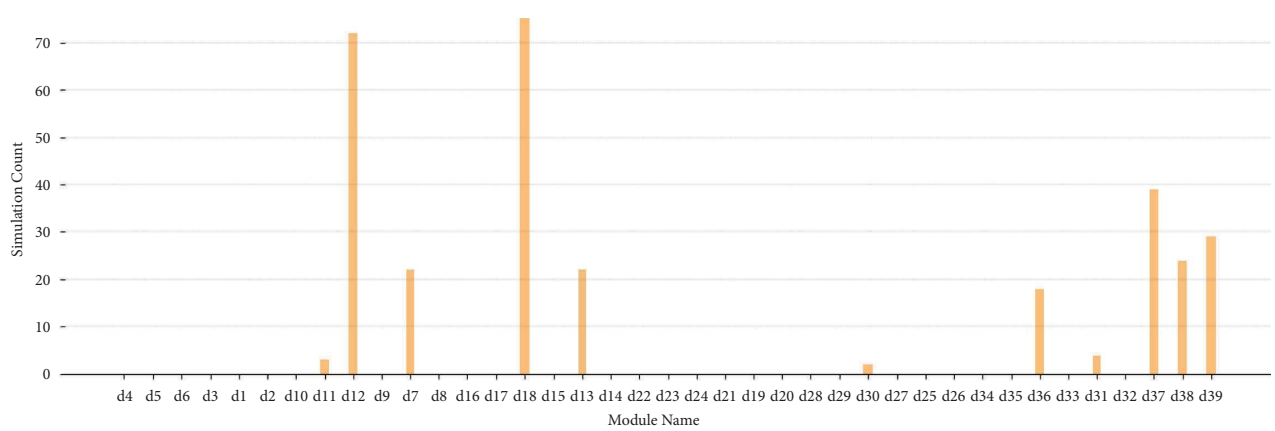


Figure 9: Number of entities stored in each module after simulation.

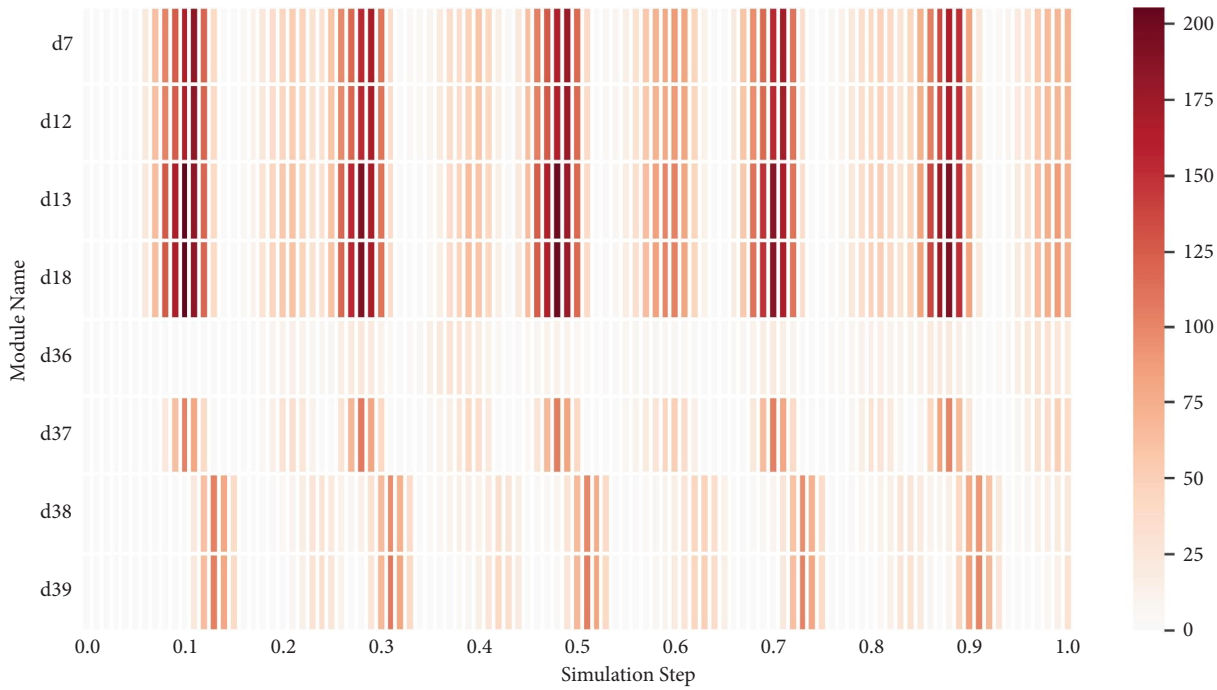


Figure 10: Timing heatmap of the number of entities stored in each module.

6. Conclusions

The above chapter introduces the application of discrete event simulation in the modeling of the ATS information interaction surface through application examples. In essence, this work is a supplement and extension of ATS architecture research and effectively solves the problem that most current simulation studies on ATS architecture modeling do not pay attention to the operating mechanism of IEP. There are several advantages to establish a discrete event simulation system for IEP:

(1) *Grasp the overall changes*: Researchers can obtain the changes of each part of the target object with the simulation time during the simulation process.

(2) *Quantitative impact*: Different statistical indicators are set according to the information flow changes in the simulation. When the architecture design or related technologies used in the architecture change, it is convenient for researchers to directly evaluate the degree of change according to the statistical situation of the indicators.

(3) *Dynamic adjustment*: Researchers can adjust the architecture according to the simulation results. After the adjustment, the architecture can be simulated again, and the adjustment can be evaluated objectively by comparing the simulation results before and after the adjustment.

Research on the CRSL between its entities can assist relevant researchers in more thoroughly analyzing the running mechanism, which is driven by both internal demands and external technologies of the system. The relationship between various entities may well be efficiently described using the DES for IEPs in ATS suggested in this research. Researchers could use this to enhance the architecture's pertinent design and do more research on the diagnosis, evaluation, verification, and optimization.

Many different industries can benefit from DES. The simulation modeling technique for IEPs in the ATS architecture described in this work, namely DESIEP, can also be used to examine other structures with the same characteristics, such as the process of Internet information transmission. Similarly, DES can be applied to study information-interactive system architecture. Nonetheless, it should be noted that modeling the design of a complicated system will become increasingly challenging.

IEP simulation and architecture simulation could be combined in future research. Through the simulation calculation of IEPs, the volume of IS between components in the architecture will be obtained, and the architecture simulation might benefit from receiving the feedback information to improve its accuracy.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

The research is supported by the National Key R&D Program of China (2020YFB1600400).

References

[1] L. J. Yue, F. Wang, and C. X. Zhao, "Cooperative application and command and control assumption of unmanned combat system in sea battlefield," *Command Control and Simulation*, vol. 44, no. 4, pp. 1–17, 2022.

[2] R. M. Gan, Z. K. Jia, Y. Y. Wang, J. Li, and Z. Xiao, "Optimization of network transmission link in UAV assisted vehicle network," *Journal of Terahertz Science and Electronic Information Technology*, vol. 20, no. 7, pp. 673–677, 2022.

[3] Z. Xue, B. Ai, G. Y. Ma, Y. Y. Ma, and G. Q. Li, "A novel multiple access approach for 5G-R massive IoT scenario," *Journal of the China Railway Society*, vol. 44, no. 2, pp. 56–63, 2022.

[4] Z. Chen, Y. Lin, L. Wu, S. Cheng, and P. Lin, "Development of a capacitor charging based quick I-V curve tracer with automatic parameter extraction for photovoltaic arrays," *Energy Conversion and Management*, vol. 226, Article ID 113521, 2020.

[5] United States Department of Transportation, "Architecture reference for cooperative and intelligent transportation," 2022, https://www.arc-it.net/.

[6] Its Canada, "Intelligent transportation system Canada," 2022, https://www.itscanada.ca/index.html.

[7] F. Zhang, "The current situation and development thinking of the intelligent transportation system in China," in *Proceedings of the 2010 International Conference on Mechanic Automation and Control Engineering*, pp. 2826–2829, IEEE, Wuhan, China, June 2010.

[8] J. Michal and Z. Moler, "Modular framework for simulation modelling of interaction-rich transport systems," in *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 2152–2159, IEEE, Indianapolis, IN, USA, October 2013.

[9] R. Salazar-Cabrera and A. Pachon, "Methodology for design of an intelligent transport system (ITS) architecture for intermediate colombian city," *Ingeniería y competitividad*, vol. 21, no. 1, pp. 49–62, 2019.

[10] L. You, J. He, W. Wang, and M. Cai, "Autonomous transportation systems and services enabled by the next-generation network," *IEEE Network*, vol. 36, no. 3, pp. 66–72, 2022.

[11] X. Shen, Y. Lu, Y. Zhang, X. Liu, and L. Zhang, "An innovative data integrity verification scheme in the Internet of things assisted information exchange in transportation systems," *Cluster Computing*, vol. 25, no. 3, pp. 1791–1803, 2022.

[12] J. Gao, G. Manogaran, T. N. Nguyen, S. Kadry, C. H. Hsu, and P. M. Kumar, "A vehicle-consensus information exchange scheme for traffic management in vehicular ad-hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 19602–19612, 2022.

[13] S. Jacobsson, P. O. Arnäs, and G. Stefansson, "Automatic information exchange between interoperable information systems: potential improvement of access management in a seaport terminal," *Research in Transportation Business and Management*, vol. 35, Article ID 100429, 2020.

[14] W. ShangGuan, J. Yu, B. Cai, and J. Wang, "Research on unsigned intersection control method based on cooperative vehicle infrastructure system," in *Proceedings of the 2017 Chinese Automation Congress (CAC)*, pp. 6436–6441, IEEE, Hangzhou, China, November 2019.

[15] Y. Li, Y. Song, T. Zheng, and H. Feng, "TransModeler based implementation of autonomous vehicular platoon control," in *Proceedings of the 2017 29th Chinese Control and Decision Conference (CCDC)*, pp. 4087–4092, IEEE, Chongqing, China, May 2017.

[16] M. Tahmidul Haq, F. Ahmed, and K. Ksaibati, "Estimating passing sight distances for overtaking truck platoons–Calibration and validation using VISSIM," *International Journal of Transportation Science and Technology*, vol. 11, no. 2, 2021.

[17] M. İ. Akbaş, C. A. Long, S. K. Hanumanthu, E. Anderson, and R. Razdan, "FpolyOS: A simulation platform to explore breakthrough concepts in intelligent transportation," in *Proceedings of the 2019 SoutheastCon*, pp. 1–6, IEEE, Huntsville, AL, USA, April 2019.

[18] X. Koutsoukos, G. Karsai, A. Laszka et al., "SURE: a modeling and simulation integration platform for evaluation of secure and resilient cyber–physical systems," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 93–112, 2018.

[19] M. Aramrattana, T. Larsson, J. Jansson, and A. Nåbo, "A simulation framework for cooperative intelligent transport systems testing and evaluation," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 61, pp. 268–280, 2019.

[20] C. Kogler and P. Rauch, "Discrete event simulation of multimodal and unimodal transportation in the wood supply chain: a literature review," *Silva Fennica*, vol. 52, no. 4, 2018.

[21] J. Z. Felde, C. Alias, and G. Alexander, "Comparing generic and dedicated tools of discrete-event simulation for examining inland waterway transportation services," in *Proceedings of the 2022 IEEE 6th International Conference on Logistics Operations Management (GOL)*, pp. 1–10, IEEE, Strasbourg, France, June 2022.

[22] M. T. Sebastiani, R. Lüders, and K. V. O. Fonseca, "Evaluating electric bus operation for a real-world BRT public transportation using simulation optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2777–2786, 2016.

[23] L. Wang, R. Iida, and A. M. Wyglinski, "Vehicular network simulation environment via discrete event system modeling," *IEEE Access*, vol. 7, pp. 87246–87264, 2019.

[24] S. Treber and G. Lanza, "Transparency in global production networks: improving disruption management by increased information exchange," *Procedia CIRP*, vol. 72, pp. 898–903, 2018.

[25] H. Yang and H. Rakha, "Feedback control speed harmonization algorithm: methodology and preliminary testing," *Transportation Research Part C: Emerging Technologies*, vol. 81, pp. 209–226, 2017.

[26] N. E. Friedkin, A. V. Proskurnikov, R. Tempo, and S. E. Parsegov, "Network science on belief system dynamics under logic constraints," *Science*, vol. 354, pp. 321–326, 2016.

[27] B. Kanti Bala, F. Mohamed Arshad, and K. M. Noh, "System dynamics," *Modeling and Simulation*, p. 274, Springer, Berlin, Germany, 2017.

[28] D. J. van der Zee, "Model simplification in manufacturing simulation–Review and framework," *Computers & Industrial Engineering*, vol. 127, pp. 1056–1067, 2019.

[29] S. Robinson, *Simulation: The Practice of Model Development and Use*, Bloomsbury Publishing, London, UK, 2014.

[30] M. Morrison, *Reconstructing Reality: Models, Mathematics, and Simulations*, University of Oxford, Oxford, UK, 2015.

[31] B.-X. Dong, M. Shan, and B.-G. Hwang, "Simulation of transportation infrastructures resilience: a comprehensive review," *Environmental Science and Pollution Research*, vol. 29, no. 9, pp. 12965–12983, 2022.

[32] S. Hao, L. Yang, and Y. Shi, "Data-driven car-following model based on rough set theory," *IET Intelligent Transport Systems*, vol. 12, no. 1, pp. 49–57, 2018.

[33] X. Ma, H. Yu, Y. Wang, and Y. Wang, "Large-scale transportation network congestion evolution prediction using deep learning theory," *PLoS One*, vol. 10, no. 3, Article ID 119044, 2015.

[34] A. Mukherjee, D. K. Jain, P. Goswami, Q. Xin, L. Yang, and J. J. Rodrigues, "Back propagation neural network based cluster head identification in MIMO sensor networks for intelligent transportation systems," *IEEE Access*, vol. 8, pp. 28524–28532, 2020.

[35] Z. Lv, R. Lou, and A. K. Singh, "AI empowered communication systems for intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4579–4587, 2021.

[36] A. D. Christian and W. P. Seering, "A model of information exchange in the design process," in *Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 17179, pp. 323–328, American Society of Mechanical Engineers, St. Louis, MO, USA, August 1995.

[37] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, Springer US, New York, NY, USA, 2008.

[38] K. Wang and X. Han, "Real-time vehicle detection based on optimized You only look once V2 algorithm," *Journal of University of Jinan (Science and Technology)*, vol. 34, no. 5, pp. 443–449, 2020.

[39] Z. Liu, Y. Zhu, and H. Wang, "Multi-target real-time detection based on convolutional neural network," *Computer Engineering and Design*, vol. 40, no. 4, pp. 1085–1090, 2019.

[40] J. Yang, X. Wang, and H. Lin, "Review of one-stage vehicle detection algorithms based on deep learning," *Computer Engineering and Applications*, vol. 58, no. 7, pp. 56–67, 2022.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 8707205, 15 pages
https://doi.org/10.1155/2023/8707205

Research Article

Mapping Relationship Discovery of Multidimensional Architectures in Autonomous Transportation System Based on Text-Matching Model

You Liu, Haonan Tuo, Mingfan He, Qiang Fu, Tianjian Yu, and Jinjun Tang

Smart Transport Key Laboratory of Hunan Province, School of Traffic and Transportation Engineering, Central South University, Changsha 410075, China

Correspondence should be addressed to Tianjian Yu; yutianjian@csu.edu.cn and Jinjun Tang; jinjuntang@csu.edu.cn

Received 20 June 2022; Revised 27 July 2022; Accepted 6 April 2023; Published 30 April 2023

Academic Editor: Linlin You

Mapping relationships of multidimensional architectures play an essential role in the autonomous transportation system (ATS), which can help understand the complex relationships between multidimensional architectures. The current mapping relationship discovery for multidimensional architectures in ATS requires significant manual involvement, leading to the underutilization of textual data and the intense subjectivity of results. In order to address the above issues, it is necessary to mine and further utilize the semantic information in the textual data. This study applies the text-matching model to the mapping relationship discovery of multi-dimensional architectures, which can calculate the semantic similarity between texts. On this basis, a method based on the Siamese-BERT-wwm-Bi-LSTM model is proposed, which incorporates Chinese BERT with whole word masking (BERT-wwm), bidirectional long-short term memory (Bi-LSTM), and the Siamese Network. A series of experiments are conducted with different text-matching models. The results show that the precision rate, recall rate, and F1-score exceed 80% for most applied methods, which verifies the feasibility of using the text-matching model for mapping relationship discovery. These results expect to provide a method with good performance that can automatically perform mapping relationship discovery.

1. Introduction

With the development of emerging technologies, such as artificial intelligence and 5G communication, the ability of transportation systems to self-organize operating and autonomous serving is rapidly improving [1–3]. Over time, the intelligent transportation system (ITS) is gradually transforming into a proactive Autonomous Transportation System (ATS), which is a new generation of systems with less human intervention and greater autonomy [4–6]. Compared to ITS supported by packet-switched networks, ATS supported by emerging technologies can automatically and intelligently manage mobility demand and supply through a self-actuating cycle consisting of sensing, learning, rearranging, and reacting steps [7]. For the new transportation system, a matching transportation system framework needs to be proposed. China, the United States, and Europe have recently conducted many studies on the transportation system framework. However, these studies generally focus on urban transportation system services, resulting in weak adaptability to emerging technologies. Therefore, there is a need to construct ATS architecture foundation theories capable of studying road traffic and integrated transportation in depth [8–13]. The multidimensional architectures in ATS consist of functional, logical, and physical architectures, which describe the system components from the corresponding perspectives. The mapping relationships of multidimensional architectures reveal the matching relationship between the constituent elements of different architectures. The in-depth elaboration of mapping relationships is one of the essential aspects of the theoretical study for ATS architectures, which helps to provide

academic support for the complex relationship exploration and the generational evolution study of multidimensional architectures.

While constructing the multidimensional architectures in ATS, textual data such as function sets, data dictionaries, physical entity object tables, and knowledge bases in different service scenarios are formed. These textual data contain constituent elements belonging to different architectures and information flow in various service scenarios. The textual data provides essential references for the mapping relationship discovery of multidimensional architectures [14]. Methods based on the fusion of horizontal and vertical and the fuzzy theory have already been applied to study ATS architectures. These two approaches alleviate the intense subjectivity and underutilization of textual data by establishing specific mapping relationship construction principles [15–18]. However, the methods above still suffer from the high requirement for manual involvement and the difficulty of directly utilizing textual data.

The text-matching model-based method is a prominent approach to studying the mapping relationship through big data mining and textual information extraction. The method determines whether two texts have a matching relationship based on constructing a model to extract their linguistic features. Several text-matching approaches, such as word statistical information-based, grammatical structure analysis-based, and semantic information-based, have been applied to explore the mapping relationship. However, the application of the first two methods is often limited by their shortcomings. Precisely, the method based on word statistical information can only extract shallow features such as word frequency while ignoring deep features such as the text structure and semantic information [19–21]. The methods of grammatical structure analysis require plenty of textual data with part-of-speech annotations, which have terrible performance in dealing with the domain text with many specialized words. In contrast, methods based on semantic information have recently been widely used in text matching. These methods can mine the deep semantic information of the textual data and support the text-matching decision by introducing the deep learning model based on the word embedding models.

The neural-network models, such as convolution neural network (CNN), recurrent neural network (RNN), and long-short term memory (LSTM), have been widely applied for deep learning. CNN can efficiently extract local features, the linguistic features at different granularities, of the textual data [22]. Yin et al. [23] introduced the attention mechanism to CNN and developed an attention-based CNN (ABCNN) for text matching. However, CNN performs poorly in extracting serialization features and long-distance dependencies from the textual data. Conversely, RNN can better extract serialization features by retaining historical information, but the vanishing and exploding gradient problem of RNN lead to learning only short-term information in the network [24]. LSTM has solved the problem of learning long-term information on account of the application of the gating mechanism. Generally, LSTM includes two main approaches: one-way LSTM and bidirectional long-short term memory (Bi-LSTM). One-way LSTM tends to neglect the beginning part of the sequences while processing long sequences [25]. Moreover, it cannot learn the information from subsequent units, which makes it challenging to handle specific problems that require the fusion of backward and forward information. In contrast, Bi-LSTM is more capable of processing textual data with an emphasis on order, and it can learn the bidirectional information by combining the forward LSTM and backward LSTM. Palangi et al. [26] introduced Siamese-Bi-LSTM for text-matching based on combining Siamese Network and Bi-LSTM. Chen et al. [27] presented the text-matching model, Enhanced LSTM for natural language inference (ESIM), by integrating the attention mechanism and Bi-LSTM.

Previous findings illustrated that the word embedding model is the other meaningful part of the text-matching model based on semantic information. It can represent text in a low-dimensional space by converting the words into dense numeric vectors that the computer can process. Mikolov et al. [28] presented the word representation model, Word2vec, by applying the continuous bag-of-words model and the continuous skip-gram model. Pennington et al. [29] created a global word representation model, GloVe, based on the co-occurrence matrix of words from the corpus. Both Word2vec and GloVe are static word embedding models, which are disabled to solve the problems of semantic nesting and polysemy. The extractor for deep bidirectional text features, bidirectional representation from transformers (BERT), solved the problem remarkably [30]. With the masked language model (MLM) and next sentence prediction (NSP) as pretraining tasks, BERT can extract deep semantic information from textual data. Moreover, by adjusting the details of BERT, it can generate a series of derived models. Liu et al. [31] discovered a robustly optimized BERT pretraining approach (Roberta), which uses a dynamic masking mechanism and removes the next sentence prediction mission. Lan et al. [32] presented a lite BERT (ALBERT) by reducing the quantities of parameters. Cui et al. [33] recommended the Chinese BERT with whole word masking (BERT-wwm), which was more suitable for processing Chinese textual data. Reimers and Gurevych [34] put forward the sentence-BERT with pretraining in Siamese network. Compared to BERT, Sentence-BERT can represent sentences better as embedding vectors, making it feasible to evaluate the text match based on the cosine similarity of sentence embedding vectors. In this way, Sentence-BERT can solve the problem of taking too many computer sources and the slow prediction of applying BERT for text-matching. It should be noted that the outputs of BERT and its derivative models are complex and need dimensionality reduction. The common pooling methods for dimensional reduction, such as [CLS] pooling and mean pooling, tend to result in partial information loss. Dong et al. [35] suggested combining BERT and CNN, extracting more detailed semantic information. Agrawal et al. [36] replaced pooling operations with Bi-LSTM, representing sentences better in vector space. Choi et al. [37] and Viji and Revathy [38] applied the Siamese Network for text-matching based on BERT-CNN and BERT-Bi-LSTM, achieving high matching accuracy.

This paper aims to develop a hybrid method, Siamese-BERT-wwm-Bi-LSTM, simultaneously considering engineering efficiency and matching accuracy. Siamese-BERT-wwm-Bi-LSTM takes the Siamese Network as the basic structure in which the combination of the superior Chinese pretraining model BERT-wwm and Bi-LSTM can accurately calculate the semantic similarity of text. The proposed method can perform mapping relationship discovery automatically by learning the semantic information in textual data, which alleviates the underutilization of textual data, low efficiency, and high subjectivity of the existing methods.

The remainder of this paper is organized as follows. Section 2 introduces the framework and methodology of this study. In Section 3, data sources and experiment details are presented. In Section 4, we discussed the performance of various models by analyzing the results. In Section 5, we summarize the conclusion of this paper.

2. Methodology

The multidimensional architectures of ATS consist of functional, logical, and physical architectures, whose essence is to describe the composition of ATS from three perspectives. The mapping relationships illustrate matching relationships between constituent elements from distinct architectures. Figure 1 shows an example of the mapping relationship of elements for monitoring traffic events. In different architectures, the same traffic scenario will be parsed into the various elements of the different architectures. In the functional architecture, monitoring traffic events is resolved into functional elements, such as collecting traffic information, generating traffic event information, and posting traffic event information. This same scene is decomposed into logical elements in the logical architecture, such as network information, traffic status information, and location and traffic events. Physical elements, such as roadside devices, network management platforms, and traffic operations managers, achieved monitoring traffic events in the physical architecture. Among them, collecting traffic information is implemented in reality by passing the location of traffic events from roadside devices. These three elements from different architectures form a complete data flow, as shown in Table 1. The architecture elements involved in the same data flow are identified as having mapping relationships with each other. The mapping relationship is shown in Table 2, where the label "1" means a mapping relationship between the front two elements. In the previous studies, the mapping relationship discovery of architecture elements was achieved by manual analysis of elements. However, manual analysis is always with the problem of inefficiency and underutilization of textual data. For the issues above, this study applies a method based on semantic information to discover mapping relationships automatically. The name of elements contains rich information, and the names of elements with mapping relationships have similar semantic information. It can be determined if there is a mapping relationship between elements by analyzing their semantic similarities. Additionally, the development of NLP brings the ability to mine and extract the reliable semantic information of elements. Therefore, it is a feasible way to discover mapping relationships by text-matching based on semantic analysis.

This study proposes a text-matching model-based method, Siamese-BERT-wwm-Bi-LSTM, for discovering mapping relationships. The process adopts the Siamese Network as the basic framework. Based on the framework, the technique combines the superior Chinese pretraining models BERT-wwm and Bi-LSTM. Figure 2 displays the framework of Siamese-BERT-wwm-Bi-LSTM. Firstly, convert the multi-dimensional architecture data into the required mapping relationship data in Table 2. Then, input the mapping relationship data into BERT-wwm. The BERT-wwm encodes the data set into embedding vectors. Afterwards, the generated vectors are used as the input of Bi-LSTM. Bi-LSTM can extract the bidirectional features of each input data and combine the forward and reverse LSTM to get a complete hidden state sequence as the element embedding vectors. Finally, determine whether there is a mapping relationship between elements by calculating the cosine similarity of the corresponding embedding vectors. If the cosine similarity between the two is greater than or equal to 0.5, it is believed that there is a mapping relationship between them.

2.1. Preprocessing for Textual Data. Textual data from multidimensional architecture in ATS cannot be directly used for discovering mapping relationships. After some preprocessing, these data need to convert to the form shown in Table 2 above after some preprocessing. Firstly, extract the data flow like Table 1 from the multidimensional architecture data. It is supposed that different elements involved in the same data flow have a mapping relationship. According to the principle, withdraw the mapping relationship data from different data flows. It is difficult for the model to learn the boundaries of the mapping relationship from only positive samples. Therefore, expand the data by using random negative sampling to enhance its feature. Randomly select some pairs of elements that do not have a mapping relationship with each other to expand the data. The expanded data is shown in Table 3. In Table 3, the label "1" respectively marks a mapping relationship between the front two elements, and "0" represents the opposite.

2.2. Pretraining Model: BERT-wwm. When processing textual data, the small size of the labeled example makes it hard for the model to learn enough information. Pretraining models solve the issue above. Firstly, the pretraining model forms through self-supervised learning from large-scale unlabeled data. The developed model can extract a word's semantic representation in a particular context. Secondly, finetune the model according to the specific tasks, which means tuning the parameters of the pretraining based on the labeled samples. In recent years, the approach based on pretraining models has become common in NLP.

BERT and its derivative models are the most popular pretraining models in recent years. The textual data of multidimensional architectures are in Chinese. Hence, the

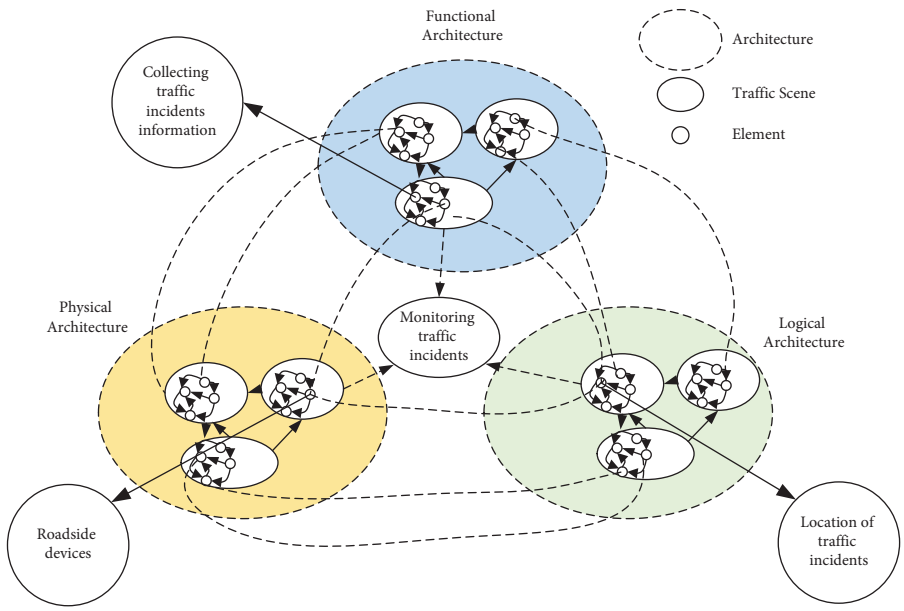


Figure 1: The mapping relationship of elements in monitoring traffic incidents.

pretraining models involved in this study are all Chinese versions. Figure 3 displays the structure of BERT-wwm, which is the same as the official Chinese BERT. The network architecture uses a multilayer transformer structure [33], which transforms the distance between two words at any position into "1" through an attention mechanism, effectively solving the thorny long-term dependency problem in NLP [39]. Figure 4 is the diagram of a transformer block corresponding to a "Trm" in Figure 3. The transformer adopts a multi-headed self-attention mechanism that enables it to learn hidden information from distinct semantic scenarios. BERT-wwm is based on the bidirectional transformer that jointly adjusts the left-to-right and right-to-left transformers. This approach effectively considers the interactivity between words. Figure 4 demonstrates the framework of the transformer block. By processing N combinations of multiheaded self-attention mechanisms and feedforward fully connected networks, the word in the sentence converts into an abstract representation of embedding vectors. The vectors generated in this way contain richer semantic information. Figure 5 shows the input for BERT-wwm. The model input forms by splicing word, segment, and position embeddings. Taking the sentence "Intelligent Transportation System is a hotspot" as an example, the sentence is shredded into separate words with [CLS] and [SEP] identified and added. [CLS] identifies the start of a sentence, and [SEP] is the symbol for separating two independent sentences.

The differences between the official Chinese BERT and BERT-wwm are mainly pretraining tasks. The pretraining tasks for BERT are MLM and NSP. For the MLM task, the training approach is to randomly replace the tokens in the input text sequence with the identifier [MASK] and then predict those masked tokens. However, BERT's original word segmentation method based on Word Piece [30] will cut a complete term into several tokens. This approach can extract only token-level semantic features without considering the traditional Chinese word segmentation. Whole word masking (WWM) will mask the other parts of the same term if a part of a complete term is masked. In this way, the term-level semantic information can be extracted. While remaining the NSP task, BERT-wwm can better deal with Chinese NLP issues by replacing the pretraining mission MML with WWM.

2.3. Bi-LSTM. According to Figure 3, the output of BERT-wwm consists of embedding vectors of [CLS], [SEP], and other tokens. Due to its high dimensionality, the output needs to be processed through pooling operations in the sentence-level task. Typically, the [CLS] pooling strategy is used for dimensionality reduction, which refers to taking the [CLS] identifier vector as the whole sentence representation [28]. Compared to CLS pooling, the model's performance can be further improved by processing the model via models such as CNN and Bi-LSTM [35–38]. Bi-LSTM, which has an excellent performance in extracting the textual sequence features, is used in this study.

Bi-LSTM is a combination of forward and backward LSTM, which is often used to model contextual information for NLP tasks. LSTM, as a variant of the RNN network, solves the problem of gradient disappearance and explosion by setting multiple gating units to achieve long-term memory. The unit control for LSTM is accomplished by an input gate, a forgetting gate, and an output gate. Figure 6 is the diagram of the structure for an LSTM unit. In Figure 6, x_t represents the input sequence. h_t is the implied layer output, i.e., the output result of each LSTM cell. f_t, i_t, O_t, and $\hat{C}_t$ are respectively the forgotten gate, input gate, output gate, and candidate memory. б and Tanh, respectively, represent the activation function of Sigmoid and hyperbolic tangent. W is the weight matrix. The symbol $\times$ stands for the dot product.

Table 1: Data flow of multidimensional architecture.

Traffic scene	Data flow	Type	Data source	Type	Data destination	Type
Monitoring traffic events	Location of traffic events	Logical	Roadside devices	Physical	Collecting traffic events information	Functional

Table 2: Mapping relationship data for monitoring traffic events.

Element p	Element q	Label
Location of traffic events	Roadside devices	1
Location of traffic events	Collecting traffic event information	1
Roadside devices	Collecting traffic event information	1

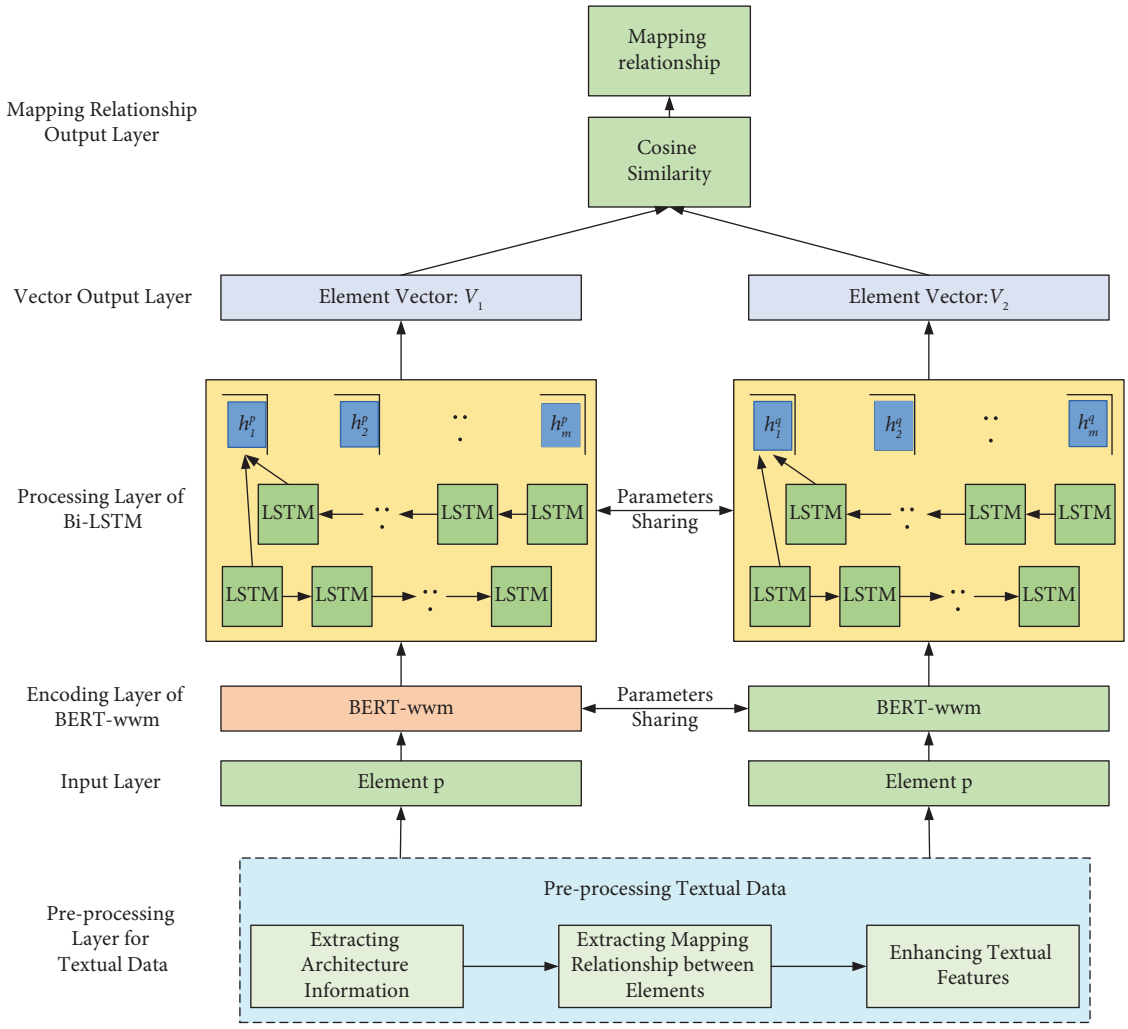


Figure 2: The framework of Siamese-BERT-wwm-Bi-LSTM.

Table 3: Expanded mapping relationship data for monitoring traffic events.

Element p	Element q	Label
Location of traffic events	Roadside devices	1
Location of traffic events	Collecting traffic event information	1
Roadside devices	Collecting traffic event information	1
Service facility information	Station management facilities	1
Service facility information	Analyzing toll station data	1
Station management facilities	Analyzing toll station data	1
Location of traffic events	Station management facilities	0
Vehicle status data	Analyzing toll station data	0

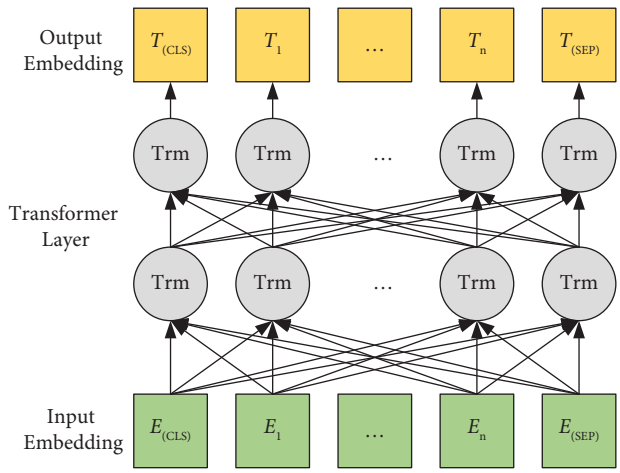


Figure 3: The structure of BERT-wwm.

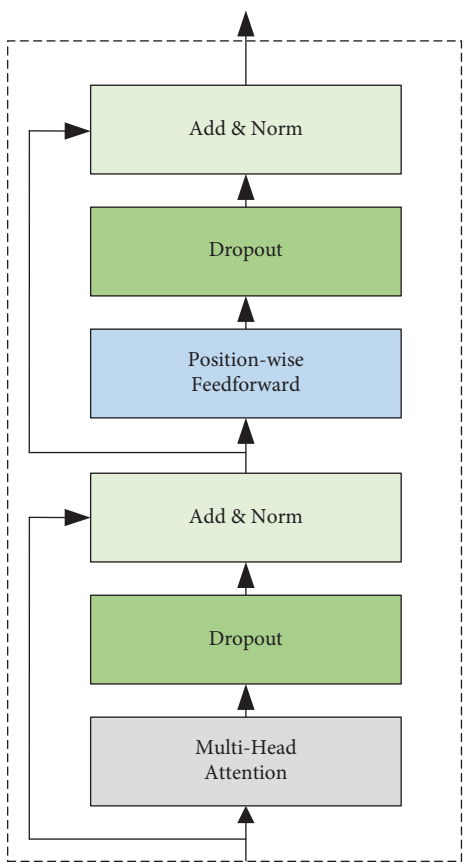


Figure 4: Transformer block.

The forgotten gate determines the degree to which the unit state at the last time is retained in the current state, which is calculated as follows:

$$f_t = \sigma\left(\mathbf{W}_f \cdot [h_{t-1}, x_t] + b_f\right), \tag{1}$$

where $\mathbf{W}_f$ and b_f are the weight matrix and the deviation of the forgotten gate.

The input gate determines the number of the input to the network saved to the unit state at the current moment. It is calculated as follows:

$$\begin{aligned} i_t &= \sigma\left(\mathbf{W}_i \cdot [h_{t-1}, x_t] + b_i\right), \\ \hat{C}_t &= \tan h\left(\mathbf{W}_c \cdot [h_{t-1}, x_t] + b_c\right), \\ C_t &= f_t \cdot C_{t-1} + i_t \cdot \hat{C}_t. \end{aligned} \tag{2}$$

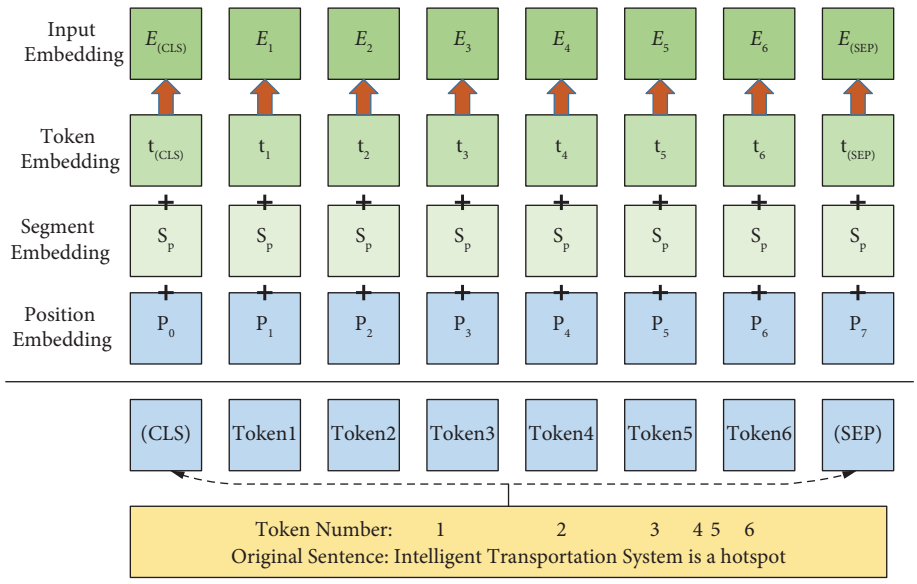


Figure 5: BERT-wwm input for the term intelligent traffic system.

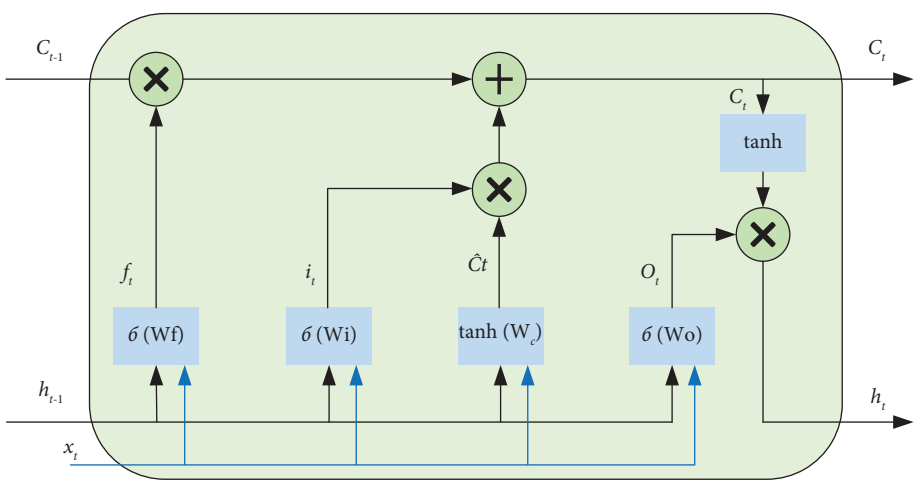


Figure 6: The structure of the LSTM unit.

$\mathbf{W}_i$ and $\mathbf{W}_c$ are the weight matrix of the input gate and the candidate memory. b_i and b_c represent the deviation of the input gate and the candidate memory. $\hat{C}_t$ is the candidate memory, and C_t is the current unit state.

The output gates control the number of unit states saved to the LSTM output at the current moment, which is calculated as follows:

$$\begin{aligned} O_t &= \text{б}(\mathbf{W}_o \cdot [h_{t-1}, x_t] + b_o), \\ h_t &= O_t \cdot \tan h(C_t), \end{aligned} \tag{3}$$

where $\mathbf{W}_o$ and b_o are the weight matrix and the deviation of the output gate.

Bi-LSTM learns information from previous and future moments through forward and backward LSTM. Figure 7 shows the structure of Bi-LSTM, which combines the forward and backward calculations. This approach solved the problem that only a one-way timing sequence was involved and that parameter calculations were greatly affected by the timing sequence. In the LSTM hidden layer, the forward calculated $\overrightarrow{h}$ and backward calculated h are preserved. The merger of $\overrightarrow{h}$ and h forms the final output of Bi-LSTM.

2.4. Siamese Network. The Siamese network is a coupled structure built on two neural networks with mutually shared weights. While text matching, the Siamese network is commonly used as the underlying framework. The framework takes two samples as input and outputs embedding vectors as their representation in high-dimensional space to compare the similarity of the two samples. ABCNN, Siamese-Bi-LSTM, and ESIM are text-matching models using the Siamese network as a framework.

Currently, most text-matching methods based on BERT and its derivative models use the original framework recommended by the author of BERT. Figure 8 is the diagram of the framework. The process splices two sentences and inputs them into the model for semantic interaction, which takes up

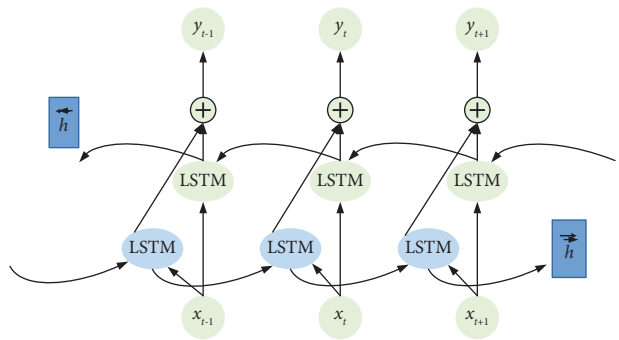


Figure 7: The structure of Bi-LSTM.

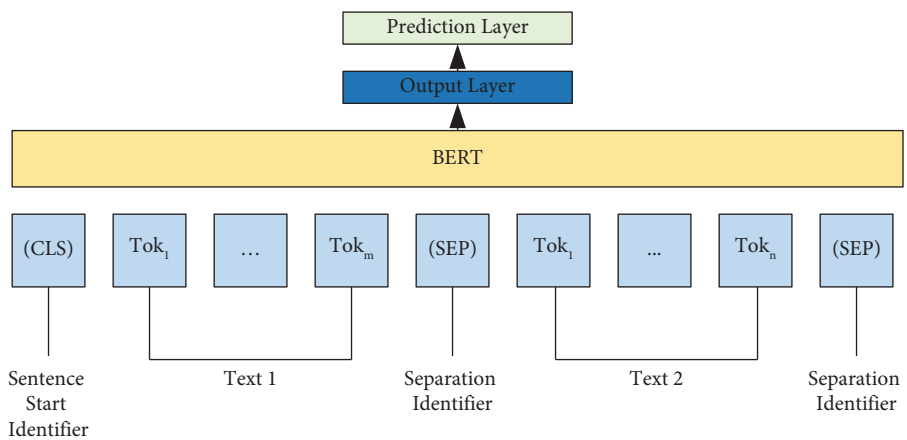


Figure 8: Text matching based on BERT and its derivative models.

a lot of computational resources. It requires a lot of time for performing semantic searches. Sentence-BERT adopts the structure of the Siamese Network, which can substantially reduce the time needed for semantic searches while ensuring high accuracy [34]. As shown in Figure 9, Sentence-BERT inputs two sentences independently, which can preserve the independent features of the sentences. The semantic interaction is implemented by similarity calculation. The model can represent textual data as embedding vectors with semantic information in this approach. Text matching can be achieved by calculating embedding vectors' cosine similarity and Manhattan distance, reducing computational resource consumption [34].

When performing mapping relationship discovery, it requires searching for pairs with mapping relationships among many architectural elements. The improvement of engineering efficiency is of great importance for this task. Therefore, in the hybrid model of this study, the Siamese network is used as the basic framework.

2.5. Siamese-BERT-wwm-Bi-LSTM. Compared with BERT and its other variant models, BERT-wwm considers the features of Chinese word segmentation. Bi-LSTM can capture more semantic information than CLS pooling. The Siamese network can improve engineering efficiency while ensuring high accuracy. This study applies a hybrid method of Siamese-BERT-wwm-Bi-LSTM for mapping relationship discovery of the multidimensional architecture, integrating the advantages of each model. Figure 2 is the diagram for the structure of Siamese-BERT-wwm-Bi-LSTM.

3. Data Source and Experiment Details

3.1. Data Set Preprocessing. Multidimensional architecture data is generated along with the construction of each architecture. However, textual data of multidimensional architecture needs preprocessing, as shown in Section 2.1, before it can be used for mapping relationship discovery. The preprocessed data consists of 5149 items with an approximate positive to negative sample ratio of 1 : 1. They are divided into a training set, a validation set, and a test set in the percentage of 6 : 2 : 2. The data format is shown in Table 3.

3.2. Experiment Environment. The experimental hardware configuration includes a 5-core Intel (R) Xeon (R) Silver 4210R CPU 64G and an Nvidia RTX 3090 GPU 24G. The operating system is Ubuntu 18.04. The development language is Python 3.8, and the development framework is Tensorflow 2.3.0 deep learning framework.

3.3. Experiment Setting. To verify the effectiveness of the proposed method in this study, the following experiments are designed:

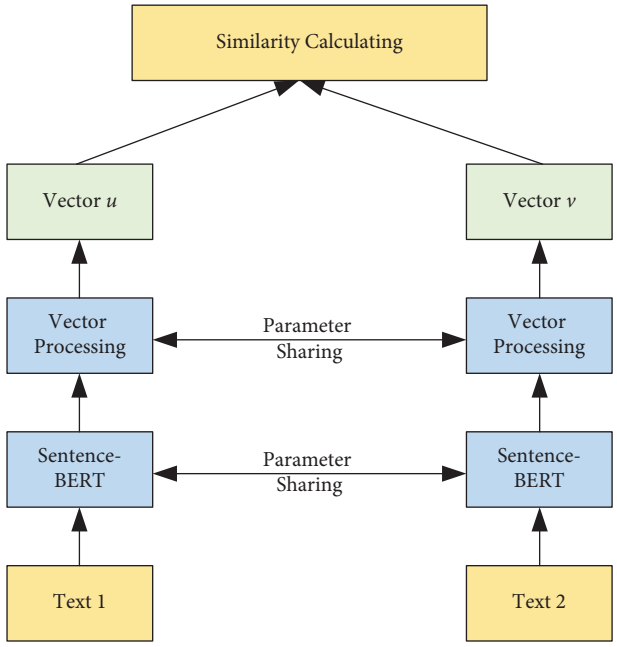


Figure 9: Text matching based on sentence-BERT.

(1) Comparison of the performance of different text-matching models for discovering mapping relationships.

(2) Ablation experiments on the hybrid model.

(3) Comparison of the performance of BERT and its derivative models based on the structure of the proposed method.

(4) Exploration experiments on the performance of the joint of CNN and Bi-LSTM.

Table 4 shows the parameters of the different pretraining models. The parameters are consistent with the original paper [30–33]. Table 5 shows the experimental parameter settings of the comparison models in this study. In Table 5, all the CNN-based models use Relu as the activation function. The size of the convolutional kernel in ABCNN is three. The rest of the CNN-based models use three kernels whose size is three, four, and five. In this way, the model outputs forms by splicing the features extracted from 3 kernels. F stands for the forward LSTM, and B represents the backward LSTM. CLS is on behalf of the strategy of CLS pooling. Bi-LSTM-CNN represents the Bi-LSTM output accessed into CNN as input. CNN-Bi-LSTM means the CNN output is accessed into Bi-LSTM as input. CNN + Bi-LSTM refers to the fact that the CNN output is spliced with Bi-LSTM output. Figures 10(a)–10(c) show the structure for Bi-LSTM-CNN, CNN-Bi-LSTM, and CNN + Bi-LSTM.

3.4. Evaluation Metrics. This study uses a text-matching model for mapping relationship discovery, transforming the problem into a binary classification. According to the session specification evaluation guidelines from the message understanding conference, apply precision rate, recall rate, and F_1-Score as the evaluation metrics. They are calculated as follows:

Table 4: Parameters of BERT and its derivative models.

Pretraining model	Number of hidden layers	Size of hidden layer	Size of output
BERT	12	768	768
ALBERT	12	768	128
Roberta	12	768	768
BERT-wwm	12	768	768

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\%,$$
$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%, \quad (4)$$
$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

TP denotes the number of positive samples predicted to be positive. FP refers to the number of negative samples predicted to be positive. FN stands for the number of positive samples predicted to be negative.

4. Experiment Results and Analysis

Table 6 shows the results of different text-matching models in the mapping relationship discovery task, where model 9 is the developed one. The developed model achieves the best performance in evaluation metrics of precision rate, recall rate, and F_1-score for discovering mapping relationships. Compared with traditional models based on the neural network, such as model 1, model 2, and model 3, the F_1-score of model 9 is remarkably higher by more than 12%. The

Table 5: Parameters of text-matching model.

Text-matching model	CNN		Number of LSTM units	Dropout rate	Optimizer	Number of epochs	Batch size
	Number of filters	Size of kernels					
ABCNN	50	3	—	0.3	Adam ($1e-3$)	50	200
Siamese-Bi-LSTM	—	—	F: 100 B: 100	0.2	Adam ($5e-4$)	100	256
ESIM	—	—	F: 256 B: 256	0.3	Adam ($5e-4$)	50	50
BERT-wwm-CLS	—	—	—	0.1	Adam ($2e-5$)	20	32
BERT-wwm-Bi-LSTM	—	—	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
BERT-wwm-CNN	128	3,4,5	—	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-wwm-CLS	—	—	—	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-wwm-CNN	128	3,4,5	—	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-wwm-Bi-LSTM-CNN	128	3,4,5	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-wwm-CNN-Bi-LSTM	128	3,4,5	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-wwm-CNN + Bi-LSTM	128	3,4,5	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-Bi-LSTM	—	—	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
Siamese-ALBERT-Bi-LSTM	—	—	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
Siamese-roberta-Bi-LSTM	—	—	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32
Siamese-BERT-wwm-Bi-LSTM	—	—	F: 100 B: 100	0.1	Adam ($2e-5$)	20	32

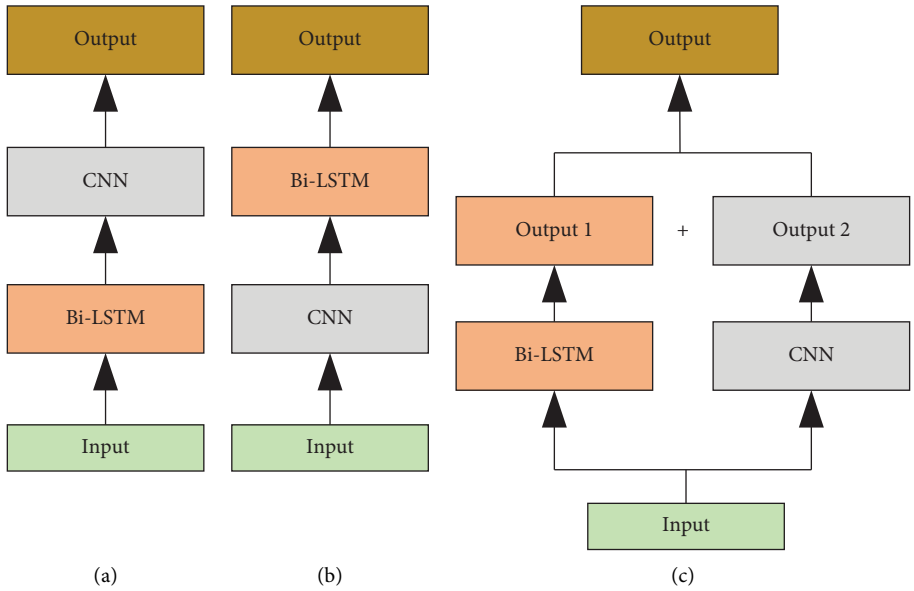


Figure 10: The joint use of Bi-LSTM and CNN.

results show that the pretraining model can effectively represent the textual data. Model 4, model 5, and model 6 are BERT-wwm-based models with the original framework, as shown in Figure 8. Model 9 has an excellent F_1-score, higher than models 4, 5, and 6 by 2.69%, 4.06%, and 3.63%. It is easy to conclude that the Siamese Network can help improve the

Table 6: The performance of text-matching models for discovering mapping relationship.

No	Text-matching model	Precision (%)	Recall (%)	F_1 (%)
1	ABCNN	66.01	79.81	72.26
2	Siamese-Bi-LSTM	61.31	82.42	70.31
3	ESIM	71.13	79.57	75.11
4	BERT-wwm-CLS	84.84	85.42	85.13
5	BERT-wwm-Bi-LSTM	83.49	84.03	83.76
6	BERT-wwm-CNN	83.84	84.55	84.19
7	Siamese-BERT-wwm-CLS	85.45	85.91	85.68
8	Siamese-BERT-wwm-CNN	86.65	86.18	86.41
9	Siamese-BERT-wwm-Bi-LSTM	**87.30**	**88.35**	**87.82**

Bold values refer to the maximum prediction performance in each circumstance.

Table 7: The results of the ablation experiments.

No	Text-matching model	Precision (%)	Recall (%)	F_1 (%)
1	Siamese-Bi-LSTM	62.64	80.05	70.28
2	BERT-wwm-CLS	84.84	85.42	85.13
3	BERT-wwm-Bi-LSTM	83.49	84.03	83.76
4	Siamese-BERT-wwm-CLS	85.45	85.91	85.68
5	Siamese-BERT-wwm-Bi-LSTM	**87.30**	**88.35**	**87.82**

Bold values refer to the maximum prediction performance in each circumstance.

Table 8: The performance of different pretraining models.

No	Text-matching model	Precision (%)	Recall (%)	F_1 (%)
1	Siamese-BERT-Bi-LSTM	87.03	87.26	87.15
2	Siamese-ALBERT-Bi-LSTM	87.28	86.18	86.73
3	Siamese-roberta-Bi-LSTM	87.02	86.72	86.87
4	Siamese-BERT-wwm-Bi-LSTM	**87.30**	**88.35**	**87.82**

Bold values refer to the maximum prediction performance in each circumstance.

performance of models. Model 7, model 8, and model 9, respectively, apply the strategies of CLS pooling, CNN, and Bi-LSTM for dimensionality duction. Their performance indicates that the strategy of processing through neural networks can help mine more detailed semantic information than CLS pooling. At the same time, Bi-LSTM shows superiority over CNN in semantics mining.

Table 7 displays the results of the ablation experiments, where model 5 is the proposed one. By comparing the results in Table 7, BERT-wwm, Bi-LSTM, and Siamese Network have improved the model performance. The metrics of model 1 and model 5 demonstrate the usefulness of the pretraining model in the mapping relationship discovery task. It can be proved that replacing CLS pooling with Bi-LSTM from the performance of model 2, model 3, model 4, and model 5. The comparison between model 4 and model 5 indicates the Siamese network is positive for the performance of models. While setting the number of epochs as 20, model 3 takes 7726 s to train based on the training set and 774 s to predict based on the validation set. With the same number of epochs, model 5 takes 3928 s to train based on the training set and 75 s to predict based on the validation set. The differences between the time lengths for training and prediction indicate that the Siamese network can effectively reduce the consumption of computing resources. In general, the various modules of the hybrid model are valid. And the Siamese network can improve engineering efficiency.

Table 8 shows the results of replacing BERT-wwm with other pretraining models in the introduced framework. The presented framework is shown in Figure 2. The pretraining models involved are BERT, Albert, and Roberta. The abovementioned models are all pretraining based on the Chinese Wikipedia Corpus. According to the metrics in Table 8, BERT-wwm shows superiority over other pretraining models. The results indicate that the mission of WWM can effectively improve the performance of pretraining models in dealing with Chinese NLP problems.

The performance of the models in Table 8 varies widely in the value of the precision rate and narrowly in the value of the recall rate. It suggests that BERT-wwm can help identify more mapping relationships while ensuring the same precision rate.

In some NLP scenarios, the joint use of Bi-LSTM and CNN works better than either model alone [40–43]. No study attempts to jointly use LSTM and CNN for text-matching models based on BERT and its derivative models. Therefore, the experiments in Table 9 are conducted, where model 5 is the proposed one. Model 1 forms by replacing Bi-LSTM with Bi-LSTM-CNN in the developed framework, as shown in Figure 2. Bi-LSTM-CNN first extracts long-range features from the output of the hidden layer by Bi-LSTM and then inputs the long-range features into CNN for the extraction of local features [40]. The Bi-LSTM part of the developed framework is changed

Table 9: The performance for the combination of Bi-LSTM and CNN.

No	Text-matching model	Precision (%)	Recall (%)	F_1 (%)
1	Siamese-BERT-wwm-Bi-LSTM-CNN	86.37	85.37	85.87
2	Siamese-BERT-wwm-CNN-Bi-LSTM	80.80	81.03	80.91
3	Siamese-BERT-wwm-CNN + Bi-LSTM	86.41	86.12	86.27
4	Siamese-BERT-wwm-CNN	86.65	86.18	86.41
5	Siamese-BERT-wwm-Bi-LSTM	**87.30**	**88.35**	**87.82**

Bold values refer to the maximum prediction performance in each circumstance.

to CNN-Bi-LSTM in model 2. CNN-Bi-LSTM refers to using Bi-LSTM to get the long-range features from the extracted local features from CNN [41]. CNN + Bi-LSTM splices the long-range features from Bi-LSTM and local features from CNN as a new representation of features [42, 43].

As shown in Table 9, replacing Bi-LSTM or CNN with the combination of Bi-LSTM and CNN cannot improve mapping relationship discovery performance. On the one hand, it may be that a little semantic information extracted by BERT-wwm remains after the multilayer neural network. On the other hand, it may be that the elements are too short to contain adequate information. There is an overlap between long-range features and local features for the short text, resulting in large weights for some features and weakening the model performance.

5. Conclusions

Mapping relationship discovery for multi-dimensional architectures in ATS is essential for theoretical research. It makes sense to accurately and quickly identify the pairs of elements that have mapping relationships. This study develops a method based on Siamese-BERT-wwm-Bi-LSTM for mapping relationship discovery to alleviate the high subjectivity and underutilization of textual data. The method automates the mapping relationship discovery of multidimensional architectures. According to the experimental results, the text-matching models are feasible for the mapping relationship discovery task. By comparing the results of distinct models, Siamese-BERT-wwm-Bi-LSTM achieves superior performance in the metrics of precision rate, recall rate, and F_1-score.

In this paper, we apply various models for mapping relationship discovery. By analyzing the results of the models, it is concluded that Siamese-BERT-wwm-Bi-LSTM has a better performance. And the ablation experiments verify the validity of each module of the hybrid model. Also, we explore the applicability of the joint use of Bi-LSTM and CNN in the scenario of this paper. The combination of Bi-LSTM and CNN leads to performance degradation, and it is better to use either model alone.

The preprocessing method in this paper for textual data of the multidimensional architectures can improve. When performing text feature enhancement, a random negative sampling method is used to expand the data. However, this approach does not fully utilize the information from negative samples in practice. In future studies, we will improve the way of negative sampling. At the same time, we will try to enhance the text features through text generation.

Data Availability

The textual data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This research was funded in part by the National Key R&D Program of China (grant no. 2020YFB1600400) and Innovation-Driven Project of Central South University (grant no. 2020CX041).

References

[1] Y. Li, S. E. Li, X. Jia, S. Zeng, and Y. Wang, "FPGA accelerated model predictive control for autonomous driving," *Journal of Intelligent and Connected Vehicles*, vol. 5, no. 2, pp. 63–71, 2022.

[2] L. Yue, M. Abdel-Aty, and Z. Wang, "Effects of connected and autonomous vehicle merging behavior on mainline human-driven vehicle," *Journal of Intelligent and Connected Vehicles*, vol. 5, no. 1, pp. 36–45, 2022.

[3] T. Olovsson, T. Svensson, and J. Wu, "Future connected vehicles: communications demands, privacy and cybersecurity," *Communications in Transportation Research*, vol. 2, no. 100, Article ID 100056, 2022.

[4] N. Coretti Sanchez, I. Martinez, L. Alonso Pastor, and K. Larson, "On the simulation of shared autonomous micromobility," *Communications in Transportation Research*, vol. 2, no. 100, Article ID 100065, 2022.

[5] L. L. You, F. Zhao, L. Cheah, K. Jeong, P. C. Zegras, and M. Ben-Akiva, "A generic future mobility sensing system for travel data collection, management, fusion, and visualization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4149–4160, 2020.

[6] L. L. You, B. Tuncer, and H. Xing, "Harnessing multi-source data about public sentiments and activities for informed design," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 2, pp. 343–356, 2019.

[7] L. L. You, J. S. He, W. Wang, and M. Cai, *Autonomous Transportation Systems and Services Enabled by the Next-Generation Network*, IEEE Network, Manhattan, NY, USA, 2022.

[8] K. Zhang, H. Liu, and D. M. Liu, *Construction Method and Application of Intelligent Transportation System Framework*, China Communications Press, Beijing, China, 2013.

[9] X. J. Wang, Z. W. Zhang, and T. Y. Qi, *China Intelligent Transportation System Framework*, China Communication Press, Beijing, China, 2003.

[10] X. J. Wang, "Study on the framework of intelligent transportation system," *Computer and Communications*, vol. 19, no. 4, pp. 4–7, 2001.

[11] K. Zhang, T. Y. Qi, D. M. Liu, C. Y. Wang, R. H. He, and H. Liu, "The latest achievements of Chinese national ITS architecture," *Journal of Transportation Systems Engineering and Information Technology*, vol. 5, no. 5, pp. 6–11, 2005.

[12] Y. S. Su and S. Q. Ni, "Research on the structure of intelligent railway system in China," *Journal of Transportation Systems Engineering and Information Technology*, vol. 15, no. 4, pp. 135–140, 2017.

[13] J. Niu, "Research standards for iot architectures for intelligent transportation systems," *China Standardization*, vol. 14, no. 1, pp. 231-232, 2017.

[14] W. Wei, L. Zheng, and M. Cai, "Research on user needs of Intelligent Transportation System for autonomous transportation," *Technology and Economy in Areas of Communications*, vol. 24, pp. 1–6, 2022.

[15] Z. Wang, L. M. Jia, Y. H. Wang, Y. Qin, and X. Q. Cheng, "Structural design method of railway intelligent transportation system," *China Railway Science*, vol. 27, no. 2, pp. 116–119, 2006.

[16] J. J. Tang, Y. Liu, Q. Fu, and H. N. Tuo, "Multi-dimensional mapping relationship modeling in autonomous transportation system Based on Fuzzy Theory," *Forest engineering*, vol. 38, no. 2, pp. 146–153, 2022.

[17] X. H. He, *Study on Key Issues in the Framework Design of Inland River Shipping System*, Tongji University, Shanghai, China, 2007.

[18] J. Ma, "Research based on multi-level mapping relationship model for dynamic system based on medium granularity," *Journal of Systems Science and Mathematical Sciences*, vol. 38, no. 8, pp. 886–880, 2018.

[19] C. L. Wang, Y.. , H. Yang, and F. Deng, "A review of text similarity approaches," *Information Science*, vol. 37, no. 3, pp. 158–168, 2019.

[20] R. Maipradit, H. Hata, and K. Matsumoto, "Sentiment classification using N-gram inverse document frequency and automated machine learning," *IEEE Software*, vol. 36, no. 5, pp. 65–70, 2019.

[21] M. Ayub, M. A. Ghazanfar, and M. Maqsood, "A Jaccard base similarity measure to improve performance of CF based recommender systems," in *Proceedings of the 32nd International Conference on Information Networking(ICOIN)*, pp. 1–6, Wuhan, China, April 2018.

[22] Y. Chen, *Convolutional Neural Network for Sentence Classification*, University of Waterloo, Waterloo, UK, 2015.

[23] W. Yin, H. Schütze, B. Xiang, and B. Zhou, "Abcnn: attention-based convolutional neural network for modeling sentence pairs," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 259–272, 2016.

[24] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," 2016, https://arxiv.org/abs/1605.05101.

[25] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[26] H. Palangi, L. Deng, and Y. Shen, "Semantic modelling with long-short-term memory for information retrieval," 2014, https://arxiv.org/abs/1412.6629.

[27] Q. Chen, X. Zhu, and Z. Ling, "Enhanced LSTM for natural language inference," 2016, https://arxiv.org/abs/1609.06038.

[28] T. Mikolov, K. Chen, and G. Corrado, "Efficient estimation of word representations in vector space," 2013, https://arxiv.org/abs/1301.3781.

[29] J. Pennington, R. Socher, and C. D. Manning, "Glove: global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing(EMNLP)*, pp. 1532–1543, Doha, Qatar, October 2014.

[30] J. Devlin, M. W. Chang, and K. Lee, "Bert: pre-training of deep bidirectional transformers for language understanding," 2018, https://arxiv.org/abs/1810.04805.

[31] Y. Liu, M. Ott, and N. Goyal, "Roberta: a robustly optimized bert pretraining approach," 2019, https://arxiv.org/abs/1907.11692.

[32] Z. Lan, M. Chen, and S. Goodman, "Albert: a lite bert for self-supervised learning of language representations," 2019, https://arxiv.org/abs/1909.11942.

[33] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang, "Pre-training with whole word masking for Chinese bert," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3504–3514, 2021.

[34] N. Reimers and I. Gurevych, "Sentence-bert: sentence embeddings using siamese bert-networks," 2019, https://arxiv.org/abs/1908.10084.

[35] J. Dong, F. He, and Y. Guo, "A commodity review sentiment analysis based on BERT-CNN model," in *Proceedings of the 2020 5th International Conference on Computer and Communication Systems(ICCCS)*, pp. 143–147, Shanghai, China, May 2020.

[36] S. Agrawal, S. Dutta, and B. K. Patra, "Sentiment analysis of short informal text by tuning BERT-Bi-LSTM model," in *Proceedings of the IEEE EUROCON 2021-19th International Conference on Smart Technologies, SmartTech-IC*, pp. 98–102, Lviv, Ukraine, July 2021.

[37] H. Choi, J. Kim, and S. Joe, "Evaluation of bert and albert sentence embedding performance on downstream nlp tasks," in *Proceedings of the 2020 25th International Conference on Pattern Recognition(ICPR)*, pp. 5482–5487, Mico, Italy, May 2021.

[38] D. Viji and S. Revathy, "A hybrid approach of Weighted Fine-Tuned BERT extraction with deep Siamese Bi–LSTM model for semantic text similarity identification," *Multimedia Tools and Applications*, vol. 81, no. 5, pp. 6131–6157, 2022.

[39] C. Liu, C. Zhu, and W. Zhu, "Chinese named entity recognition based on BERT with whole word masking," in *Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence*, pp. 311–316, Tianjin, China, April 2020.

[40] J. Zhang, Y. Li, and J. Tian, "LSTM-CNN hybrid model for text classification," in *Proceedings of the 3rd Advanced Information Technology, Electronic and Automation Control Conference(IAEAC)*, pp. 1675–1680, Chongqing, China, October 2018.

[41] Y. Luan and S. Lin, "Research on text classification based on CNN and LSTM," in *Proceedings of the IEEE international conference on artificial intelligence and computer applications(ICAICA)*, pp. 352–355, Dalian, China, March 2019.

[42] N. Jin, C. J. Zhao, H. R. Wu, Y. S. Miu, H. C. Yang, and B. Z. Yang, "Agricultural short text matching technology Based on multi-semantic features," *Transactions of the Chinese Society for Agricultural*, vol. 53, no. 5, pp. 325–331, 2022.

[43] W. Li, S. Gao, and H. Zhou, "The automatic text classification method based on bert and feature union," in *Proceedings of the 2019 IEEE 25th International Conference on Parallel and Distributed Systems(ICPADS)*, pp. 774–777, Beijing, China, December 2019.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 1440763, 12 pages
https://doi.org/10.1155/2023/1440763

Research Article

Air Traffic Safety Assurance Based on Flight Plan Risk Assessment and Early Warning

Ruiwei Liu,[1] Siqi Hao ,[2] Yaping Zhang,[3] and Guan Lian[4]

[1]*Department of Naval Architecture and Marine Engineering, Guangzhou Maritime University, Guangzhou 510000, China*
[2]*Department of Ports and Shipping Management, Guangzhou Maritime University, Guangzhou 510000, China*
[3]*School of Transportation Science and Technology, Harbin Institute of Technology, Harbin 150090, China*
[4]*School of Transportation and Architecture Engineering, Guilin University of Electronic Technology, Guilin 541000, China*

Correspondence should be addressed to Siqi Hao; siqihao47@163.com

Received 14 October 2022; Revised 28 January 2023; Accepted 5 April 2023; Published 27 April 2023

Academic Editor: Linlin You

This article presents a novel risk assessment method for aircraft flight plans in a four-dimensional trajectory-based operation background. With the increasing air transport volume and airspace demand and during the allocation and modification of flight plans, safety should also be considered in addition to economic benefits and competitiveness. By employing the time geography concept, this article first quantifies the space-time accessibility and estimates the visiting probability of aircraft in the four-dimensional trajectory-based operation context. On this basis, this article studies the method and process to evaluate the conflict risk between one flight plan and a special airspace, the weather-influenced airspace, and another flight plan. This article also proposes the conflict risk alerting strategy. The achievements of the study are expected to provide assistance for air traffic controllers in strategic conflict avoidance and are of great significance in assuring air transport safety.

1. Introduction

A flight plan is the core product of an airline, and the basis of the airline company's business activities is an important basis for providing air traffic services. A reasonable flight plan helps reduce the potential conflict risk of aircraft, ensure air transport safety, and balance airspace utilization [1, 2]. Civil airspace resources are limited, and airspace resources are becoming increasingly tense with the continuous increase in air traffic demand. The competition for airspace resources between aircraft is bound to cause conflicts, which seriously affects the service quality of air transportation and causes great safety risks.

Under the current air transport management system, the formulation of flight plan is completed by the airline. The economic benefit and competitiveness of the flight plan is the primary consideration of airlines [3]. The impact of flight planning on air traffic safety has not been fully and scientifically assessed. Studies on conflict risk assessment methods for flight plans are insufficient [4].

However, the increase in air traffic demand and the demand for airspace resources bring greater challenges to air transport safety, especially under the operation concept of four-dimensional trajectory-based operations, which allow aircraft pilots to fly a relatively free trajectory as long as the waypoints to pass and the arrival time of each waypoint's requirements are met [5]. The controller and pilot workload consequently increased, and the airspace could not achieve effective use.

Therefore, this study proposes a novel conflict risk assessment method for flight plans, which aims to quantify the conflict risk of aircraft flight plans, provide a multidimensional reference for air traffic management, further adjust flight plans before flight departure, minimize conflict risks during the strategic phase, and eventually maximize air transport safety.

This article is structured as follows. After this introduction, Section 2 reviews the researches on air traffic safety assurance from two aspects: risk identification and monitoring and risk warning. Section 3 introduces the

problem to be solved by analysing the concept of four-dimensional trajectory-based operation and the challenges and opportunities that it brings to the modern air transport industry. Section 4 proposes the quantification method of the space-time accessibility of flight plans and the calculation method of the visiting probability of aircraft in the context of four-dimensional trajectory-based operation. The conflict risk assessment method between one flight plan and a special airspace, a weather-influenced airspace, and another flight plan is also addressed in Section 4. The risk assessment process is elaborated, and the alerting strategy for flight plan adjustment is outlined as well. Section 5 provides a three-aircraft scenario to verify and illustrate the proposed risk assessment method. Section 6 summarizes the main results of this article and discusses future research directions.

2. Literature Review

Since the birth of aircraft in the early 20th century, the aviation industry has begun to explore how to ensure flight safety, especially in terms of how to avoid various flight errors or mistakes and ensure the safety management of passengers, flight attendants, and ground staff. In the research results of risk identification and monitoring, many scholars have conducted in-depth research on air traffic safety from the main dimensions, including people, equipment, environment, and management, and on the objects and technologies in the risk monitoring process.

Some researchers study the aircraft conflict risk by analysing main influencing factors, especially human factor, and establish a model between these factors and the conflict risk [6, 7]. Eyferth et al. established an air traffic controller's psychological activity model and outlined the factors to which the controller paid attention when a traffic collision occurred [8]. Shyur quantified the aviation risk caused by human error by studying aviation accidents and safety indicators and introduced the baseline risk function as a quadratic curve function into aviation risk assessment to obtain a proportional hazard model. The model is used to investigate nonlinear aviation safety factors to assess safety risks [9]. Kirwan and Gibson considered that the ATM is human-centred in recent decades and studied a human reliability assessment tool for air traffic safety management [10].

Conflict risk has also been assessed by predicting and estimating aircraft conflict probability [11–13]. Pérez-Castán et al. defined conflict risk as the combination of conflict probability between aircraft pairs and estimated air traffic flow, and he proposed a new approach to determine the conflict probability [11]. Netjasov presented a framework for airspace planning and a conflict risk assessment method using the conflict probability. The presented method has been verified to be effective in both en route and terminal manoeuvring airspaces [14]. Lehouillier et al. use a geometrical approach to explore the aircraft conflict resolution problem under uncertainties caused by the wind effect, aircraft speed prediction and flight delay in the execution of manoeuvres. Monte Carlo simulation was conducted to validate the method, and the results show that it can generate 10 solutions for 35 aircraft within 3 minutes [15]. Data driven is also one of the major research directions to assess the risk. Natalia and Salvatore proposed a novel quantitative risk assessment method for a civil airport based on historical data of aircraft accidents for nearly 30 years [16]. Oztekin develop a safety assessment tool for air traffic control system, in which safety associated with an ATC facility is modelled as an influence network using a set of risk factors [17].

In general, scholars have made many research achievements on air traffic conflict risk management and mainly focused on risk identification methods, evaluation methods and monitoring technologies. In the construction of index systems and risk evaluation, fuzzy comprehensive evaluation, analytic hierarchy processes or weighted average methods are often used for comprehensive risk evaluation [13, 18–20]. Besides, the early warning idea is reflected in safety risk management in the aviation field, and attempts have been made in the aspects of early warning principles, methods, and technical support in formulating management standards and norms [21]. However, there are few studies on aviation safety early warning management to avoid conflict from the perspective of prevention. Therefore, under the 4D TBO framework in next generation air transport system, this article proposed a flight plan risk assessment method by quantifying potential conflict probability of flight plan, thus the first barrier of aviation safety can be established by early warning and strategically optimize the flight plan.

3. Research Background and Problem Description

Currently, the task of air safety assurance generally focuses on the real-time aircraft safety separation maintenance. Specifically, the ground controller monitors the real-time aircraft' positions and status, and predict their future tracks and potential conflict between aircraft. For the aircraft to involve in potential conflict, the controller gives instructions by adjusting the heading, speed, and altitude to maintain the safety separation between aircraft. In view of the uncertainty of track prediction, many studies have divided the problem of conflict detection into deterministic and probabilistic conflict detection according to the model of uncertainty; meanwhile, conflict resolution methods have also developed from imperative to automatic track planning. With the increasing flight flow, the workload of ground controllers far exceeds their rated working capacity, especially in peak hours and some high-flow and high-density airspaces. Controller are faced with tremendous work pressure, and the safety of whole aviation system also suffer certain hidden dangers.

The future air traffic management system proposes the concept of four-dimensional trajectory-based operation (4D TBO), which provides support for active aircraft safety assurance [22]. Specifically, the key to the four-dimensional trajectory-based operation is to integrate the time dimension into the trajectory, and to accurately reflect the entire flight process by accurately describing the spatial position and time of each point on the trajectory [23, 24]. Trajectory information is able to share between the aircraft, the airline, and the air traffic control department, promote the

coordinated decision-making of all departments. The whole process of ensuring the operation of the aircraft is visible, controllable, and accessible [25, 26]. By designing an effective uncongested flight trajectory for the aircraft in advance in the strategic phase and the pretactical phase, the air traffic controller's workload in the tactical phase is expected to be reduced, so that the number of aircraft that can be handled in each working hour unit can be increased, thereby increasing the airspace capacity. Besides, under the framework of the four-dimensional trajectory-based operation, the aircraft can realize the trajectory sharing between the aircraft, thus improve the pilot's situational awareness and autonomy for trajectory modification. Airspace users have higher flexibility to optimize their operation by selecting the optimal trajectory. At the same time, four-dimensional trajectory-based operation enables different safety separation criteria according to aircraft performance. Digital track management will greatly improve the automation level of air traffic control. Four-dimensional trajectory-based operation makes it possible to accurately grasp the aircraft's flight intentions, improve the predictability of the trajectory, and ensure the overall performance of the air traffic management network. Therefore, four-dimensional trajectory-based operation is an effective means to manage the airspace under the conditions of large flow, high density, and small safety separation in the future, which can significantly reduce the uncertainty of aircraft trajectory and improve the safety and utilization of airspace and airport resources [27].

At present, active safety is an important development direction of traffic system safety theory in road, railway, and maritime transportation system. Therefore, the aviation system should also focus on active conflict avoidance, which advance the air safety assurance work to the strategic stage, and adjust the flight plan with high-conflict risk to a reasonable level through the safety assessment of flight plan and early warning. In traditional air traffic control system, the flight planning phase is relatively separated from the execution phase. A flight plan containing basic information on the aircraft's flight intent was formulated and submitted several days before performing the flying mission. Economic benefit and operation efficiency are the most important objectives to achieve when designing a flight plan to maximize the airline's interests [28, 29]. Therefore, to advance the air safety assurance work to the strategic stage, it is important to propose a safety assessment method for flight plan, which is able to provide a quantitative evaluation criterion for flight plan adjustment and early warning, and provide better assurance for the aviation system.

4. Methods

4.1. Aircraft Accessibility and Visiting Probability. Since the 4D flight plan limits all passing air waypoints of aircraft and the corresponding arrival times of each waypoint, the space-time reachable area of the aircraft is limited when flying on each air route segment. The basis of flight plan risk assessment is to determine the aircraft's space-time reachable area and calculate the aircraft's visiting probability to each point inside, which are elaborated in this section.

4.1.1. Aircraft Reachable Domain Generation. Under the operation concept of four-dimensional trajectory-based operation, the range of motion of aircraft is limited to a certain space-time airspace, which is defined as the space-time accessibility of aircraft [30]. The space-time accessibility of the aircraft can reflect the pros and cons of the flight plan. A good flight plan will not give the aircraft very high accessibility, since this will increase the uncertainty during the flying process. However, it should have certain accessibility to ensure freedom for aircraft autonomous trajectory planning.

The aircraft accessibility is measured by its space-time reachable area, which can be obtained by the intersection of two cones, as illustrated in Figure 1: (1) one reverse cone with the vertex at the origin waypoint, its height is the flight duration between waypoints, and its radius is the multiplier of the flight duration and maximum flying velocity of aircraft and (2) one forward cone with the vertex at the destination waypoint and same height and radius as the reverse cone.

The first step of aircraft space-time accessibility quantification is to determine the maximum cruising velocity V_m of the aircraft according to the aircraft type, which can be found in the flight plan. Next step is to calculate the set $R_s(t)$ of all discretization units that can be reached at time t after the aircraft leaves the first waypoint s denoted by (x_s, y_s) at time t_s.

$$R_s(t) = \{(x, y)\} \mid \sqrt{(x - x_s)^2 + (y - y_s)^2} \leq (t - t_s) \times V_m. \tag{1}$$

Then, we calculated the set $R_e(t)$ of all discretization units that can reach the second waypoint e denoted by (x_e, y_e) at time t_e.

$$R_e(t) = \{(x, y)\} \mid \sqrt{(x - x_e)^2 + (y - y_e)^2} \leq (t - t_e) \times V_m, \tag{2}$$

where (x, y) represents the coordinate of the discretized unit.

The space-time reachable area $R_{es}(t)$ of the aircraft at time t can be obtained as the intersection of $R_s(t)$ and $R_e(t)$.

$$R_{es}(t) = \{R_s(t) \cap R_e(t)\}. \tag{3}$$

Details on the motivation and principle of reachable domain generation can be found in our previous work [31]. The reachable area $R_{es}(t)$ indicates the space-time accessibility of the aircraft on the air route segment r with origin waypoint s and destination waypoint e. The reachable domain $R_{es}(t)$ of the aircraft A on the air route segment r can be written as R_{Ar}. The reachable area represents the space that the aircraft can reach on the premise of meeting the requirements of passing the waypoint during flight. Computing based on maximum speed is a minimum condition, which can ensure that conflicts are avoided to the greatest extent. Further constraints, such as the over point speed limits, may narrow the reachable space and reduce the conflict

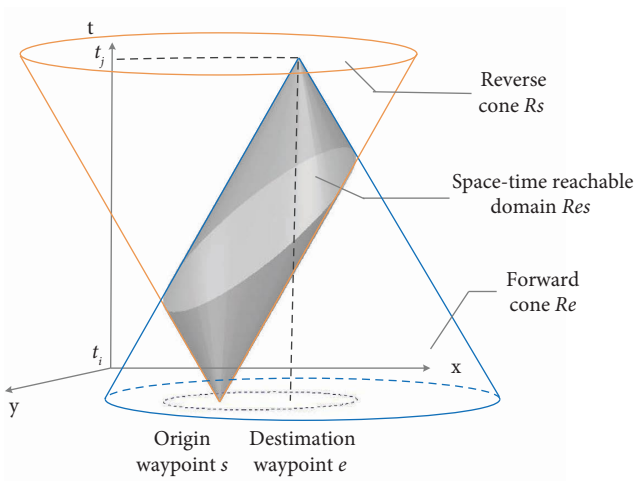


Figure 1: Illustration of space-time reachable area of aircraft.

probability and increase the accuracy of alert. This method of reachable area can be easily extended to different speed constraints by changing the V_m in equations (1) and (2) to the constrained speed value.

4.1.2. Visiting Probability Estimation. The key to risk assessment is to calculate the aircraft conflict probability, which is based on the knowledge of the potential visiting probability of aircraft to each location. The process of visiting probability calculation is as follows.

The first step is to discretize the airspace by employing grid method. As illustrated in the Figure 2, the red node represents the blue cube after discretization.

Based on the discretization, next step is to build the aircraft motion model by employing the Brownian bridge method. The expected location of aircraft A at time t satisfies the following distribution:

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \mathrm{N}\left(\begin{bmatrix} \mu_X(t) \\ \mu_Y(t) \end{bmatrix}, \begin{bmatrix} \sigma_X^2(t) & 0 \\ 0 & \sigma_Y^2(t) \end{bmatrix} \right), \tag{4}$$

where

$$\begin{aligned} \mu_X(t) &= \frac{(t-t_s)x_e + (t_e - t)x_s}{t_e - t_s}, \\ \mu_Y(t) &= \frac{(t-t_s)y_e + (t_e - t)y_s}{t_e - t_s}, \\ \sigma_X^2(t) = \sigma_Y^2(t) &= \frac{(t-t_s)(t_e - t)}{t_e - t_s}. \end{aligned} \tag{5}$$

The visiting probability to locations out of the space-time reachable area is zero. We used a truncated distribution to model the distribution probability of aircraft.

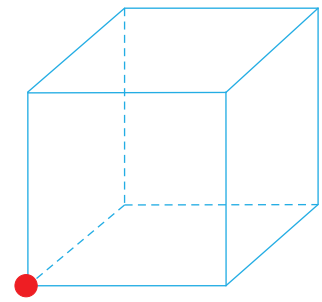


Figure 2: Illustration of airspace discretization.

$$\begin{aligned} T(x(t)) &= \frac{\mathrm{N}\left(\mu_X(t), \sigma_X^2(t)\right)}{\phi(U_X(t)) - \phi(L_X(t))}, \\ T(y(t)|x(t)) &= \frac{\mathrm{N}\left(\mu_Y(t), \sigma_Y^2(t)\right)}{\phi(U_Y(t)) - \phi(L_Y(t))}, \end{aligned} \tag{6}$$

where $T(x(t))$ is the location distribution probability along the x-axis, and $T(y(t)|x(t))$ is that along the y-axis. $\phi(\cdot)$ is the cumulative density function of the above normal distribution. $U_X(t)$ and $L_X(t)$ are the upper and lower bounds of the space-time reachable area along the x-axis at time t, and $U_Y(t)$ and $L_Y(t)$ are those along the y-axis at time t. The specific derivation and calculation process is available in our previous work. Thus, the visiting probability of aircraft A to location (x, y) at time t is able to be deduced by equation (10).

$$\begin{aligned} \mathrm{Prob}(x, y, t) = &\frac{\mathrm{N}\left(\mu_X(t), \sigma_X^2(t)\right)}{\phi(U_X(t)) - \phi(L_X(t))} \\ &\times \frac{\mathrm{N}\left(\mu_Y(t), \sigma_Y^2(t)\right)}{\phi(U_Y(t)) - \phi(L_Y(t))} \times \frac{1}{t_e - t_s}. \end{aligned} \tag{7}$$

The proposed method is able to extend to three-dimensional application, one approach is to generate the space-time reachable area in x–z plane with the same way as in x-y plane. By multiplying the conflict probability of x-y dimension and x–z dimension of each discretized point and adding them up, we are able to get the conflict probability of aircraft in three-dimension scale.

4.2. Flight Plan Risk Assessment. Based on the quantification of the aircraft's visiting probability of the space-time reachable area, it is possible to estimate the conflict probability and risk level of the flight plan. The most common type of conflict risk of flight plans is the entrance to a certain special airspace (SA) that is restricted to access, the airspace affected by extreme weather (WA), or the space-time reachable area of other flight plans. This section proposes three conflict risk calculation methods according to different characteristics of different types of conflict and addresses the risk assessment process and alerting strategy.

4.2.1. Conflict Risk with Special Airspace. SA includes the airspace that is restricted for military purposes (RA) and danger airspace (DA). RA is a defined airspace that limits the entrance of aircraft in time or altitude. DA is the airspace delineated for the existence of dangerous activities during the specified time. The common characteristic of these two types of airspace is that all of them have certain open and closed times known in advance, and the size, shape, and position of the airspace often do not change with time [32]. Therefore, to facilitate calculations, these two types of airspace are often depicted by regular geometric shapes.

To generate the space-time special airspace set L, the first step is to set the plane of route segment r as the x-y plane, set the time dimension as the z-axis, and establish a three-dimensional coordinate system to represent the space-time airspace. Second, select a proper discretization parameter to discretize the space-time domain into several discretized units. Then, the discretized units of SA are marked as 1, and the rest are marked as 0 according to the information such as the location, boundary, and open time of the SA. The space-time SA set L is generated by all discretized units of 1.

Since the aircraft is strictly restricted from entering these two types of airspace, the conflict risk ($\mathrm{CR}_{\mathrm{AL}}$) between aircraft A and the two types of airspace can be obtained by the following equation:

$$CR_{AL} = \sum_{i=1}^{n} I_{\mathrm{SA}} \times \mathrm{Prob}_A\left(x_i, y_i, t_i\right), i \in \mathrm{STPCS}_{AL}, \tag{8}$$

where $\mathrm{STPCS}_{\mathrm{AL}}$ is the space-time potential conflict space between aircraft A and SA, which can be generated by intersecting the space-time reachable area of aircraft A and the space-time SA set L, as shown in Figure 3. I_{SA} is the severity index of the aircraft that enters SA, which can be decided by the air traffic controller according to their experience.

4.2.2. Conflict Risk with the Weather-Influenced Airspace. Dangerous weather conditions such as lightning, tornadoes, thunderstorms, gusts, hail, and downhill winds in the mountains pose a safety hazard to aircraft flying and can cause flight delays and affect the operational efficiency of airspace systems. Avoidance of the airspace influenced by hazardous weather (WA) is necessary during the flying process. The characteristic of WA is that the vertical size, shape, upper and lower altitude bound, and the location of airspace tend to change over time [33]. Therefore, it is necessary to model WA according to the results of weather forecasting; then, the subsequent conflict probability calculation can be performed.

The process of generating the space-time weather influenced airspace set W as follows. First, the centre of WA is considered the centroid of its movement, and the space-time path of WA movement can be modelled according to predicted movement information, including vertical and altitude direction and velocity. Then, we determined the range of WA with a radius of the expected fluctuation capacity. Using the discretization method in Section 4.1, the discretized units of WA are marked as 1, and the remaining units are marked as 0 according to the information, including the space-time path range of WA. The space-time WA set W is generated by all discretized units of 1.

Since the entry of aircraft into the WA is not completely prohibited, it is necessary to build a model for the severity index of intruding the WA. Notice that the risk of aircraft entering the marginal of WA is much less than the risk of aircraft entering the core of WA, and we assume that the severity index of intruding the WA I_{WA} follows a normal distribution. Thus, the conflict risk (CR_{A2}) between aircraft A and WA can be obtained by the following equation:

$$CR_{AW} = \sum_{i=1}^{n} \varphi_{IA}\left(x_i, y_i, t_i\right) \times \mathrm{Prob}_A\left(x_i, y_i, t_i\right), i \in \mathrm{STPCS}_{AW}, \tag{9}$$

where $\mathrm{STPCS}_{\mathrm{AW}}$ is the space-time potential conflict space between aircrafts A and WA, which can be generated by intersecting the space-time reachable area of aircraft A and space-time WA set W, as shown in Figure 4.

4.2.3. Aircraft Conflict Risk between Aircraft Pairs. There are often potential conflicts between aircraft. When the distance between two aircraft is less than the safety separation, conflict occurs [34]. Therefore, the conflict risk ($\mathrm{CR}_{\mathrm{AB}}$) between aircraft A and B can be obtained by the following equation:

$$CR_{AB} = \sum_{i=1}^{n}\sum_{j=1}^{n} I_{AB} \times \mathrm{Prob}_{AB}(i, j), i, j \in \mathrm{STPCS}_{AB},$$
$$\begin{cases} \mathrm{Prob}_{AB}(i, j) = \mathrm{Prob}_A\left(x_i, y_i, t_i\right) \times \mathrm{Prob}_B\left(x_j, y_j, t_j\right), \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \le D, \\ \mathrm{Prob}_{AB}(i, j) = 0, \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} > D, \end{cases} \tag{10}$$

where $\mathrm{STPCS}_{\mathrm{AB}}$ is the potential space-time conflict area between aircraft A and B, which can be generated by intersecting the space-time reachable area of the two aircraft, as shown in Figure 5. D is the safety separation criteria. i and j are any two discretization units in the potential space-time conflict domain; I_{AB} is the severity index of the conflict

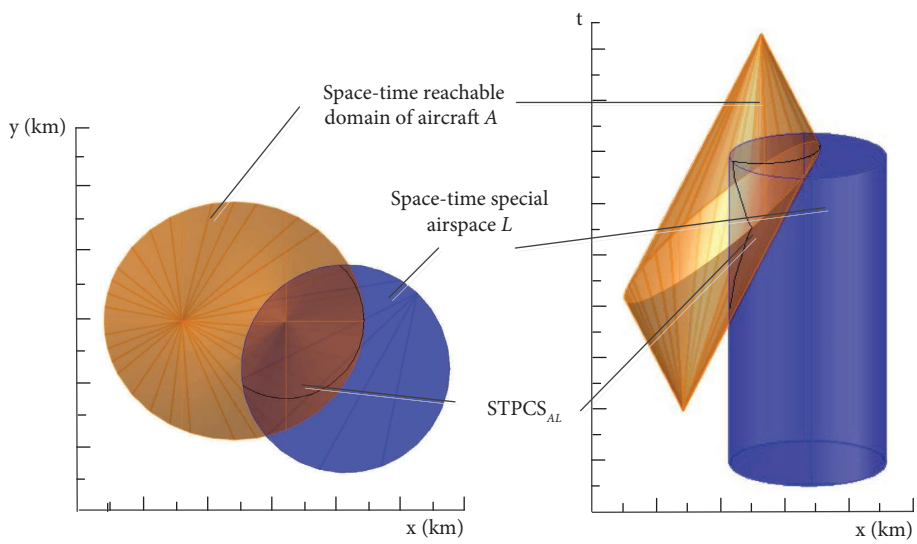


Figure 3: Conflict with the special airspace.

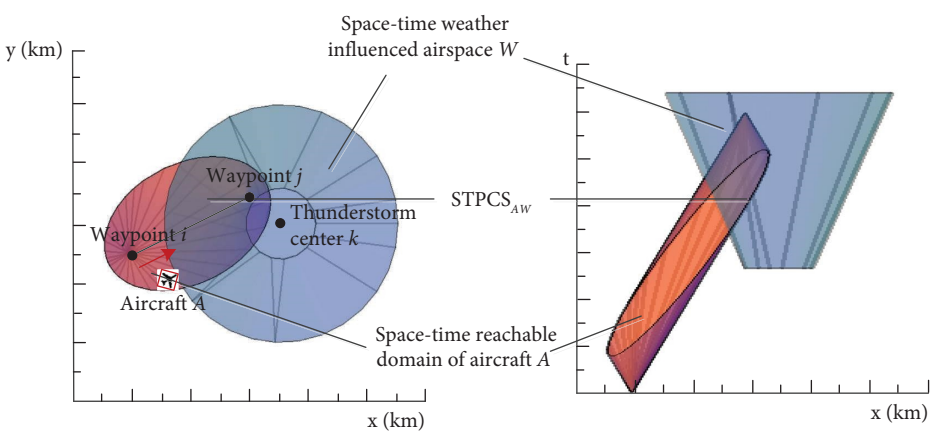


Figure 4: Conflict with weather-influenced airspace.

between aircraft pairs, which can be decided by the air traffic controller according to their experience.

4.2.4. Risk Assessment Process and Alerting Strategy. Based on the results of the conflict risk calculation of the three types of conflicts, the safety index SI_A of flight plan A can be calculated by the following equation:

$$SI_A = \frac{1}{(\alpha \cdot CR_{AS} + \beta \cdot CR_{AW} + \gamma \cdot CR_{AB} + 0.0001)}, \quad (11)$$

where α, β, γ are the weights of conflict with SA, WA, and other aircraft, respectively, which can be set and adjusted according to the requirements of the air traffic controller considering different sectors, traffic volume, and peak and low peak periods, since different airspace environments have different degrees of tolerance and restriction to various types of conflicts [35]. If the aircraft involves no conflict, we add a value of 0.0001 to the formula to ensure that the safety index is always solvable. In other words, when the flight plan can be executed with no interference, it will have a high safety index of 10000. It is noteworthy that here we consider that various type of conflict is independent, so the probability under each conflict type is superposed to obtain the total risk index of aircraft. Sometimes, superposition of multiple type of conflict may exists. In this situation, comparison between conflict probabilities of different types should be made, and the larger conflict probability is used to calculate the risk index. For example, if an aircraft's reachable area enters into a restricted airspace, we considered this flight plan have conflict risk with restricted airspace. If its reachable area also intersects with the reachable area of other aircraft, we need to calculate these two situations and judge which one has higher conflict probability. The higher conflict probability will be used to assess the conflict risk of flight plan.

In our study, the whole flying procedure is divided into several air route segments, we tried to calculate the conflict probability on each air route segments and sum them up to reflect the total conflict risk of the whole flying procedure.

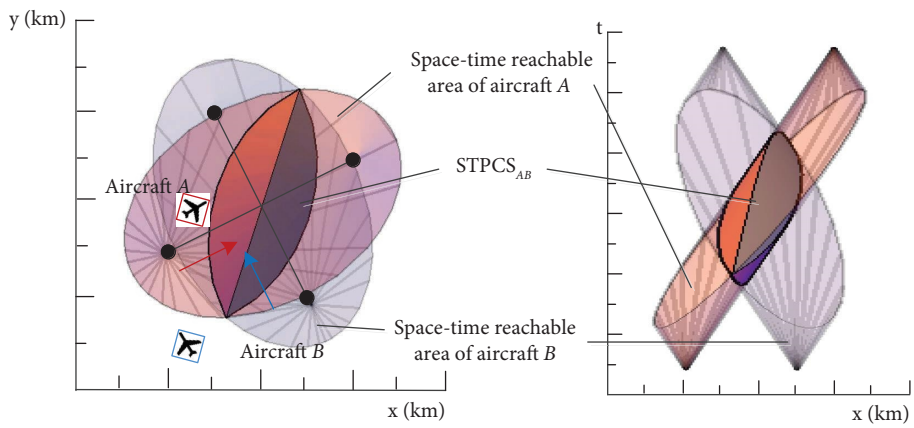


Figure 5: Conflict of flight plan between aircraft pairs.

Based on the proposed flight plan risk assessment method, the risk assessment process can be conducted as follows.

Step 1. M hours before the beginning of the flight mission, extract the information including the flight model, waypoints, planned time of passing each waypoint, and maximum flight velocity according to flight plan F_A of aircraft A.

Step 2. Number the route segment formed by each set of two waypoints r, $r = 1, 2, 3, \ldots, k$;

Step 3. Extract the airspace configuration information related to flight plan F_A, determine whether the airspace contains a special airspace, and generate a space-time special airspace set L according to the boundary, location, and opening time of the special airspace;

Step 4. Extract the airspace meteorological information related to flight plan FA, determine whether there is hazardous weather that affects the aircraft's flying activities, and generate a space-time weather-influenced airspace set W according to the type of hazardous weather, expected spread range and velocity, occurrence period, and moving trajectory;

Step 5. Generate the space-time reachable area RAr, $r = 1, 2, 3, \ldots, k$, on the route segment r of flight plan FA. The set of RAr on all route segments is defined as the space-time reachable area of flight plan FA, which is denoted by RA;

Step 6. Determine the space-time reachable areas RA, RB, ..., RN for each aircraft A, B, ..., N with flying missions;

Step 7. Determine whether the space-time reachable area of the aircraft and the corresponding space-time SA set L, space-time WA set W, and space-time reachable areas of other aircraft RA, RB,..., RN intersect. If the intersection is an empty set, output the safety index value of 10000 and offer the instruction of "no need to further adjust the flight plan";

Step 8. Calculate the conflict risk and safety index if there is an intersection set and conduct safety alert according to the value of the safety index. Determine the threshold value according to different airspace situations and different requirements, and compare the safety index value with the threshold value of alerting. For flight plans with safety index values greater than the threshold, output the instructions of "safety not guaranteed" and "flight plan adjustment recommended," and trigger a secondary level warning; for flight plans with safety index values lower than the threshold, output the instructions of "high conflict risk", "please adjust flight plan," and trigger a first-level warning.

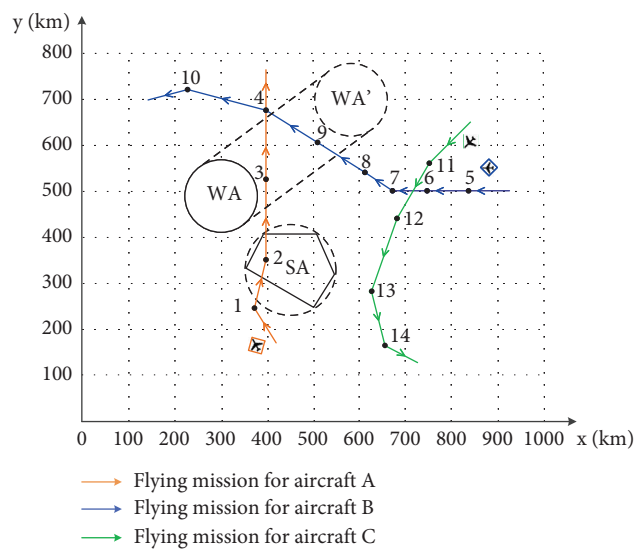


Figure 6: Flight plan of aircraft (A, B, and C) and the airspace environment.

5. Simulation Results and Discussions

To validate the effectiveness of the proposed flight plan risk assessment method, this section adopts a three-aircraft scenario by considering conditions with and without the three types of conflict.

The simulation airspace configuration is illustrated in Figure 6. WA is a thunderstorm area initially at location (300, 490), moves northeast from 14 : 14 : 00 to 15 : 14 : 00,

Table 1: 14 waypoints' locations.

Waypoint no	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Location	(375, 245)	(400, 350)	(400, 525)	(400, 675)	(840, 500)	(750, 500)	(675, 500)	(615, 540)	(512, 605)	(230, 720)	(750, 560)	(680, 440)	(635, 285)	(645, 175)

Table 2: Passing waypoints and time of arrival for aircraft (A, B, and C).

Aircraft	Passing waypoints	Time of arrival
A	1-2-3-4	14:25:10–14:35:10–14:50:00–14:06:50
B	5-6-7-8-9-4-10	14:19:00–14:27:20–14:33:10–14:41:40–14:49:30–15:02:30–15:14:00
C	11-12-13-14	14:38:00–14:50:00–15:01:40–15:09:30

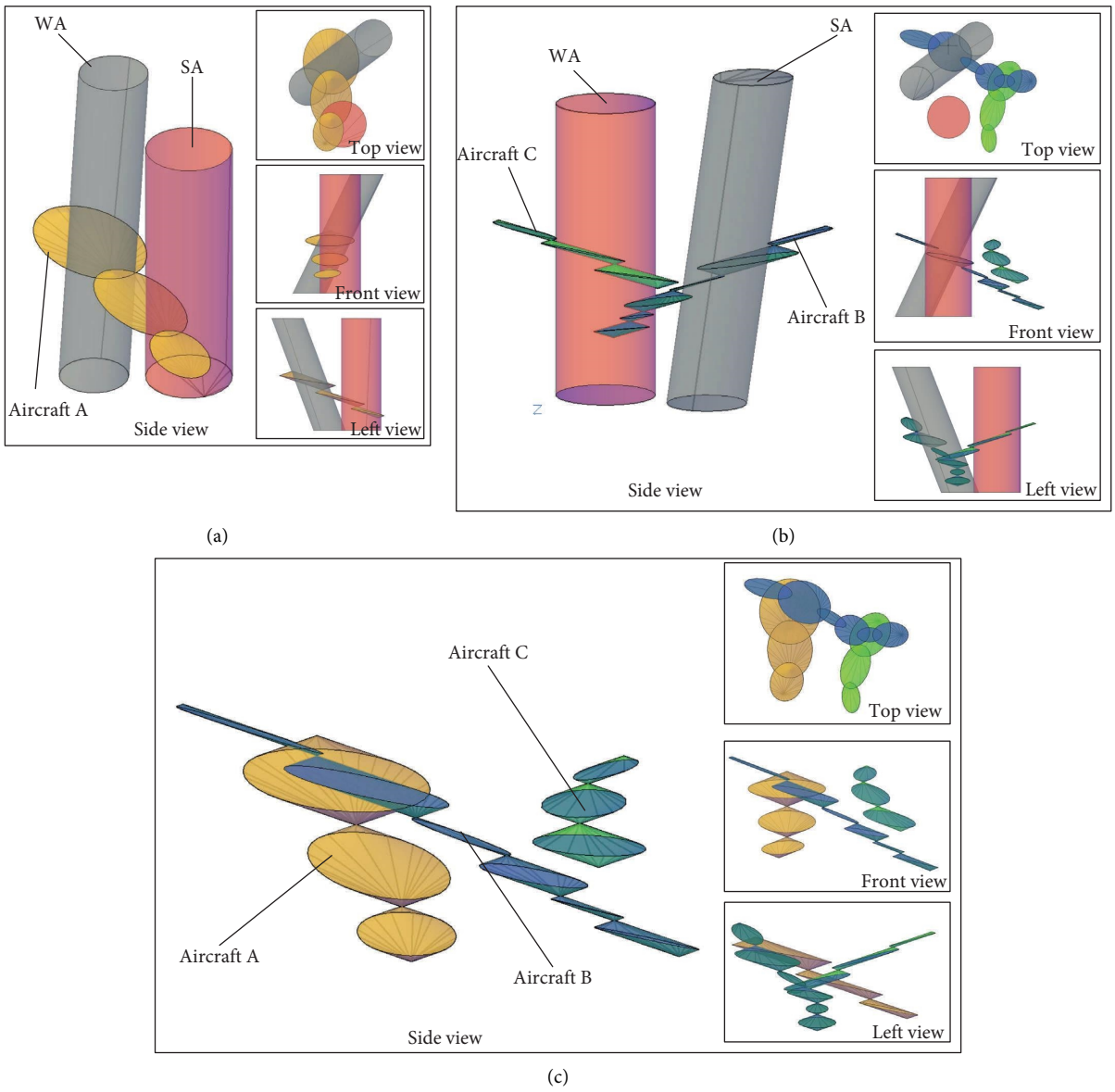


Figure 7: Conflict situation: (a) conflict of aircraft A; (b) conflict of aircraft B; and (c) conflict of aircraft C.

and arrives at location (580, 700) after 60 minutes of movement. The influence radius of the thunderstorm area is approximately 80 km. SA is a polygonal sector with restricted access due to military control, which can be abstracted by its adjacent circle. Besides, the conflict probability of different particle size may have some differences due to calculation error, but there will be no tremendous differences in the order of magnitude. Thus, to balance between the calculation efficiency and accuracy, the discretization parameter is set to be 5 km × 5 km × 10 s. In this example, SA is represented by a circle with a centre at (450, 330) and a radius of 100 km. Detailed parameters of each flight plan can be found in Tables 1 and 2. For example, the flight plan of aircraft A will pass waypoint 1 at 14 : 25 : 10, waypoint 2 at 14 : 35 : 10, waypoint 3 at 14 : 50 : 00, and finally waypoint 4 at 14 : 06 : 50. The maximum speed of the aircraft A, B, and C is assumed to be 0.279 km/s.

By generating the space-time reachable area of 3 aircraft in each route segment, we find that aircraft A has conflicts with SA, WA, and aircraft B. Besides, Aircraft B has conflicts

Table 3: Conflict probability of each aircraft.

Aircraft	*A*				*B*				*C*			
Conflict with	SA	WA	*B*	*C*	SA	WA	*A*	*C*	SA	WA	*A*	*B*
Conflict probability	0.083	0.458	$9.25e-15$	0	0	0.066	$9.25e-15$	0	0	0	0	0

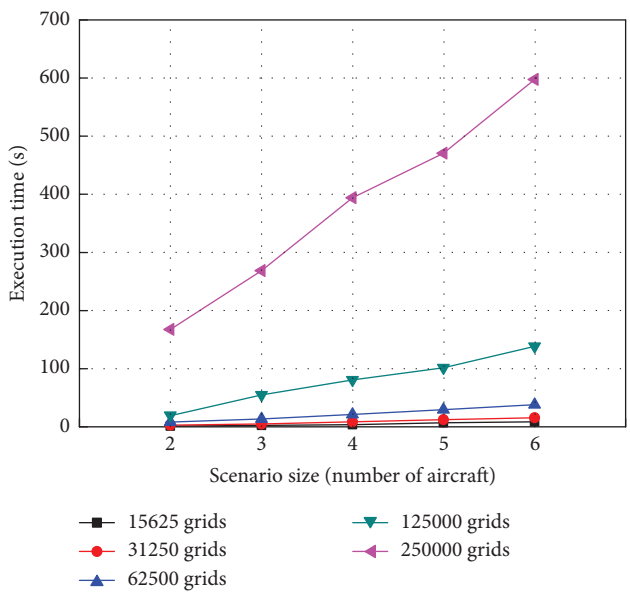


Figure 8: Relation between execution time and scenario size.

with aircraft *A* and WA. Aircraft *C* has no conflicts. The conflict situation is shown in Figure 7.

In this case, we assume that the weight of conflict with SA is set to 3, the weight of conflict with WA is set to 2, the weight of actual conflict with other aircraft is set to 4, and the weight of potential conflict with other aircraft is set to 1.

The results show that the conflict probability of each aircraft is shown in Table 3.

Using equation (10), we can obtain the safety index of flight plans *A*, *B*, and *C*. The results show that the flight plan of aircraft *A* has a very low safety level of 1.05, which requires subsequent flight plan adjustment to improve its safety index to an acceptable level. Aircraft *B* has a reasonable safety level of 36.90, which must be considered during the flying process. Aircraft *C* has a very high safety level of 10000.

Besides, to analyse how the algorithm scales with scenario size, the algorithm was performed for two to six aircraft situations. Each situation was performed three times to achieve an average execution time. As illustrated in Figure 8, the algorithm execution time increases linearly with scenario size.

Comparing with conventional tactical aircraft conflict detection and resolution method, the proposed flight plan risk assessment method is able to judge potential conflicts and estimate conflict probability and evaluate flight plan risk before aircraft takeoff; by optimizing the flight plan, the proposed method has great potential in avoiding tactical aircraft conflict risk and establish the first barrier of the aviation safety.

6. Conclusions

This study provides a quantitative basis for the risk assessment of flight plans in the context of four-dimensional trajectory-based operations. Taking the flight plan as the input, before the flying mission starts, the space-time reachable area between every two waypoints is calculated for each aircraft. The problem of when and where will have potential conflicts which can be also determined by finding the intersection of the space-time reachable areas of aircraft pairs. Aircraft motion considering various uncertain factors is considered a random Brownian motion. The definition of the safety index is proposed, and its calculation process and category method are addressed.

The proposed risk assessment method is expected to effectively reduce aircraft conflict risk through flight plan adjustment in the strategic stage, which will improve the safety of the entire air transport system and balance the high-speed growth of air traffic demand and limited airspace resources. In addition, because the proposed risk assessment method can serve as an assistant for flight plan adjustment in the strategic stage, it can greatly reduce the workload of air traffic controllers and pilots caused by trajectory adjustment in the tactical stage, reduce the impact on flight due to uncertainties, improve the on-time rate of flights and reduce conflict risks.

Determine various types of conflict risk weight coefficients and alarm thresholds according to different scenarios is a rather complex problem, since different airspace environments have different degrees of tolerance and restriction to various types of conflicts. First, scientific classification should be carried out for different airspace scenarios. Then, according to the actual aircraft operation data and historical warning data, the conflict risk weight coefficient and early warning threshold should be determined for each classification. This conflict risk coefficient and threshold should also be verified by the experienced air traffic controllers. Through this combination of subjective and objective methods, it is possible to ensure that the determination of the conflict risk weight coefficient and threshold is consistent with the actual situation, which can assist the system to provide timely, accurate, and reliable early warning information. On the basis of the conflict risk assessment framework proposed in this article, reasonably determining the conflict risk weight coefficient and alarm threshold is the key to the practical application of this method, which is also our main further research work.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to acknowledge the financial support from the National Natural Science Foundation of China (Grant no. 52102410), Guangxi Science and Technology Base and Talent Special Project (AD19245021), 2022 Guangdong Social Science Planning Project (GD22YGL18), 2022 "Yangcheng Scholar" Research Project of Guangzhou Education Bureau (202235334), 2023 Basic Research Plan Program of Guangzhou (SL2023A04J00686), and Young Innovative Talents Project of General Colleges and Universities of Guangdong Province (Grant no. 2021KQNCX073).

References

[1] M. Rinaldi, R. Montanari, and E. Bottani, "Improving the efficiency of public administrations through business process reengineering and simulation: a case study," *Business Process Management Journal*, vol. 21, no. 2, pp. 419–462, 2015.

[2] E. K. Burke, P. De Causmaecker, G. De Maere, J. Mulder, M. Paelinck, and G. Vanden Berghe, "A multi-objective approach for robust airline scheduling," *Computers & Operations Research*, vol. 37, no. 5, pp. 822–832, 2010.

[3] J. W. Yen and J. R. Birge, "A stochastic programming approach to the airline crew scheduling problem," *Transportation Science*, vol. 40, no. 1, pp. 3–14, 2006.

[4] P. J. Lederer and R. S. Nambimadom, "Airline network design," *Operations Research*, vol. 46, no. 6, pp. 785–804, 1998.

[5] S. Cheng, Y. Zhang, S. Hao, R. Liu, X. Luo, and Q. Luo, "Study of flight departure delay and causal factor using spatial analysis," *Journal of Advanced Transportation*, vol. 2019, no. 29, Article ID 3525912, 11 pages, 2019.

[6] N. Zimmermann and F. A. C. Mendonca, "The impact of human factors and maintenance documentation on aviation safety: an analysis of 15 years of accident data through the PEAR framework," *Collegiate Aviation Review International*, vol. 39, no. 2, pp. 1–25, 2021.

[7] R. Xu and F. Luo, "Risk prediction and early warning for air traffic controllers' unsafe acts using association rule mining and random forest," *Safety Science*, vol. 135, Article ID 105125, 2021.

[8] K. Eyferth, C. Niessen, and O. Spaeth, "A model of air traffic controllers' conflict detection and conflict resolution," *Aerospace Science and Technology*, vol. 7, no. 6, pp. 409–416, 2003.

[9] H. J. Shyur, "A quantitative model for aviation safety risk assessment," *Computers & Industrial Engineering*, vol. 54, no. 1, pp. 34–44, 2008.

[10] B. Kirwan and H. Gibson, "A human reliability assessment tool for air traffic safety management -technical basis and preliminary architecture," in *The Safety of Systems*, F. Redmill and T. Anderson, Eds., Springer, London, UK, 2007.

[11] J. A. Pérez-Castán, F. Gómez Comendador, Á. Rodríguez-Sanz, and R. M. Arnaldo Valdés, "Conflict-risk assessment model for continuous climb operations," *Aerospace Science and Technology*, vol. 84, pp. 812–820, 2019.

[12] M. Wu, W. Yang, and K. Bi, "Conflict resolution strategy based on flight conflict network optimal dominating set," *International Journal of Aerospace engineering*, vol. 10, Article ID 9747531, 2022.

[13] X. Zhang and S. Mahadevan, "Bayesian network modeling of accident investigation reports for aviation safety assessment," *Reliability Engineering & System Safety*, vol. 209, Article ID 107371, 2021.

[14] F. Netjasov, "Framework for airspace planning and design based on conflict risk assessment: Part 1: conflict risk assessment model for airspace strategic planning," *Transportation Research Part C: Emerging Technologies*, vol. 24, pp. 190–212, 2012.

[15] T. Lehouillier, M. I. Nasri, F. Soumis, G. Desaulniers, and J. Omer, "Solving the air conflict resolution problem under uncertainty using an iterative bi-objective mixed integer programming approach," *Transportation Science*, vol. 51, no. 4, pp. 1242–1258, 2017.

[16] D. Natalia and L. Salvatore, "Risk assessment for civil airport," *International Journal of Traffic and Transportation Engineering*, vol. 4, no. 1, Article ID 01528255, 2014.

[17] A. Oztekin, "Development of a safety assessment tool for air traffic control system," SAE International, Warrendale, PL, USA, Technical Paper, 2015.

[18] R. M. Arnaldo Valdés, V. F. Gómez Comendador, L. Perez Sanz, and A. Rodriguez Sanz, "Prediction of aircraft safety incidents using Bayesian inference and hierarchical structures," *Safety Science*, vol. 104, pp. 216–230, 2018.

[19] L. Cui, J. Zhang, B. Ren, and H. Chen, "Research on a new aviation safety index and its solution under uncertainty conditions," *Safety Science*, vol. 107, pp. 55–61, 2018.

[20] X. Zhang, Y. Liu, Y. Zhang, X. Guan, D. Delahaye, and L. Tang, "Safety assessment and risk estimation for unmanned aerial vehicles operating in national airspace System," *Journal of Advanced Transportation*, vol. 2018, Article ID 4731585, 11 pages, 2018.

[21] S. H. Stroeve, P. Som, B. A. van Doorn, and G. Bert Bakker, "Strengthening air traffic safety management by moving from outcome-based towards risk-based evaluation of runway incursions," *Reliability Engineering & System Safety*, vol. 147, pp. 93–108, 2016.

[22] M. John, "Icao/atmrpp wg/25 wp/603 proposal for the development of tbo concept," 2014, https://www.icao.int/airnavigation/tbo/Pages/Why-Global-TBO-Concept.aspx.

[23] J. Richard, *Icao/Atmrpp Wg/27 Wp/649 Input to The Atmrpp Trajectory Based Operations (Tbo) Conops*, Richard Jehlen (FAA), Burke, VA, USA, 20 October 2014.

[24] J. Krozel, "Intent inference for free flight aircraft," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Denver, CO, USA, August 2000.

[25] L. Guan, Y. Zhang, and Z. Xing, "An optimal simulation model of aircraft dynamic pushback control strategies," *Journal of Nanjing University of Aeronautics & Astronautics*, vol. 6, pp. 54–60, 2018.

[26] H. Li, Z. Wang, J. Wang, and F. Shen, "Deep reinforcement learning based conflict detection and resolution in air traffic control," *IET Intelligent Transport Systems*, vol. 13, no. 6, pp. 1041–1047, 2019.

[27] P. Nordlund and F. Gustafsson, "Probabilistic noncooperative near mid-air collision avoidance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 2, pp. 1265–1276, 2011.

[28] A. Reynolds-Feighan, "Traffic distribution in low-cost and full-service carrier networks in the US air transportation market," *Journal of Air Transport Management*, vol. 7, no. 5, pp. 265–275, 2001.

[29] P. Barla and C. Constantatos, "Airline network structure under demand uncertainty," *Transportation Research Part E: Logistics and Transportation Review*, vol. 36, no. 3, pp. 173–180, 2000.

[30] S. Hao, S. Cheng, and Y. Zhang, "A multi-aircraft conflict detection and resolution method for 4-dimensional trajectory-based operation," *Chinese Journal of Aeronautics*, vol. 31, no. 7, pp. 1579–1593, 2018.

[31] S. Hao, Y. Zhang, S. Cheng, R. Liu, and Z. Xing, "Probabilistic multi-aircraft conflict detection approach for trajectory-based operation," *Transportation Research Part C: Emerging Technologies*, vol. 95, pp. 698–712, 2018.

[32] M. Soler, A. Olivares, and E. Staffetti, "Multiphase optimal control framework for commercial aircraft four-dimensional flight-planning problems," *Journal of Aircraft*, vol. 52, no. 1, pp. 274–286, 2015.

[33] A. Franco and D. Rivas, "Optimization of multiphase aircraft trajectories using hybrid optimal control," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 3, pp. 452–467, 2015.

[34] J. Hoekstra, R. Gent, and R. Ruigrok, "Conceptual design of free flight with airborne separation assurance," in *Proceedings of the AIAA Guidance, Navigation, Control Conference*, Boston, MA, USA, August 1998.

[35] K. J. Ruskin, C. Corvin, S. Rice, G. Richards, S. R. Winter, and A. Clebone Ruskin, "Alarms, alerts, and warnings in air traffic control: an analysis of reports from the Aviation Safety Reporting System," *Transportation Research Interdisciplinary Perspectives*, vol. 12, Article ID 100502, 2021.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 1513008, 13 pages
https://doi.org/10.1155/2023/1513008

Research Article

An Integrated Lateral and Longitudinal Decision-Making Model for Autonomous Driving Based on Deep Reinforcement Learning

Jianxun Cui,[1] **Boyuan Zhao**,[1] **and Mingcheng Qu**[2]

[1]*School of Transportation Science and Engineering, Harbin Institute of Technology, Harbin 150090, China*
[2]*Department of Software, Harbin Institute of Technology, Harbin 150001, China*

Correspondence should be addressed to Jianxun Cui; cuijianxun@hit.edu.cn

Received 12 August 2022; Revised 17 September 2022; Accepted 23 September 2022; Published 13 April 2023

Academic Editor: Rui Zhu

Decision-making is an important component of autonomous driving perception, decision-making, planning, and control pipeline, which undertakes the task of how the ego vehicle makes high-level decision-making behaviors (such as lane change and car following) after sensing the environmental state, and then these high-level decision-making behaviors can be transmitted to the downstream planning and control module for specific low-level action execution. Based on the method of deep reinforcement learning (specifically, Deep Q network (DQN) and its variants), an integrated lateral and longitudinal decision-making model for autonomous driving is proposed in a multilane highway environment with both autonomous driving vehicle (ADV) and manual driving vehicle (MDV). The classic MOBIL and IDM models are used for the lateral and longitudinal decisions of MDV (i.e., lane changing and car following), while the lateral and longitudinal decisions of ADV are dominated by deep reinforcement learning models. In addition, this paper also uses the nonlinear kinematic bicycle model and two-point visual control model to realize the low-level control of both MDV and ADV. By setting a reasonable state, action, and reward function, this paper has carried out a large number of simulation experiments on the proposed autonomous driving decision-making model based on deep reinforcement learning in a three-lane road environment. The results show that under such scenario setting conditions, the deep reinforcement learning-based model proposed in this paper performs well in autonomous driving safety and travel efficiency. At the same time, when compared with the classical rule-based decision-making model (MOBIL&IDM), it is found that the model proposed in this paper can significantly achieve better results in episode rewards after stable training. In addition, through a large number of hyper-parameter tuning experiments, the performance of DQN, DDQN, and dueling DQN models, which are also deep reinforcement learning-based decision-making models, under different hyper-parametric configurations is compared and analyzed, which can provide a valuable reference for the specific scenario application of these models.

1. Introduction

Autonomous driving is hot research and practical issue in the fields of road traffic engineering, vehicle engineering, and artificial intelligence in recent years, which is considered to have great potential in alleviating traffic congestion, reducing environmental pollution, improving traffic safety performance, and even systematically changing the future traffic mobility pattern [1]. In order to realize autonomous driving, a vehicle needs to be able to accurately perceive the state of itself and the surrounding environment, then make corresponding behavioral decisions and consequently generate a safe, efficient trajectory based on perceptual understanding, and finally track the generated trajectory as accurately as possible by controlling the throttle, brake pedal, and steering wheel [2]. This autonomous driving process is usually described as a modular pipeline as shown in Figure 1. After the travel user gives the global information, such as the travel destination and navigation route, the autonomous vehicle will collect the environmental information through its own installed cameras, LIDAR, and other types of sensors at a certain frequency, and then the collected

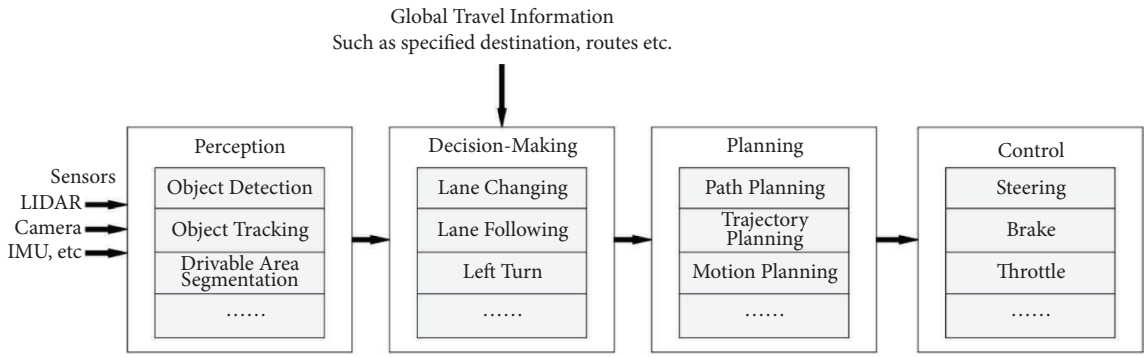


Figure 1: Modular pipeline of autonomous driving.

raw sensor data will be input into the perception module for environmental semantic understanding tasks such as object detection and tracking. Further, on the basis of state perception and the user's global travel information, the autonomous vehicle will make local behavior decisions such as whether to change lanes and further generate behavior instructions to the planning module to generate the optimal trajectory. Finally, the generated trajectory can be tracked by controlling the throttle, brake, steering wheel, and other actors.

Since decision-making is an important part that links perception and trajectory planning and greatly determines the safety and efficiency of autonomous driving, extensive research around this issue can be found in the literature. In general, the typical research method of autonomous driving decision-making can be mainly categorized into 4 classes: rule-based [3–5], classical machine learning-based [6–8], deep reinforcement learning-based [9–12], and deep imitation learning-based [13–15]. Among many research methods, deep reinforcement learning has received great attention in recent years because it does not need a lot of human labeled training data, the learning style is closer to human learning, and the generalization ability is strong. Despite the above advantages, for the application of deep reinforcement learning in automated decision-making modeling, how to construct an effective representation of the environmental state, how to design an effective reward function, and how to compare and analyze the performance differences between the deep reinforcement learning model and the traditional rule-based model are still challenging and needed to be further studied. In view of this, this paper aims to study the modeling of autonomous driving decision-making based on the DQN and its variants under the condition of the mixture of autonomous driving vehicles and manual driving vehicles in a specific scenario of a multi-lane highway. It is hoped that this research can provide effective models for safety, efficiency, etc. for decision-making in multilane autonomous driving scenarios. At the same time, through a large number of hyper-parameter tuning experiments, we will systematically compare the performance of several classical value-based DRL models (i.e., DQN, DDQN, and Dueling DQN) for autonomous driving decision-making, and further evaluate the performance differences between them and other traditional rule-based decision-making models, so as to provide a valuable reference for autonomous driving decision-making modeling in multilanescenarios.

The contributions of this study include the following aspects.

(1) An integrated lateral and longitudinal decision-making model based on deep reinforcement learning is proposed for autonomous driving in a multilane highway with mixed traffic composed of MDVs and ADVs. A large number of simulation experiments are conducted to verify the effectiveness of the proposed model.

(2) Extensive simulations are conducted to compare the model performance between DRL-based models (i.e., DQN, DDQN, and Dueling DQN) and rule-based models (i.e., IDM and MOBIL), results of which show that DRL-based models are significantly superior to rule-based models for autonomous driving decision making.

(3) Performance comparison between DQN and its variants (i.e., DDQN and Dueling DQN) is also conducted, results of which indicate that DDQN and Dueling DQN do improve the performance of DQN model for autonomous driving decision-making by properly estimating Q values and optimizing network structure in terms of training efficiency and reward acquisition.

(4) With different ADV penetration, the training efficiency of DQN-series models for autonomous driving decision-making is compared, according to the rising of ADV penetration, for a single ADV, the environment becomes more uncertain and complex, so the training process of the DQN-series models is more difficult to be stabilized.

The organization of this study is as follows. Section 2 presents a brief literature review of decision-making of autonomous driving. Section 3 introduces our proposed methodology for modeling decision-making of autonomous driving and Section 4 conducts a large number of simulation experiments to verify the proposed models and the results of

which are discussed. Finally, Section 5 concludes this manuscript and briefly discusses future research directions.

2. Literature Review

Decision-making corresponds to a high-level behavior of an automated vehicle, which decides whether the automated vehicle will change lanes, follow or turn et al. Because decision-making represents the response of autonomous vehicles on the environmental state observation and driving goals, and plays a guiding role in the downstream planning and control module, it has attracted a lot of research in the literature.

In general, the research on autonomous driving decision-making can be divided into rule-based, finite state machine-based, and machine learning-based methods. Rule-based methods are based on some predefined parameters that would tune the algorithm for a specific environment, in which the most representative ones are MOBIL [16] for lateral decision-making and IDM [17] for longitudinal decision-making. A common limitation of these approaches is the lack of flexibility under dynamic situations and diverse driving styles [18]. Since both driving contexts and the behaviors available in each context can be modeled as finite sets, a natural approach to automating this decision-making is to model each behavior as a state in a finite state machine with transitions governed by the perceived driving context such as relative position with respect to the planned route and nearby vehicles. In fact, finite state machines were adopted as a mechanism for behavior control by most teams in the DARPA Urban Challenge [19]. However, because the context of open road autonomous driving is highly complex, dynamic, and uncertain, it is intractable to build all possible driving contexts and their corresponding behaviors into finite state machines in essence, which makes the finite state machine destined to be a simplified modeling method for autonomous driving decision-making and difficult to use in real complex scenes [20]. Machine Learning (ML) based methods have a very good generalization ability for unknown scenes when they are properly trained through a large number of data samples, and there is no need to manually specify rules in advance [21]. Vallon et al. [22] proposed a support vector machine (SVM) model to capture the lane change decision behavior of human drivers. After the lane change demand is generated, the maneuver is executed using an MPC. By extracting the features from surrounding vehicles that are relevant to the lane-changing of the subject vehicle, Bi et al. [23] used a randomized forest and back-propagation neural network to model the process of lane-changing in traffic simulation. ML-based methods above for autonomous driving decision-making research fall into the supervised learning paradigm, so it is necessary to collect a great amount of real-world driving behavior data and annotate a large number of manual driving decision-making behaviors, which is usually very time-consuming and labor-intensive. More importantly, it is difficult to pose autonomous driving as a supervised learning problem as it has a strong interaction with the environment including other vehicles, pedestrians, and road networks [10]. In recent years, another machine learning paradigm, reinforcement learning (especially Deep Reinforcement Learning, DRL), which learns the task in a trial-and-error way that does not require explicit human labeling or supervision on each data sample has been widely used in research of autonomous driving decision-making and control. Ngai and Yung [24] adopted a multiple-goal reinforcement learning (RL) framework to model complex vehicle overtaking maneuvers. For lane-keeping assisting decision-making issues, Sallab et al. [10] adopted Deep Q-Network Algorithm (DQN) and Deep Deterministic Actor-Critic Algorithm (DDAC) to model discrete actions category and continuous actions category of autonomous driving respectively. Wang and Chan [25] applied deep reinforcement learning (DRL) techniques to find optimal control policy for automating decision making on a ramp merge. The proposed methods also have the potential to be extended and applied to other autonomous driving scenarios such as driving through a complex intersection or changing lanes under varying traffic flow conditions. Hoel et al. [26] proposed a Deep Q-Network model automatically to generate a decision-making function to handle speed and lane change. For navigation at occluded intersections, Isele et al. [27] used Deep RL methods to provide efficient automated decision-making strategy, which is able to learn policies that surpass the performance of a commonly-used heuristic approach in several metrics including task completion time and goal success rate and have limited ability to generalize. Although great achievements have been made in the research of autonomous driving decision-making using DRL, applying RL to real-world applications is particularly challenging, especially for autonomous driving tasks that involve extensive interactions with other vehicles in a dynamically changing environment. One significant barrier of applying RL to real-world problems is the required definition of the reward function, which is typically unavailable or infeasible to design in practice. Inverse reinforcement learning (IRL) aims to tackle such problems by learning the reward function from expert demonstrations, thus avoiding reward function engineering and making good use of the collected expert data [28, 29]. However, because of the expensive reinforcement learning procedure in the inner loop, it has limited application in problems involving high-dimensional state and action spaces [30]. To overcome the limitation, some state-of-the-art works were conducted, such as generative adversarial imitation learning (GAIL) [30], guided cost learning (GCL) [31], and adversarial inverse reinforcement learning (AIRL) [32]. Although imitation learning theoretically provides a more stable training process, and there is no need to explicitly specify a reward function, it still needs to collect a large number of expert driving data as a demonstration compared with deep reinforcement learning and faces the problem of distribution shift [33].

In view of the learning advantages of DRL in the complex interactive autonomous driving decision-making, this paper attempts to explore a more intelligent decision-making strategy through effective environmental state representation and a fine design of reward function in a specific multilane mixed driving scenario based on DQN and its

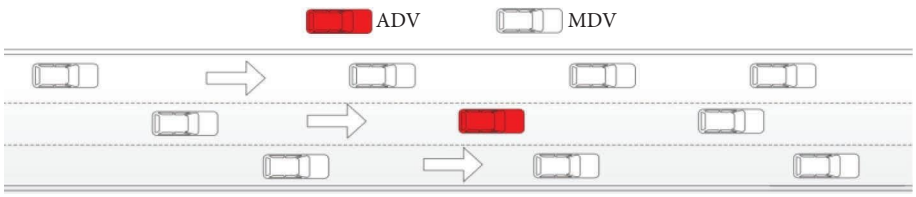


Figure 2: Multilane highway scenario.

variants. Further, combining the proposed DRL-based decision-making models with the low-level effective control model, we will conduct a large number of simulation experiments to determine optimization configuration of various hyper-parameters associated with the decision-making models. In addition, the performance of the proposed decision models will be compared with the traditional rule-based model to validate the efficiency of our models. This research is expected to provide a valuable reference for the application of deep reinforcement learning in autonomous driving decision-making research.

3. Methodology

In this section, we first give the detailed description of the problem that we are addressing in this paper. Next, the rule-based lateral and longitudinal decision-making models of MDV which act as the interacted surrounding traffic of ADV are presented. Then decision-making model of ADV is constructed based on DQN by specifying the state representation, action set, and reward function. Finally, a low-level control model based on a nonlinear kinematic bicycle model combined with two-point visual control is presented to implement the output from the decision-making model of both MDV and ADV.

3.1. Problem Statement. The autonomous driving decision-making scenario concerned in this paper is shown in Figure 2 . This multilane autonomous driving scenario consists of multiple lanes driving in the same direction, in which ADV (in red color) and MDV (in grey color) are in a mixed driving state. The Decision-making of MDV is driven by two rule-based models that are MOBIL and IDM. MOBIL is responsible for lateral decision-making and IDM is responsible for longitudinal decision-making, which will be introduced in detail later. The lateral and longitudinal decision-making of ADV is both achieved by a DRL-based model (i.e., DQN), which is the major research concern of this paper. The output of decision-making models of both MDV and ADV will immediately transmit to the low-level control model which is realized by the nonlinear kinematic bicycle model to generate specific vehicle action execution. The research problem of this paper can be summarized as how to train a safe and effective deep reinforcement learning model by properly representing the environmental state, action set, and reward function of autonomous vehicles in the aforementioned mixed driving scenarios of manual driving and autonomous driving.

3.2. Decision Making of MDV

3.2.1. Longitudinal Decision of MDV. IDM (Intelligent Driver Model) [17] which is a rule-based car following model is employed to model the longitudinal decision making of MDV. IDM was originally proposed in the field of adaptive cruise control (ACC) to generate appropriate acceleration for the ego vehicle based on its relative driving state with the leading on a single lane. The longitudinal decision-making formulas described by IDM are shown in Eq. 1-2.

$$a = a_{\max}\left(1 - \left(\frac{u}{u_d}\right)^{\delta} - \left(\frac{d^*(u, \Delta u)^2}{d}\right)\right), \tag{1}$$

$$d^*(u, \Delta u) = d_{\min} + uT + \frac{u\Delta u}{2\sqrt{ba_{\max}}}, \tag{2}$$

where, a is the instant accelarationof ego vehicle, which is needed to be determined in each decision step; $a_{\max}$ is the maximum acceleration of the ego vehicle; u and u_d is the current and desired speed of the ego vehicle; Δu is the speed difference between the ego vehicle and its leading vehicle; d is the gap between the ego vehicle and its leading vehicle; $d_{\min}$ is the minimum safety gap between the ego vehicle and its leading vehicle; T is safe time headway; b is the desired acceleration of the ego vehicle;

As it is seen in the equation (1) and (2), the original IDM model only restricted the acceleration of the ego vehicle by maximum acceleration $a_{\max}$; however, the minimum deceleration is not indicated. So, a condition depicted by equation (3) is added by us to limit the minimum deceleration of the ego vehicle.

$$a = \begin{cases} a, & a \geq a_{\min} \\ a_{\min}, & \text{otherwise}, \end{cases} \tag{3}$$

where, $a_{\min}$ is the minimum deceleration allowed.

In practice, the MDVs on each single lane execute the IDM longitudinal decision-making model respectively and then generate their own acceleration decisions in each time interval. If there is no leading vehicle in front of an MDV, its Δu and d is set to 0 and $d_{\max}$ (maximum gap for empty lane).

3.2.2. Lateral Decision of MDV. MOBIL (Minimizing Overall Braking Induced by Lane Change) [16] which is a rule-based lane change model is adopted here to make lateral decision of MDV. MOBIL determines whether lane change is safe and accessible according to the relative acceleration

between the ego vehicle and the vehicles on the adjacent lanes. MOBIL's decision-making process is divided into two steps: first, according to the limit of safety standards, the deceleration of new following vehicles should not be too low when lane changing occurs, which is described in (4).

$$\hat{a}_{\text{new-follower}} > b_{\text{safe}}, \tag{4}$$

where, $\hat{a}_{\text{new-follower}}$ is the acceleration of new following vehicles after lane change of the ego vehicle, which can be calculated by IDM; b_{safe} is the maximum safe deceleration. Second, if the first condition defined in equation (4) is met, MOBIL will check the second condition defined in equation (5) to make a final decision about whether trigger a lane change of the ego vehicle.

$$\hat{a}_{\text{ego}} - a_{\text{ego}} + p\left(\hat{a}_{\text{new-follower}} - a_{\text{new-follower}}\right) + q\left(\hat{a}_{\text{old-follower}} - a_{\text{old-follower}}\right) > a_{th}, \tag{5}$$

where, $\hat{a}_{\text{ego}}, a_{\text{ego}}$ are the new acceleration of the ego vehicle calculated by IDM after lane change and the old acceleration before lane change; $\hat{a}_{\text{new-follower}}, a_{\text{new-follower}}$ are the new and old accelerations respectively of the new follower vehicle when lane change of the ego vehicle occurs; $\hat{a}_{\text{old-follower}}, a_{\text{old-follower}}$ are the new and old accelerations respectively of the old follower vehicle when lane change of the ego vehicle occurs; p and q are politeness factors respectively of the new and old following vehicles; a_{th} is a predefined threshold value. Equation (5) indicates that only when the collective acceleration gain is greater than a predefined threshold, the lane change behavior of the ego vehicle can be truly triggered.

3.3. Decision Making of ADV. Both lateral and longitudinal decisions of ADV are modeled by the DRL method which here refers to DQN specifically. DQN was originally proposed by Mnih et al. [34] for playing Atari games, which is an effective DRL algorithm for discrete decision problems by combing deep learning and reinforcement learning. Traditionally, the Q value function corresponding to a specific state and action is represented by a table, which is hard to handle the problem with a large space of state variable. DQN overcomes this problem by using a deep neural network to represent the Q value function as $Q(s, a, \theta)$ instead of a table, where θ represents the learnable parameters of the neural network.

(1) **Q value function** $Q(s, a, \theta)$ **of ADV.** Each decision-making action (e.g. left change and right change) of one ADV at the arbitrary time step is realized by choosing the action with the best-expected return according to the strategy of ε -greedy, which needs to establish the Q value function, $Q(s, a, \theta)$ of each state-action pair $(s, a) \cdot (s \in S, a \in A$, where S, A are state and action sets respectively). Here, a fully-connected neural network which takes one specific state as input and the corresponding Q value of each available action as output will be used to represent the Q value function.

(2) **Updating rule of** $Q(s, a, \theta)$. The updating rule of $Q(s, a, \theta)$ is described in Equation .

$$Q_{k+1}(s, a) = Q_k(s, a) + \alpha\left[r(s, a) + \gamma Q_k(s', a') - Q_k(s, a)\right], \tag{6}$$

where, $Q_k(s, a), Q_{k+1}(s, a)$ represents the Q values of kth and $k+1$th step, respectively; $r(s, a)$ is the instant reward received by executing action a under the state s; γ is the discount factor of future return; $\alpha \in [0, 1]$ is the learning rate which is used to trade-off between old and new learned experiences; s' is the state of next step after ADV takes action a under the state s; a' is the adopted action by ADV under state s' according to ε -greedy strategy with the kth Q value function (current or unupdated Q value function).

(3) **Exploration strategy of ADV**. In the process of updating of ADV's observed state, a suitable action must be determined for every step based on the function of the current state and $Q(s, a, \theta)$. If the action of ADV is taken completely according to the past experience; that is, the ADV chooses the action with the largest corresponding Q value, it is possible to be restricted in the existing experience and unable to find out the new action behavior with larger value; on the other hand, if ADV only focuses on exploring new actions, the majority of actions will be worthless, which leads to a very slow learning speed of Q function. Here, ε-greedy strategy which can makes a good balance between experience and exploration [35] is adopted here to select a suitable action under a specific state.

$$\pi(a|s) = \begin{cases} \text{argmax}_{a'} Q(s, a'), & 1 - \varepsilon \\ \text{randomly select an action from } A, & \varepsilon, \end{cases} \tag{7}$$

where, $\pi(a|s)$ is the action exploration function of ADV; ε represents a small probability, usually smaller than 0.05.

(4) **Buffer Replay.** Each update of ADV's Q function requires a lot of state-action pairs and corresponding instant rewards which can be collected only when ADV interacts with the environment. This leads to sample inefficiency which is a usually criticized problem in deep reinforcement learning. Buffer replay originally proposed by Mnih et al. [34] is adopted here to alleviate this problem and improve the performance of the DQN algorithms. A role of the replay buffer is crucial in terms of accessibility to a variety of data from various time steps, which

makes time-independent learning possible, and it allows the DQN algorithm to learn a robust decision policy

(5) **State, Action and Reward of ADV.**

State. Effective state representation directly affects the performance of the deep reinforcement learning algorithm. In the DQN algorithm, the state is the input of the Q network, which represents the ADV's observation of the surrounding environment. For lane change and car following decision making, an ADV should be able to observe its own state (such as speed and position) and the states of other vehicles within a certain range around it. This research uses an ego-centric reference frame as proposed by Bai et al. [36] to represent the states observed by the ego vehicle. Firstly, each lane of the highway is divided into equidistant cells longitudinally, length of each cell is set as the average car length. In each decision step, taking as the cell occupied by ego ADV as the center point, a span of 10 cells in the longitudinal direction is considered as the observable range of this ego ADV. Given there are 3 lanes in the driving direction of ADV, there are total of 30 cells' states should be referred by ADV to make decision. Each cell's state should be described by whether it is occupied by MDV and the current speed of the occupying MDV (if not occupied by a MDV, the speed of the cell will be set to zero). So, at each step, totally 60 variables (totally 30 cells in censoring range and each cell is described by two variables to indicate whether it is occupied and the speed of occupied vehicle) will be used to represent the surrounding environment state observed by ADV.

Action. The decision-making of ADV includes both lateral and longitudinal actions. The action space of ADV is described in Table 1.

Reward. The design of rewards is crucial to the effectiveness of a reinforcement learning algorithm. In order to encourage high-speed travel and realize complete collision avoidance, reward function should try to balance between travel safety and travel efficiency. Meanwhile, the unconscious violation of the egovehicle during lane change (such as changing from the edge lane to the curb) should also be prohibited. In other words, the criterion for a good decision is that no collision and violation occur. So, totally, reward function proposed in this research is composed of three parts: safety-related reward, efficiency-related reward, and lane change-related reward, which are defined separately in Equation (8)–(11).

Safety-related reward:

$$r_s \begin{cases} 0 & \text{no Collision} \\ -100 & \text{otherwise.} \end{cases} \tag{8}$$

Efficiency-related reward:

Table 1: Action space of ADV.

Action	Description
a_1	Lane change to left
a_2	No lane change (keep current lane)
a_3	Lane change to right
a_4	Acceleration
a_5	Deceleration

$$r_e = l\frac{u_{\text{ego}} - u_{\min}}{u_{\max} - u_{\min}}, \tag{9}$$

where, $u_{\text{ego}}, u_{\max}, u_{\min}$ represents the current speed of ego ADV, the maximum and minimum allowed speed; l is the reward factor.

Lane change-related reward:

$$r_1 = \begin{cases} 1 & \text{lang change success} \\ -1 & \text{otherwise.} \end{cases} \tag{10}$$

Total reward:

$$r_{\text{total}} = \lambda_s r_s + \lambda_e r_e + \lambda_l r_l, \tag{11}$$

where, $\lambda_s, \lambda_e, \lambda_l$ are weight coefficients of different reward components, which can be adjusted to balance between safety and efficiency. Here, $\lambda_s, \lambda_e, \lambda_l$ are set to be 0.5, 0.4, and 0.1, respectively.

3.4. Low-Level Control of MDV and ADV. After receiving the action instruction from the decision-making (car following or lane changing), the low-level controller will control the vehicle accordingly to realize this instruction. Here, nonlinear kinematic bicycle model is used for the simulation of dynamics of both ADV and MDV. The control inputs for the kinematic bicycle model are the front steering angle δ_f and the acceleration a, in which δ_f is calculated by a two-point visual control model of steering [37], and a is calculated by IDM. The description of two-point visual control model can be seen in Figure 3 . The model uses two tangent angles (i.e., θ_n and θ_f in Figure 3) of two reference points in near and far regions to calculate steering angle δ_f, which is described in Equation (12).

$$\delta_f = k_f\theta_f + k_n\theta_n + k_I \int \theta_n \mathrm{d}t, \tag{12}$$

where, k_f, k_n, k_I are the unable parameters of the proportional integration (PI) controller. l_n, l_f are determined by the positions of near and far reference points. When lane change occurs, for empty target lane, l_n, l_f are fixed, while for an occupied target lane, l_n remain fixed but l_f will be the distance between the new leading vehicle and the ego vehicle.

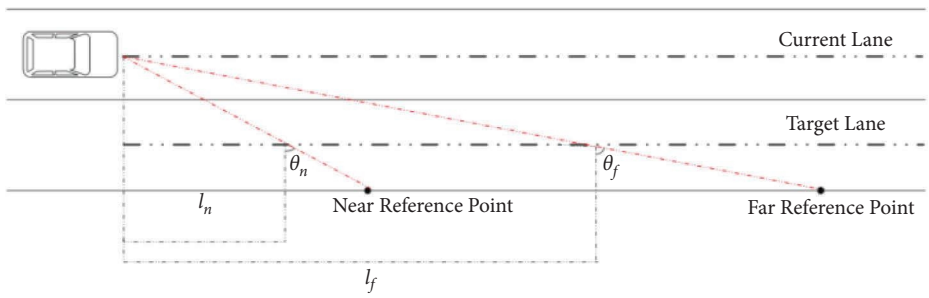


Figure 3: Two-point visual control model.

4. Numerical Experiment and Results

In this section, the proposed DQN-based multilane highway decision-making policy is evaluated by extensive simulation experiments.

4.1. Settings

(1) **Simulation Scenario.** The simulation scenario for evaluating the automatic driving decision-making model based on DQN proposed in this paper is a highway composed of three lanes driving in the same direction, which is shown in Figure 2. The length of this highway is set to 4 km. Once a simulation episode is started, MDV will be continuously generated according to the negative exponential distribution from the leftmost starting point at each lane of the highway, with the average arrival rate of traffic flow λ set as 0.25 veh/s default. By tuning the value of the parameter λ, we can conveniently train and evaluate our proposed decision models under various traffic density conditions. Also, we can assign different values of λ two different lanes, therefore, the imbalance of traffic flow between lanes will be increased, which will potentially trigger more lane-changing needs to better evaluate our proposed model's applicability. The maximum time step of each episode is set to 200 and the time span of each step is set to 1 second, that is within each second, ADV and MDV should make corresponding actions according to the environmental state and their own decision-making models (ADV is driven by DQN-based model while MDV is driven by IDM and MOBIL). One episode will be terminated, and the next episode is started immediately when a collision occurs or the maximum episode duration is arrived.

(2) **Parameters of IDM and MOBIL.** In the simulation experiments, MDVs are driven by IDM and MOBIL for making a longitudinal and lateral decision. The related parameters of IDM and MOBIL are set according to Tables 2 and 3, which are mostly taken from [38].

(3) **Parameters of low-level control model.** In the low-level control layer, the parameters of the two-point visual control model are set according to Table 4.

Table 2: Parameters setting of IDM.

Parameters	Description	Values
$a_{\max}$	Maximum acceleration	0.6m/s^2
$a_{\min}$	Minimum deceleration	−20m/s^2
δ	Acceleration exponent	4
$d_{\min}$	Minimum gap	2 m
T	Safe time headway	1.6 s
b	Desired deceleration	1.7m/s^2
$d_{\max}$	Maximum gap for empty lane	10000 m

Table 3: Parameters setting of MOBIL.

Parameters	Description	Values
b_{safe}	Maximum safe deceleration	−4m/s^2
p	Politeness factor for new following vehicle	1
q	Politeness factor for old following vehicle	0.5
a_{th}	Changing threshold	0.1m/s^2

Table 4: Parameters setting of two-point visual control model.

Parameters	Description	Values
l_n	Distance to near point	5 m
l_f	Distance to far point	100 m
k_f	Proportional gain far point	20
k_n	Proportional gain near point	9
k_I	Integral gain near point	10 s^{-1}

(4) **Hyper-Parameters of DQN-based Decision Model.** We use a fully connected neural network with two hidden layers to realize the Q value function of ADV. The number of neurons in the first and second hidden layer is 128 and 64 and, the number of neurons in the input layer and output layer is 60 and 5, respectively, since each state is represented by a 60-dimension vector and the Q network will output corresponding values for 5 possible actions defined in action sets. The activation functions of hidden layers and output layer are set as RELU and linear, respectively. Also, the best values of other main hyper-parameters are chosen using the tree-structured parzen estimator (TPE) [39] through extensive simulation experiments, results of which are listed in Table 5.

Table 5: Optimal hyper-parameters of DQN-based model.

Hyper-parameters	Best values
Learning rate, α	0.001
Batch size	64
Size of replay buffer	100000
Discount factor, γ	0.9
ε for Greedy exploration	0.01

4.2. DQN-Based Decision Model Performance Analysis. In this section, we show the results about the performance evaluation of the DQN-based decision model of ADV. One metric (i.e., Training Loss) which is used to evaluate the learning performance of the proposed model, and two other metrics (i.e., Average Collision Rate, ACR and Average Episode Reward, AER) which are used to quantify the safety and efficiency of the proposed model are defined as follows:

(1) **Training loss**: The core task of DQN model training is to update the Q-value network according to the Equation (6) step by step with a batch of samples. In Equation (6), the item "$r(s,a) + \gamma Q_k(s', a') - Q_k(s,a)$" reflects the deviation between the estimated Q value and the true Q value. With the increase of training steps, it is expected that this deviation (i.e., Training Loss) to be smaller and smaller, which indicates that the learning of DQN tends to be stable.

(2) **Average Collision Rate (ACR).** ACR is equal to the number of collisions in each episode divided by the total number of decisions made by ADV. The collision counts both rear-end collisions and side-impact collisions. ACR reflects the safety performance of the autonomous driving decision-making model.

(3) **Average Episode Reward (AER).** AER is the total reward obtained in each episode divided by the number of decisions made. AER reflects the comprehensive performance of the autonomous driving decision-making model with respect to safety, efficiency, and lane change success rate.

The loss of the DQN model under 4000 and 65000 training steps are depicted in Figures 4(a) and 4(b) respectively. In Figure 4(a), no significant loss decrease is found, while in Figure 4(b), loss shows a trend of increasing first, then decreasing, and finally stabilizing. This reveals that when the number of training steps reaches enough, the DQN-based decision-making model proposed in this paper can achieve very good training performance.

Further, in order to evaluate the safety and efficiency of our proposed model, the changing curve of ACR and AER with respect to episodes is also depicted in Figures 5(a) and 5(b). It is obvious that although both ACR and ACR show a certain degree of oscillation, their average values tend to decrease and increase steadily. The results show that with the increasing of training steps, the decision-making model based on DQN proposed in this paper can achieve very good results in terms of driving safety and efficiency.

In order to make the changing trend of ACR and AER more clearly to be seen, we used a simple differential filtering method to process the time series values of them, and the results are shown in Figures 6(a) and 6(b).

4.3. Comparative Analysis between DQN and MOBIL&IDM. In this section, in order to further verify the efficiency of our proposed DRL-based model, we conduct simulation experiments to compare the safety and efficiency of our proposed DQN model with rule-based models (i.e., IDM and MOBIL). For ADV, we use DQN-based decision-making model and IDM combining with MOBIL to drive them for extensive simulation experiments separately, the AERs recorded are shown in Figure 7. It can be seen that MOBIL has a high average reward in the initial stage of the experiment, but with the increasing of training steps, DQN reaches an average reward higher than MOBIL by about 10% after full convergence, which means that DQN can do better in this multilane highway environment where exists dynamic and complex interactions between ADV and MDVs.

4.4. Other Variants of DQN-Based Decision Model. In this section, we further try other variants of the DQN model (i.e., DDQN (Double DQN) and Dueling DQN) to depict ADV's decision-making behavior. DDQN was proposed as a specific adaptation to the DQN algorithm to reduce the observed overestimations [40], while Dueling DQN uses a different network architecture with what is used in DQN to separate the estimation of the state value function and the state-dependent action advantage function [41]. Both DDQN and Dueling DQN are considered could improve the performance of DQN in some extent, so they are attempted to model the decision making of ADV, and the performance comparison between them and DQN are conducted separately.

4.4.1. DQN vs. DDQN. We systematically compare the model performance between DQN and DDQN with respect to ACR, AER, loss and Q value, results are shown in Figures 8(a)–8(d) respectively.

Figure 8 shows that in respect to the decision accuracy and the number of convergence episodes, the two algorithms show relatively similar learning efficiency (DDQN is faster in the early stage and DQN catches up in the later stage), and DDQN has more stable oscillation than DQN in terms of AER, ACR, and network loss, while the network loss and ACR are somewhat lower than DQN.

The Q value of DDQN is significantly lower than that of DQN, and it can be seen that after optimization by DDQN, the decision of the agent tends to be more conservative, which can theoretically have a higher decision accuracy in the application process.

4.4.2. DQN vs. Dueling DQN. The second improvement of DQN is the modification of its network structure. Both DQN and DDQN are single-branch network structures, and the improved Dueling DQN is a dyadic network structure. With

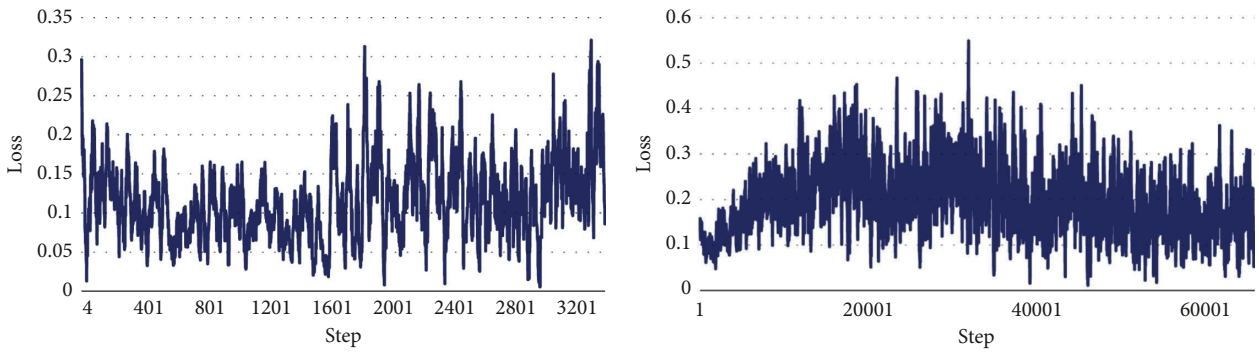


Figure 4: Training loss of DQN-based decision model of ADV.(a) 4000 steps. (b) 65000 steps.

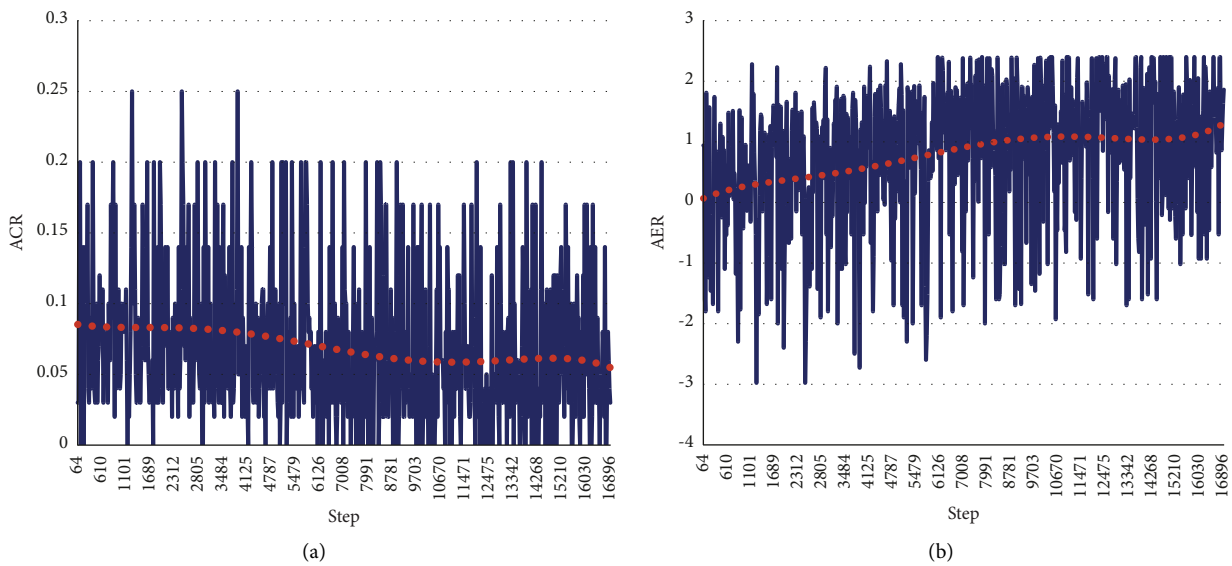


Figure 5: ACR and AER with respect to training steps.(a) ACR. (b) AER.

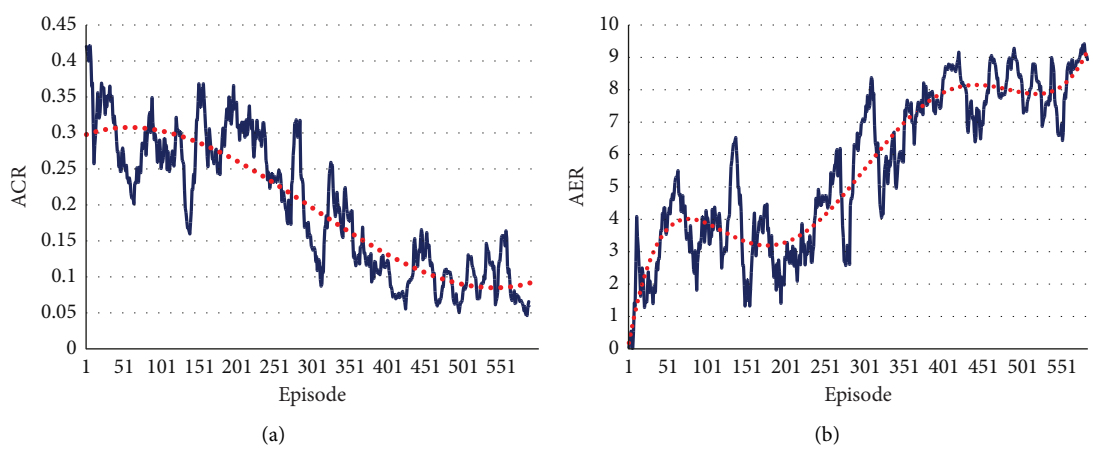


Figure 6: ACR and AER with respect to episodes after processing with differential filtering.(a) ACR. (b) AER.

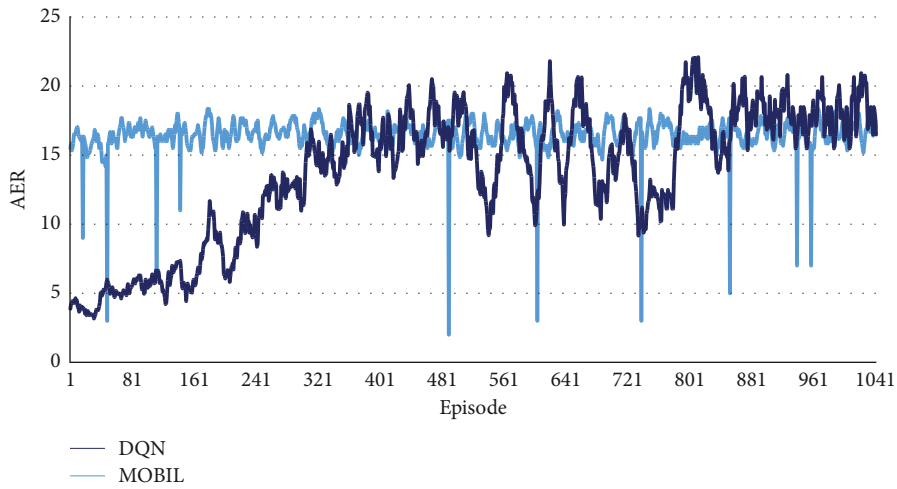


Figure 7: Comparison of AER between DQN and MOBIL&IDM.

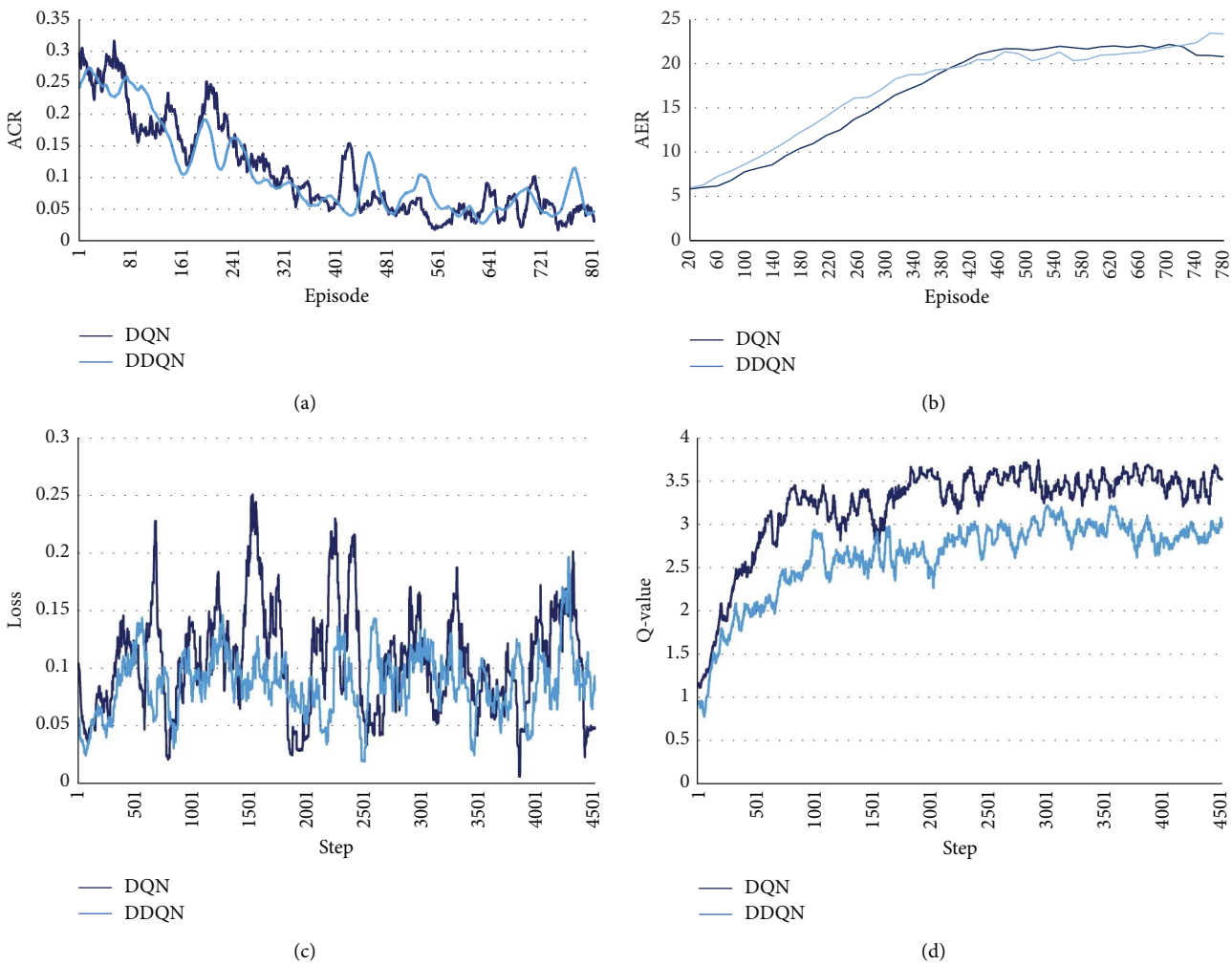


Figure 8: Performance comparison between DQN and DDQN. (a) ACR. (b) AER.(c) Loss. (d) Q value.

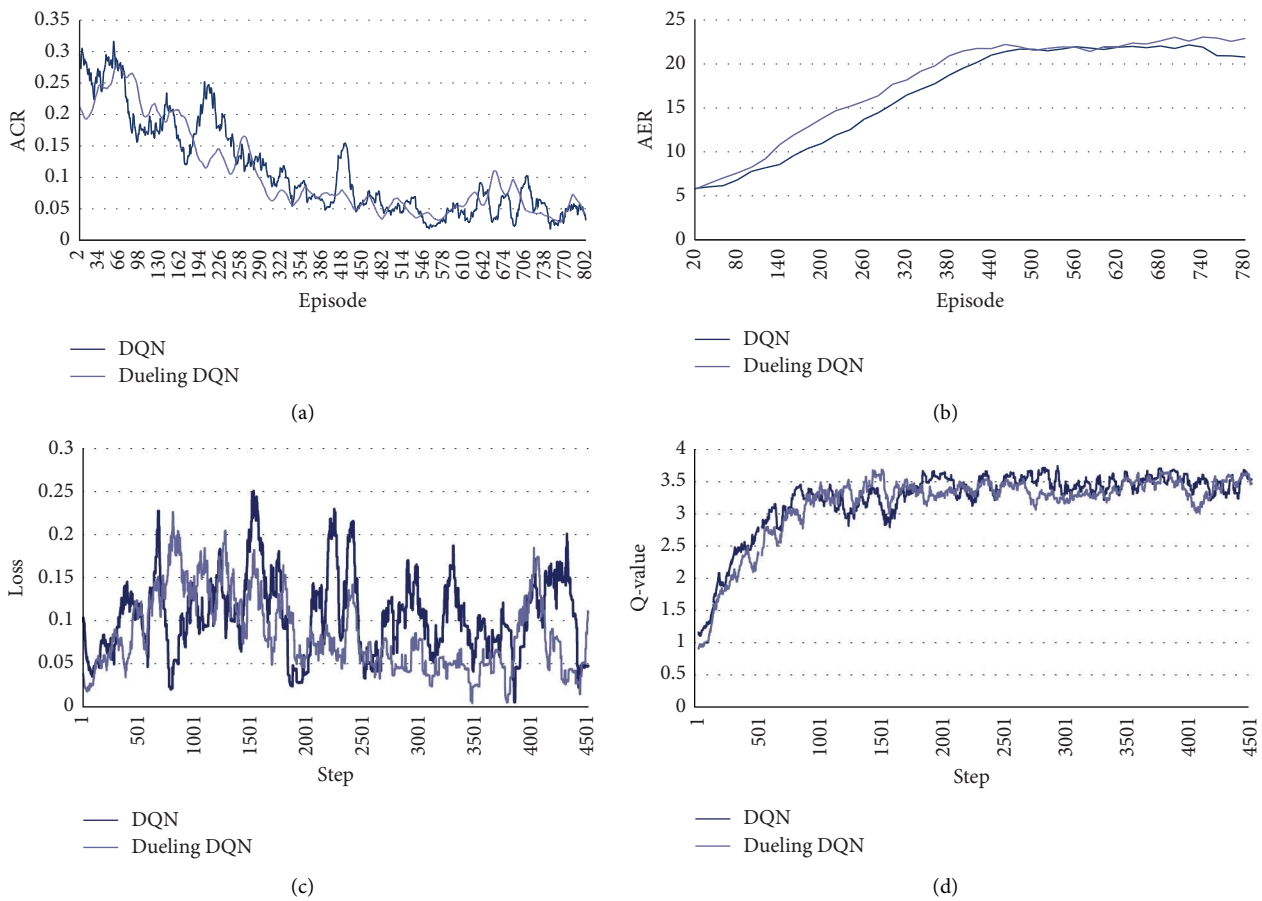


Figure 9: Performance comparison between DQN and Dueling DQN. (a) ACR. (b) AER.(c) Loss. (d) Q-value.

the unchanged input, the output of Dueling DQN will go through two fully connected layer branches corresponding to state values and dominance values, updating the scores of all actions in each iteration, instead of just taking the maximum value as in DQN. The algorithm can increase the convergence speed to some extent under the influence of different network structures. Dueling DQN is also compared with DQN in simulation experiments with respect to ACR, AER, loss and Q value, and experimental results are shown in Figures 9(a)–9(d).

Figure 9 shows that for ACR and AER, Dueling DQN converges almost 10% faster than DQN, can have less oscillation performance in a short time, the loss of Dueling DQN is smaller than DQN, and Q value is comparable to DQN. Overall, under the same hyper-parameter configuration (e.g., learning rate), Dueling DQN can indeed perform better than DQN in terms of ACR, AER, and loss.

4.5. Performance Analysis of DQN-Series Models with Different ADV Penetration. In the previous sections, the considered decision-making scenarios of automatic driving on multilane highway are all mixed travel of a single ADV and multiple MDVs. In this section, we want to investigate the performance of DQN, DDQN and Dueling DQN-based models under different ADV penetration (i.e., the proportion of ADV in all travel vehicles) with respect to episodes needed to converge (i.e., convergence episode). Results about a number of convergence episodes of DQN, DDQN, and Dueling DQN-based models are shown in Table 6.

Table 6: Comparison of convergence episodes with different ADV penetration.

ADV: MDV	DQN	DDQN	Dueling DQN
1 : 100	823	792	691
1 : 50	1767	1743	1531
1 : 30	>2000	>2000	1958
1 : 10	>2000	>2000	>2000

In general, as the number of ADVs increases, the deep reinforcement learning algorithm (i.e., DQN, DDQN, and Dueling DQN) learns and masters the state of the environment more and more difficult, and the average convergence episodes gradually grows, and even fails to converge in finite time (i.e., convergence episode >2000). DQN and DDQN comparing with Dueling DQN converge more slowly. The superior performance of Dueling DQN is attributed to its optimized network structure based on DQN. DDQN optimizes the update logic of DQN and is able to acquire higher Q value, but it does not produce a significant

advantage over DQN in the selection of discrete behaviors, such as vehicle lane change decisions, so the performance improvement is limited. In general, due to the increase of ADV, the state faced by each ADV in the mixed travel environment is more complex, dynamic and in essence nonstationary, it is difficult for ADV to learn a stable policy for decision making and consequently leads to much more convergence episodes needed. Actually, when ADV increasing, multi-agent reinforcement learning [42] can be a good choice to model their collective decision-making behaviors, which may be our research direction to be explored in the future.

5. Conclusions

This paper proposes a reinforcement learning-based decision-making model for autonomous driving on a multilane highway with mixed traffic composed of ADV and MDV. By proper state representation, action set definition and reward function design, DQN, DDQN, and Dueling DQN-based models are developed for automatic making of both lateral and longitudinal decisions. At the same time, in order to construct the simulation environment of mixed traffic, we describe in detail the rule-based decision behavior models (i.e., IDM and MOBIL) which are used to generated decision for MDV vehicles. Further, low-level control of both ADV and MDV is realized by a nonlinear kinematic bicycle model combining with a two-point visual control model.

Through extensive simulation experiments, the safety and efficiency for autonomous driving decision making by DQN, DDQN, and Dueling DQN is verified. Comparing the experimental results of DQN and its variant models with the rule-based decision-making model, it is found that, deep reinforcement learning-based models for decision making of autonomous driving are generally superior to rule-based methods with respect to safety, efficiency, and generalization ability. It is also found, with the increasing of ADV penetration in mixed traffic flow, the training and generalization of DRL-based models becomes more and more difficult, therefore, multi-agent reinforcement learning, through joint consideration of environmental observation and collective decision-making of ADV vehicles, may be an important research direction in the future.

Data Availability

All data and code generated in our study are available at zhaoboyuan825/An-Integrated-Lateral-and-Longitudinal-Decision-Making-Model: code of model and simulation (github.com).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Authors' Contributions

The authors confirm contribution to the paper as follows: study conception and design: Jianxun Cui and Boyuan Zhao; experiments setup: Jianxun Cui and Boyuan Zhao; analysis of results: Jianxun Cui and Boyuan Zhao; draft manuscript preparation and revision: Jianxun Cui, Boyuan Zhao and Mingcheng Qu.

Acknowledgments

This research was supported by the joint guidance project of Heilongjiang Provincial Natural Science Foundation through Grant #LH2021E074 and the http://dx.doi.org/10.13039/501100012226Fundamental Research Funds for the Central Universities through Grant #HIT.NSRIF202235.

References

[1] Z. Qiao, K. Muelling, J. M. Dolan, P. Palanisamy, and P. Mudalige, "Automatically generated curriculum based reinforcement learning for autonomous vehicles in urban environment," in *Proceedings of the 29th IEEE Intelligent Vehicles Symposium (IV 2018)*, Changshu, China, June 2018.

[2] B. R. Kiran, I. Sobh, V. Talpaert et al., "Deep reinforcement learning for autonomous driving: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 1–18, 2021.

[3] H. Ahn, K. Berntorp, P. Inani, A. J. Ram, and S. Di Cairano, "Reachability-based decision-making for autonomous driving: theory and experiments," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 5, pp. 1–15, 2020.

[4] S. Zhao, Q. Hou, and Y. Zhai, "Decision mechanism of vehicle autonomous lane change based on rough set theory," in *Proceedings of the ICCIR 2021: 2021 International Conference on Control and Intelligent Robotics*, New York NY USA, June 2021.

[5] X. Liu, X. Qu, and X. Ma, "Improving flex-route transit services with modular autonomous vehicles," *Transportation Research Part E Logistics and Transportation Review*, vol. 149, p. 102331, 2021.

[6] Y. Liu, X. Wang, L. Li, S. Cheng, and Z. Chen, "A novel lane change decision-making model of autonomous vehicle based on support vector machine," *IEEE Access*, vol. 7, pp. 26543–26550, 2019.

[7] T. Yin, Y. Li, J. Fan, and Y. A. Shi, "A novel gated recurrent unit network based on svm and moth-flame optimization algorithm for behavior decision-making of autonomous vehicles," *IEEE Access*, vol. 9, p. 99, 2021.

[8] H. Chao, M. Chen, H. Peng, and Y. Xing, "Toward safe and personalized autonomous driving: decision-making and motion control with dpf and cdt techniques," *IEEE*, vol. 26, no. 2, p. 99, 2021.

[9] L. Xin, X. Xin, and Z. Lei, "Reinforcement learning based overtaking decision-making for highway autonomous driving," in *Proceedings of the 6th International Conference on Intelligent Control & Information Processing*, November 2015.

[10] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "Deep reinforcement learning framework for autonomous driving," *Electronic Imaging*, vol. 29, pp. 70–76, 2017.

[11] F. Ye, X. Cheng, P. Wang, C. Y. Chan, and J. Zhang, "Automated lane change strategy using proximal policy optimization-based deep reinforcement learning," in *Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV)*, Las Vegas, NV, USA, February 2020.

[12] H. Wang, H. Gao, S. Yuan et al., "Interpretable decision-making for autonomous vehicles at highway on-ramps with

latent space reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, p. 99, 2021.

[13] M. J. Shin and J. Kim, "Randomized adversarial imitation learning for autonomous driving," 2019, https://arxiv.org/abs/1905.05637.

[14] J. Chen, B. Yuan, and M. Tomizuka, "Deep imitation learning for autonomous driving in generic urban scenarios with enhanced safety," in *Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, November 2019.

[15] A. Jamgochian, E. Buehrle, J. Fischer, and M. J. Kochenderfer, "Safety-aware hierarchical adversarial imitation learning for autonomous driving in urban environments," 2022, https://arxiv.org/pdf/2204.01922.pdf.

[16] A. Kesting, M. Treiber, and D. Helbing, "General lane-changing model mobil for car-following models," *Transportation Research Record*, vol. 1999, pp. 86–94, 2007.

[17] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review*, vol. 62, no. 2, pp. 1805–1824, 2000.

[18] P. Wang, C. Y. Chan, and A. Fortelle, "A reinforcement learning based approach for automated lane change maneuvers," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, Changshu, China, June 2018.

[19] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, Springer, Berlin, Germany, 2009.

[20] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[21] X. Ma, H. Zhong, Y. Li, J. Ma, Z. Cui, and Y. Wang, "Forecasting transportation network speed using deep capsule networks with nested LSTM models," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 4813–4824, 2021.

[22] C. Vallon, Z. Ercan, A. Carvalho, and F. Borrelli, "A machine learning approach for personalized autonomous lane change initiation and control," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, Los Angeles, CA, USA, June 2017.

[23] H. Bi, T. Mao, Z. Wang, and Z. Deng, "A data-driven model for lane changing in traffic simulation," in *Proceedings of the ACM SIGGRAPH Symposium on Computer Animation*, Goslar, Germany, June 2016.

[24] D. C. K. Ngai and N. H. C. Yung, "Automated vehicle overtaking based on a multiple-goal reinforcement learning framework," in *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference Seattle*, pp. 818–823, Bellevue, WA, USA, September 2007.

[25] P. Wang and C. Y. Chan, "Formulation of deep reinforcement learning architecture toward autonomous driving for on-ramp merge," in *Proceedings of the IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, Japan, October 2017.

[26] C. J. Hoel, K. Wolff, and L. Laine, "Automated speed and lane change decision making using deep reinforcement learning," in *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI, USA, November 2018.

[27] D. Isele, R. Rahimi, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating occluded intersections with autonomous vehicles using deep reinforcement learning," in *Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, May 2018.

[28] L. Xin, S. E. Li, P. Wang et al., "Accelerated inverse reinforcement learning with randomly pre-sampled policies for autonomous driving reward design," in *Proceedings of the IEEE Intelligent Transportation Systems Conference ITSC*, Auckland, New Zealand, October 2019.

[29] Z. Wu, L. Sun, W. Zhan, C. Yang, and M. Tomizuka, "Efficient sampling-based maximum entropy inverse reinforcement learning with application to autonomous driving," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5355–5362, 2020.

[30] J. Ho and S. Ermon, "Generative adversarial imitation learning," pp. 4565–4573, 2016, https://arxiv.org/abs/1606.03476.

[31] C. Finn, S. Levine, and P. Abbeel, "Guided cost learning: deep inverse optimal control via policy optimization," pp. 49–58, 2016, https://arxiv.org/abs/1603.00448.

[32] J. Fu, K. Luo, and S. Levine, "Learning robust rewards with adversarial inverse reinforcement learning," 2017, https://arxiv.org/abs/1710.11248.

[33] S. Reddy, A. D. Dragan, and S. Levine, "Imitation learning via reinforcement learning with sparse rewards," 2019, https://arxiv.org/abs/1905.11108.

[34] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Playing atari with deep reinforcement learning," 2013, https://arxiv.org/abs/1312.5602.

[35] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Proceedings of the IEEE International Conference on Robotics & Automation*, New York NY USA, May 2017.

[36] Z. Bai, W. Shangguan, B. Cai, and L. Chai, "Deep reinforcement learning based high-level driving behavior decision-making model in heterogeneous traffic," in *Proceedings of the Chinese Control Conference (CCC)*, Guangzhou, China, July 2019.

[37] D. D. Salvucci and R. Gray, "A two-point visual control model of steering," *Perception*, vol. 33, no. 10, pp. 1233–1248, 2004.

[38] A. Alizadeh, M. Moghadam, Y. Bicer, N. K. Ure, U. Yavas, and C. Kurtulus, "Automated lane change decision making using deep reinforcement learning in dynamic and uncertain highway environment," in *Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, October 2019.

[39] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kegl, "Algorithms for hyper-parameter optimization," Proceedings of the 24th International Conference on Neural Information Processing Systems, Red Hook, NY, USA, pp. 2546-2554, December 2011.

[40] V. H. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proceedings of the National Conference on Artificial Intelligence AAAI Press*, New York NY USA, February 2016.

[41] Z. Y. Wang, N. Freitas, and M. Lanctot, "Dueling network architectures for deep reinforcement learning," 2015, https://arxiv.org/abs/1511.06581.

[42] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 156–172, 2008.

Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 5725009, 9 pages
https://doi.org/10.1155/2023/5725009

Research Article

Hierarchical Vehicle Scheduling Research on Tide Bicycle-Sharing Traffic of Autonomous Transportation Systems

Mai Hao,[1,2] **Ming Cai**,[1,2] **Minghui Fang**,[1,2] **and Shuxin Jin**[1,2]

[1]*School of Intelligent Systems Engineering, Sun Yat-Sen University, Shenzhen 518107, China*
[2]*Guangdong Key Laboratory of Intelligent Transport Systems, Guangzhou 510006, China*

Correspondence should be addressed to Shuxin Jin; jinshx3@mail.sysu.edu.cn

Received 4 July 2022; Revised 7 September 2022; Accepted 7 February 2023; Published 18 February 2023

Academic Editor: Alessandro Severino

To facilitate the intelligent and automated provision of mobility services by autonomous transportation systems, bike-sharing can be a supplement to public transport for addressing their point-to-point issue, namely, "last mile" service. However, according to the different nature of land use, the uneven spatio-temporal distribution of travel demand can directly lead to difficult access to bikes with high travel costs for users and operating costs for operators. Based on this, this paper analyzes the user behavior patterns within different areas by using GeoHash encoding and proposes a hierarchical autonomous vehicle scheduling model based on tide bicycle-sharing traffic, namely, *HATB*. It minimizes operating costs and maximizes user satisfaction to dynamically optimize scheduling routes and required vehicles within each layered zone. As for extracting historical orders of Mobike in Beijing, an example analysis through the genetic algorithm of *HATB* is conducted to support effective and efficient scheduling. Compared to existing scheduling methods, *HATB* has faster convergence and lower time complexity, which improves bike turnaround efficiency and practical application ability, thus making it more convenient for the public to travel during peak hours.

1. Introduction

Nowadays, as the modern transportation systems (TSs) develop, problems among mobility services (MSs), e.g., congestion, route adjustment, user dispersion, and peak-time conflicts have become commonplace [1–3] in terms of the adjustment of urban planning and the year-on-year increase in car ownership. Since the MS are fundamental in propelling current intelligent TS (ITS) evolving towards autonomous TS (ATS), they are also being renovated to assist the public on a daily basis and explore the advancement of ATS. Some theoretical development has revealed that the creation of shared transport, called bicycle-sharing, has effectively alleviated the "last mile" service (LMS), which is the most predominant pain point in the current MS [4] in terms of the most immediate interaction with users, and some emerging companies, e.g., Mobike have also taken this trend to a new level. However, bikes must be parked in GPS-identified areas to address ill-posed problems in the case of illegal parking, vandalism, or theft. Since then, to avoid constant billing, users never consider the capacity of the parking area, resulting in an uneven spatial distribution of bikes, namely, some areas suffer a severe accumulation of bikes while others are "one bike is hard to find" [5]. Therefore, scientific and reasonable scheduling strategies are required to overcome the imbalance between the supply and demand of bikes and improve resource utilization.

Rebalancing and optimizing bicycle-sharing distribution constitutes the vehicle routing problem (VRP), and most current research is based on this theory. For instance, Caggiani et al. [6] proposed a decision support system for the reallocation problem by forecasting the demand for spatio-temporal bikes. Similar research can, accordingly, be divided into static and dynamic scheduling to optimize VRP models with different objectives. Specifically, in static scheduling, Kadri et al. [7] and Dell'Amico et al. [8] developed optimization models for user satisfaction and operating cost, respectively. Yan et al. [9] investigated the deterministic and stochastic demand for bicycle-sharing in dynamic scheduling. In general, these methods assume that the overall

supply and demand within a scheduling station are in equilibrium without supporting the mobility of bikes between zones. Moreover, limited open literature has reported that current research focuses too much on mathematical modeling, neglecting the analysis of actual demands. Therefore, these methods have encountered three challenges in practice, namely, slow convergence, high time complexity, and problematic application, from perspectives of fulfilling the actual demands in real-time and adjusting scheduling as needed.

As current diversified mobility demands tend to be managed and fulfilled by more intelligent and automated systems with fewer human intervention [10], it is an urgent need to collaborate corresponding functions to renovate the conventional MS of ITS in the context of ATS [11], i.e., update the ability to sense user demands and rearrange system supplies [12, 13]. Hence, to promote the MS provided by ATS, this paper proposes a hierarchical autonomous vehicle scheduling model based on tide bicycle-sharing traffic, namely, *HATB*. This model uses GeoHash coding to divide the scheduling into three layers, i.e., top, middle, and bottom, corresponding to the scheduling terminus, area, and point. Based on the genetic algorithm (GA), the model can achieve hierarchical and dynamic scheduling of vehicles and routes to maximize user satisfaction, while minimizing operating costs.

Furthermore, in contrast to current studies on scheduling bikes in ITS, the *HATB* makes three main contributions to optimizing convergence speed, time complexity, and application difficulties of actual scheduling. The scheduling results based on actual orders ultimately demonstrate *HATB* can provide a rational reference for LMS in ATS and guide the development of bicycle-sharing regulation and operation.

The overall structure of this paper is divided into five sections. Section 2 introduces related solutions and emerging challenges. The methodology relevant to *HATB* is described in Section 3. Section 4 elaborates on the scheduling results and superiority of the model. Finally, Section 5 summarizes this study and sketches future research directions.

1.1. Related Works. In the transport domain, LMS refers to the direct interaction between the end of public transport and users, which often suffers from scattered users, peak-time conflicts, and uneven distribution. As an effective way to cope with the LMS problems, bicycle-sharing has become a non-negligible component of urban transport. For example, Cheng et al. [14] have demonstrated that bicycle-sharing increases the proportion of green transport in cities and solves the low efficiency at the end of the travel chain.

In general, current research on bike-sharing mainly focuses on its development status and travel characteristics, but few on its scheduling. Researchers like Soriguera and Jiménez-Meroño [15], Gimon [16], and Lu et al. [17] concur that even while bicycle-sharing has considerable quantities, the spatio-temporal differences in user demands, no fixed parking area, and fewer available bikes may lead to a more significant overall imbalance. Therefore, it is vital to take effective scheduling strategies to rebalance and optimize the distribution of bikes, thereby addressing difficulties in management and operation. In this context, scheduling bicycle-sharing can be regarded as a heuristic algorithm-based, e.g., GA, ant colony algorithm (ACO), and vehicle routing problem (VRP) [18–20], which can be classified as static or dynamic scheduling according to different strategies and objectives.

Dynamic scheduling mainly focuses on peak time and relies on user demands. For instance, a mathematical model for dynamic scheduling is created by Zhang et al. [21] based on the parking area's actual capacity and users' predicted arrival times. Shui and Szeto [22] partition peak time to optimize scheduling routes by regarding scheduling in each time interval as static scheduling. Chiariotti et al. [23] propose that scheduling bikes can dynamically determine the scheduling time through historical orders. In general, dynamic scheduling lessens operating costs' impact on operators by prescheduling bikes to avoid a shortage occurs. However, based on the uncertain use of bikes, frequent scheduling with complex constraints is necessary, which may lead to higher operating costs and slower convergence, thus making it challenging to fulfill user demands in real-time.

Another more common scheduling strategy is static scheduling during off-peak time. For example, Lang [24] provides a multiwarehouse model based on the Tabu search algorithm to minimize scheduling distance and improve scheduling efficiency and robustness. Bae and Moon [25] use a dual time window with customer service levels to reduce total transport and labor costs. Since static scheduling only considers the predicted demands of stations, increasing more bikes for stations to guarantee user demands means that the time complexity of the heuristic algorithm grows exponentially. Moreover, to fulfill the actual demands, the allocated bikes by these studies may exceed the station's capacity.

Besides, such above-given studies are mainly applied to typical scenarios, as presented in Figure 1, where a single scheduling station serves one zone and only the routes within the scheduling zone are considered. It is often limited in actual scheduling by the service range of the station, which needs to frequently adjust the boundaries of this scheduling zone, thus leading to some research on hierarchical scheduling strategies. By defining scheduling priorities based on demand intensity, Sakakibara et al. [26] and Ni et al. [27] highlight the feasibility and reliability of hierarchical scheduling. In order to illustrate the flexibility, Zhang [28] and Ma et al. [29] set stations with similar demands in the same layer in accordance with the spatio-temporal characteristics of bikes. However, the definition of hierarchies in these methods is too subjective and not clear, making it difficult to implement in practice.

In summary, whilst a considerable body of research has been carried out on VRP, much less fits the spatial-temporal and cross-regional mobility characteristics of bicycle-sharing. In addition, it seems to be a common problem that existing studies focus more on mathematical modeling

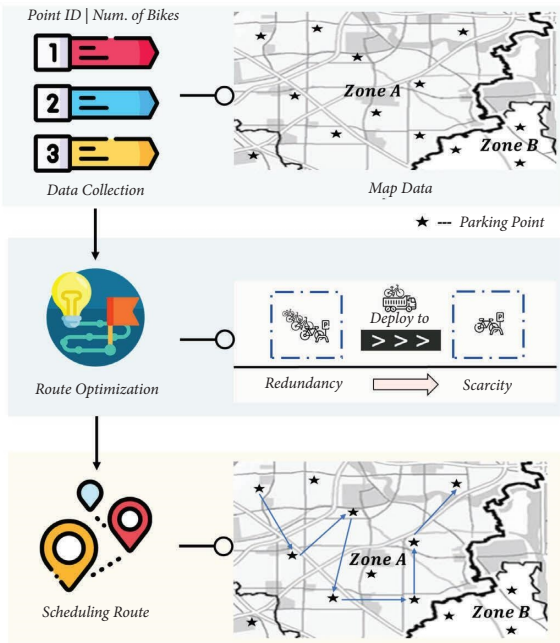


Figure 1: The schema of existing scheduling (map data (part of Beijing): https://www.beijing.gov.cn).

but neglect the analysis of actual demands. When scheduling according to the above-given methods, three challenges in terms of the problematic application in practice, namely, the high time complexity of models and slow convergence for algorithms, are increasingly apparent.

Particularly driven by diverse and emerging technologies and demands, ITS is evolving into ATS, which illustrates that MS should be autonomously fulfilled and managed by more intelligent systems with fewer human intervention [30]. Therefore, it is an urgent need to study and improve the monolithic strategies and rationalize the actual demands to achieve hierarchical and autonomous scheduling of bicycle-sharing, thus rationally guiding the provision of the LMS by ATS.

2. The *HATB* Methodology

This section uses three subsections to present the framework, hypotheses, and construction of the proposed *HATB*.

2.1. Model Framework. As existing scheduling methods, in general, require frequently adjusting boundaries and increasing vehicles to alleviate the difference between bikes' supply and demand, which may inevitably increase transport time and costs; the scheduling framework can be set up as a hierarchical scheduling structure, i.e., a top-middle-bottom hierarchy of scheduling terminus-area-point.

However, since current hierarchical methods have a subjective definition of their hierarchies, geocode can ensure the objective; e.g., what3words [31] uses fixed $3m \times 3m$ squares to divide the earth, and pluscode [32] represents each latitude and longitude level by 2-bit code, whose length range in levels 1 to 3 jumps from $110km$ and $5.5km$ to $275m$. These geocodes, accordingly, have good accuracy but loss flexibility; they may not meet the actual scheduling requirements.

Therefore, GeoHash encoding, proposed by Morton [33], can be used to better support effective and efficient scheduling. Its maximum length of 12 bits can represent a geographic location with arbitrary precision. For example, the GeoHash strings WX4ER and WX4G2 represent two regions of Beijing (China), where each character is a certain rectangular area. Moreover, the order information (Data Sources: https://biendata.xyz/competition/mobike/) on bicycle-sharing, as extracted in Table 1, also indicates the feasibility of dividing the scheduling layer via GeoHash.

The coding definition, as described in Table 1, illustrates that the 7-bit string matches the characteristics of actual bike stops, namely, area size, and the 5-bit string suits for vehicles to dispatch bikes in light of their loading capacity, i.e., 400 bikes. Therefore, the overall framework of the proposed hierarchical scheduling model, called *HATB*, can be obtained as presented in Figure 2. In general, this framework is characterized by a number of scheduling areas in each of the three layers, namely, bike stops consist of the bottom layer of scheduling, while the top and middle layers likewise have demands and capacity restrictions for bikes, and hence, the scheduling within the same layers is regional scheduling for seeking optimization.

2.2. Model Hypotheses. Considering the complexity of actual scheduling, the proposed *HATB* in this paper defines the following hypotheses and the frequency of scheduling as once in the morning peak and once in the evening peak, respectively.

(1) All scheduling vehicles own the same attributes
(2) In each scheduling route, the vehicle departs from one scheduling terminus (area) and returns to this place after deploying bikes to corresponding areas (points) contained
(3) Fuel consumption and vehicle loss should be considered
(4) Each scheduling area can only be served once
(5) The actual orders determine the scheduling demand
(6) All scheduling tasks are required to be completed within the specified scheduling cycle
(7) The scheduling areas and points have sufficient space to accommodate the bikes deployed in or out during a scheduling cycle

2.3. Model Construction. Based on the above-given hypotheses and the actual operations of bike-sharing, considering only the operating costs will gradually lose customers, and weighing only user satisfaction runs counter to the essence of business profitability. Hence combining these two factors, this paper constructs a regional scheduling model for bikes to minimize operating costs (F_1) and maximize user satisfaction (F_2).

Table 1: The order information of corresponding GeoHash string length.

GeoHash string length	5-bit	6-bit	7-bit
Average orders	436.994788	28.214119	3.689289
Area width	4.89 km	1.22 km	153 m
Area height	4.89 km	0.61 km	153 m

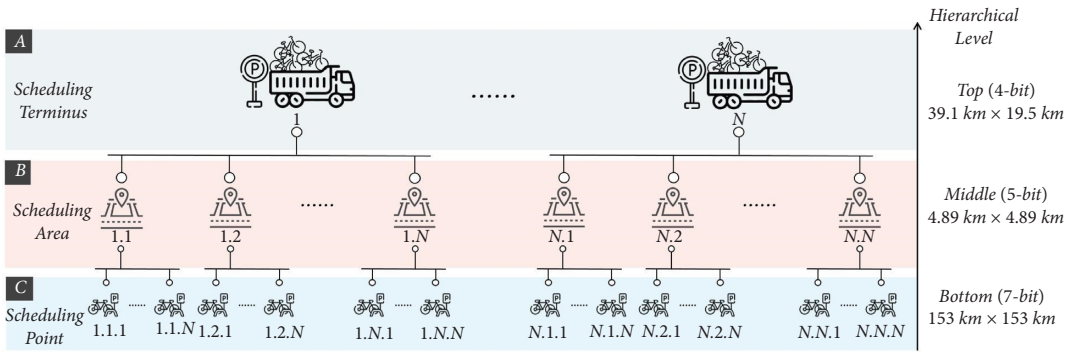


Figure 2: The framework of *HATB*.

$$\begin{aligned} &\text{min:} \quad \partial F_1 + \mu F_2, \\ &s.\text{t.} \quad \partial + \mu = 1 \ \partial, \mu > 0. \end{aligned} \tag{1}$$

The parameters ∂ and μ indicate that bike-sharing operators need to adjust the weight coefficients of operating and penalty costs according to their own emphasis.

2.3.1. $\mathbf{F}_1$: *The Objective of Minimizing Operating Costs.* The actual operation costs need to consider both fixed and flexible costs, as summarized in equation (2), which is determined jointly by the value of scheduling vehicles [34], the unit transport cost (i.e., vehicle loss: 1 CNY/*km*, fuel consumption: 1 CNY/*km*, and labor cost: 100 CNY/person), and the scheduling distance.

$$\begin{aligned} \min F_1 &= \min(Fx\ d + Flx) \\ &= \min\left(\sum_{w}^{W}\sum_{t}^{T}\sum_{i}^{I} x_{wj}^{t} C_t + \sum_{t}^{T}\sum_{i}^{R}\sum_{j}^{R} x_{ij}^{t} d_{ij} M_t\right), \end{aligned} \tag{2}$$

$$s.\text{t.} \quad \sum_{i}^{I} x_{wi}^{t} = \sum_{i}^{I} x_{wi}^{t} \le 1 \quad \forall w \in W,\ t \in T, \tag{3}$$

$$\sum_{t}^{T}\sum_{\substack{j=1 \\ j\neq i}}^{R} x_{ji}^{t} = \sum_{t}^{T}\sum_{\substack{j=1 \\ j\neq i}}^{R} x_{ji}^{t} = 1 \quad \forall i, j \in I,\ t \in T, \tag{4}$$

$$0 \le \sum_{i}^{R}\sum_{\substack{j=1 \\ j\neq i}}^{R} x_{ij}^{t} d_{ij} \le L_t \quad \forall t \in T, \tag{5}$$

$$0 \le a_{ij}^{t} \le a_{ij}^{t} x_{ij}^{t} \quad \forall i,\ j \in R,\ i \neq j,\ t \in T, \tag{6}$$

$$x_{ij}^{t} = 0 \text{ or } 1 \quad \forall i,\ j \in R,\ i \neq j,\ t \in T, \tag{7}$$

$$a_{ij}^{t} \in Z^{+} \quad \forall i,\ j \in R,\ i \neq j,\ t \in T, \tag{8}$$

Table 2: The meaning of parameters for formulations 1–14.

Parameter	Meaning
I	Scheduling area
T	Scheduling vehicles
W	Scheduling terminus
R	Vehicle pools for scheduling terminus and areas
a_t	Maximum capacity of vehicle t
a_{ij}^t	Loading bikes of vehicle t from area i to area j
L_t	Maximum operation distance
C_t	Fixed cost per vehicle
M_t	Flexible cost per vehicle
V	Average speed
d_{ij}	Transport distance by vehicle from area i to area j
GAP_i	Number of vehicles to be deployed in/out of area i
ω_i	Importance of area i
TM_i	Time for vehicle to arrive in area i
S	Time cost for loading/unloading bikes
TM_w^t	Time for vehicle to depart from terminus
$[FQ_i, FR_i]$	Penalty time window

Table 3: Examples of user travel characteristics.

Travel information	Time	Orders
Average distance: 815 m	7 a.m	189578
	18 p.m	173654
	8 a.m	171011
Median distance: 660 m	17 p.m	164126
	19 p.m	125383
	12 p.m	119883

Equation (3) indicates that this scheduling vehicle starts and ends at the terminus; equation (4) suggests that each area can only be served once; equation (5) shows that the transport distance must not exceed the maximum scheduling distance; equation (6) points that the number of bikes loaded by the vehicle must not exceed its maximum capacity, namely, 400; equation (7) means x_{ij}^t as 0-1 variable; equation (8) proves that the number of bikes deployed by the vehicle is a non-negative integer.

2.3.2. $\mathbf{F}_2$*: The Objective of Maximizing User Satisfaction.* User satisfaction can be improved by adding time window constraints, as described in equation (9), which means maximizing user satisfaction can equivalently transfer into minimizing the penalty cost of scheduling timeout.

$$\max F_2 = \min \sum_i^I \omega_i \rho_i(t_i), \tag{9}$$

$$s.\text{t.} \quad TM_w^t = 0 \quad \forall t \in T, \tag{10}$$

$$TM_j = \sum_t^T \sum_i^R x_{ij}^t \left(TM_i + TM_j + S \cdot \mathrm{GAP}_i\right), \quad \forall j \in I, \tag{11}$$

$$TM_i + TM_j + S \cdot \mathrm{GAP}_i - K\left(1 - x_{ij}^t\right) \le TM_j, \tag{12}$$

$$FQ_i \le TM_i \le FR_i, \quad \forall i \in I. \tag{13}$$

Equation (10) indicates that the vehicle departs from the terminus at time zero; equation (11) means that the time calculation for a vehicle to arrive in an area; equation (12) demonstrates that the scheduling cannot arrive in area j from area i before $TM_i + TM_j + S \cdot \mathrm{GAP}_i - K(1 - x_{ij}^t)$; equation (13) represents that vehicle needs to arrive in an area within the time window.

The meaning of the parameters in the above-given equationsare shown in Table 2.

3. Case Study

The highlights of *HATB* solving are illustrated in terms of algorithm settings, scheduling results, and model evaluation in this section.

```
Input: SchedulingArea SA, DistanceMatrix DM
Output: SchedulingRoute SR, SchedulingVehicle SV, DeployedBikes DR
Initialization: Generation Gen = 100, CrossoverRate CR = 0.8, MutationRate MR = 0.2, Population Pop ← ∅, Chromosome
CH ← ∅, Fitness Fit = Null,
For all the SA do
    Pop ← CH.generate (SA.Encoding)//Randomly Generation for Population (100)
End for
    Fit ← Fit.Calculation//Calculate the Fitness
if Fit.change or Gen < 100 then
    For all the CH do
        CH.select//Chromosome Selection
        CH.crossover//Chromosome Crossover
        CH.mutate//Chromosome Mutation
        Pop ← CH.NewGenerate//New Population
        Fit ← Fit.NewCalculation
    End for
End if
    SR ← CH.Decoding
Return SR, SV, DR
```

Algorithm 1: NGA for scheduling optimisation.

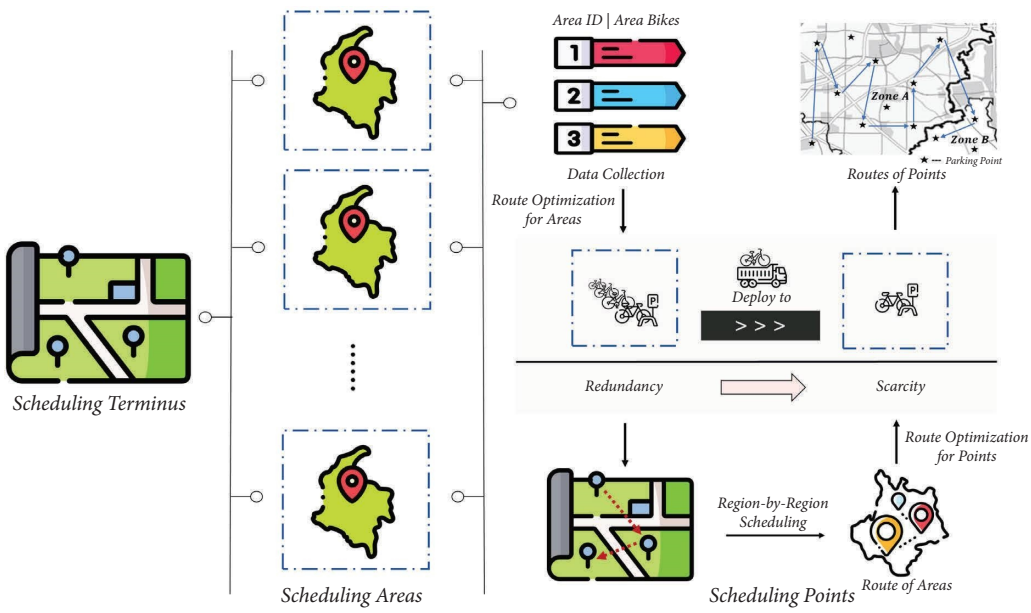


Figure 3: The schema of *HATB*.

3.1. Algorithm Settings. The experiment data comes from the 2017 Mobike cup algorithm challenge, which involves 3,214,096 orders and 485,465 bikes (10th May 2017–23rd May 2017). The characterized user travel, as presented in Table 3, reflects that the data are consistent with the "last mile" definition [16], and hence, shows its reasonable usability according to prominent tidal characteristics.

This paper proposes a GA with natural number encoding (NGA), as defined in Algorithm 1, to optimize the scheduling. In general, the first and the last 0 s represent the scheduling terminus or area, [1, N] represents the zones that need to be scheduled, and other 0 s separate the routes of different vehicles, e.g., a chromosome example might be 0-3-0-1-2-5-7-0-4-8-6-0, namely, three vehicles serving eight zones.

Since the crossover and mutated sub chromosomes may lead to transport overload and overtime and the time window is more likely to be violated, the penalty factors for the two constraints are set to 10 and 500, respectively.

3.2. Scheduling Results. A total of 220 scheduling areas in the morning peak are used as *HATB* test cases to obtain the optimal hierarchical scheduling routes, as shown in Figure 3 and Tables 4 and 5.

Table 4: The optimal routes for scheduling areas in the morning peak.

Routes for scheduling area	Demand	Number of vehicles	Total distance (km)
0-135-29-113-112-150-205-20-44-170-130-146-0	143	1	719.542007
0-40-88-138-104-65-12-191-115-52-162-83-157-0	142	1	
0-7-45-93-61-129-188-199-158-217-87-53-165-85-119-36-34-210-0	140	1	
0-5-6-50-47-41-75-63-86-27-101-125-200-178-81-78-89-122-187-195-203-207-0	140	1	
0-18-21-95-132-148-91-71-161-192-117-0	139	1	
0-181-106-102-145-135-29-113-112-150-205-20-0	138	1	
0-51-177-92-79-84-26-38-214-215-166-168-183-0	133	1	
0-31-143-114-37-118-99-156-163-137-149-194-0	114	1	
0-54-66-74-128-94-136-201-179-190-152-35-30-33-204-218-0	114	1	
0-90-78-89-97-64-105-185-131-127-189-140-100-8-82-22-25-98-24-144-124-197-0	113	1	
0-3-70-55-169-172-182-109-208-0	113	1	
0-62-103-202-43-11-16-57-23-108-153-211-0	111	1	
0-4-1-2-14-17-67-48-15-42-56-72-68-99-156-219-0	110	1	
0-44-170-30-146-159-173-174-69-9-46-49-19-141-186-175-171-39-213-154-51-60-0	110	1	
0-180-139-142-198-147-116-212-134-167-184-126-216-0	110	1	
0-76-164-10-77-73-60-111-120-196-209-0	80	1	
0-28-133-110-193-123-154-32-13-107-121-155-206-0	79	1	

Table 5: Examples of hierarchical autonomous scheduling results for route one.

Routes for scheduling point	Scheduling area	Total distance (km)
a0-a11-a4-a18-a17-a9-a1-a12-a15-a19-a5-a0	170 (a0)	1.443
b0-b1-b3-b11-b4-b48-b39-b18-b42-b38-b7-b19-b54-b2-b28-b25-b26-b27-b43-b31-b12-b21-b45- b52-b30-b20-b46-b47-b6-b29-b40-b37-b14-b33-b9-b34-b51-b24-b32-b15-b55-b10-b44-b22-b16-b41-b17-b5-b13-b23-b53-b8-b50-b49-b36-b35-b0	130 (b0)	3.905
c0-c17-c23-c13-c15-c29-c32-c10-c35-c31-c2-c34-c5-c4-c14-c33-c22-c11-c9-c3-c6-c21-c20-c25-c30-c28-c24-c16-c27-c19-c8-c12-c26-c1-c18-c7-c0	146 (c0)	2.951

Table 6: The mapped scheduling results with examples of route one (Table 5).

Scheduling area (5-bit)	Routes for scheduling point (7-bit)
WX54C (no. 170)	WX54C **with** 0W, 20, E4, 06, 0C, J8, JB, LJ, NC, BW, BX, 0W
WX4GW (no. 130)	WX4GW **with** 00, 02, 08, PL, 06, 77, 7K, P4, NF, 7M, ZH, YY, N8, P0, PF, PY, PV, PS, ND, JL, P3, PB, N9, 03, JH, PZ, 7B, 78, 7F, 7E, KH, 7U, 7Q, EQ, ZN, EN, 09, Q8,G7, ZQ, GJ, ZK, NB, P8, P9, P5, 01, 04, 2N, U4, FZ, ZV, 0D, 3K, 86, DK, 00
WX5H2 (no. 146)	WX5H2 **with** 5X, 7C, QX, DM, GL, 6M, Y6, F9, 4N, 4J, DS, DT, VC, TY, DC, D3, TZ, ER, G0, G8, 74, UE, UK, GE, 1E, 12, QK, 7D, 75, 77, W2, EP, G9, ZG, 79, 48, 5X

As described in the previous section, regional scheduling is applied for each layer. Note that, each scheduling area has a positive or negative raw demand that reflects the redundancy or scarcity of bikes. Furthermore, scheduling prioritizes self-satisfaction in the route, namely, redundancy supports scarcity, and hence, route demand indicates the self-satisfaction gap for corresponding regional scheduling.

Therefore, the experimental results can be summarized as the total scheduling distance for optimal routes in the entire Beijing is around 719.5 km. As for satisfying the demands, 17 vehicles are required to participate in the scheduling to deploy bikes. Moreover, using route one in Table 4 as an example, the scheduling can be summarized as a vehicle departing from the scheduling terminus and returning to the terminus after completing regional scheduling sequentially and autonomously in accordance with the area-point (middle-bottom) hierarchy. In addition, the routes' information in reality for Table 5 is mapped to Table 6 via GeoHash.

3.3. Model Evaluation. To further verify the reliability of this model, Figure 4 shows the comparison for iteration between the *HATB* and other models proposed in the literature with similar objectives. Specifically, based on the GA, Gao et al. [35] provided a promising perspective on improving operation efficiency by reducing operating costs and service quality during peak times to minimize the total operating

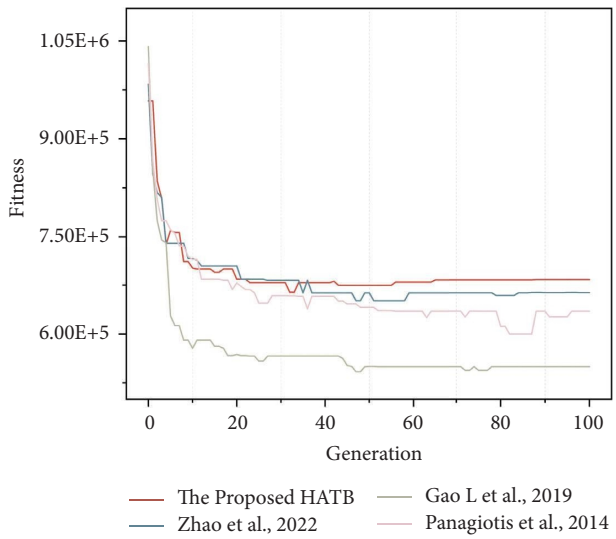


Figure 4: The comparison between different scheduling models.

costs. Angeloudis et al. [36] achieved user appeal increase by offering a new method of planning bike routes and distributions. Moreover, Zhao et al. [37] optimized the total scheduling distance to accommodate large-scale scheduling via an ACO.

The proposed *HAT*B converged at the 64th generation, and the total time cost is 148.9 s, with an average running time of 15 s per generation. Due to the different objectives, only the convergence speed of the above models is compared. It can be seen from Figure 4 that the *HATB* significantly outperforms the model proposed in the existing literature. Such a result indicates that this model is more proper for practical scheduling applications since it optimizes with higher convergence speed and lower time complexity.

4. Conclusion and Future Works

Even though the emerging and diversified technologies and demands are driving TS to renovate conventional MS to be self-actuating, the current bicycle-sharing scheduling and maintenance rely on manual experience, which lacks scientific guidance and efficiency. Therefore, to achieve sustainable development, operators urgently need to develop a rational scheduling strategy to balance the distribution conflicts and supply user demands in time.

Therefore, this paper proposes a hierarchical scheduling model, called *HATB*, to address the unsolved issues by current studies in terms of slow convergence, high time complexity, and problematic application, and hence, to support the rational and autonomous provision of LMS. In summary, according to bicycle-sharing properties, namely, spatio-temporal characteristics, cross-regional mobilities, and actual demands, *HATB* takes 220-morning peak areas as tests to validate its improved validity, feasibility, and efficiency for practical application.

As compared to the similarly used methods, *HATB* can, accordingly, obtain the following improvements. A hierarchical framework is first designed through GeoHash encoding to solve the cross-regional mobility of bikes and reduce the time complexity of global optimization. Next, a GA for regional scheduling is built by combining the tidal characteristics of bicycle-sharing to minimize operating costs and maximize user satisfaction, which significantly accelerates the algorithm's convergence. Last, the use of actual orders considerably enhances the ability in the practical application of instant response to any regional scheduling demand.

This work was carried out as a preliminary to obtain the present results. However, there are still problems such as the inability to adapt scheduling throughout 24 hours or the lack of comprehensive constraints. As the closure of this study, one recommendation for further research is to use a form of "GA + Tabu" algorithm to exploit its global search capability and thus improve the big data processing capability. Another research direction is adding weather and road characteristics to optimize the model reliability further.

Data Availability

The data used to support the findings of this study have been deposited in the 2017 Mobike Cup Algorithm Challenge repository (https://biendata.xyz/competition/mobike/).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by National Key Research and Development Program of China (Grant no. 2020YFB1600400) and Collaborative Innovation Center for Transportation of Guangzhou (Grant no. 202206010056).

References

[1] S. Moridpour, X. Qu, N. Shiwakoti, and S. Hasan, "Sustainable and resilient transport infrastructure," *Journal of Advanced Transportation*, vol. 2021, Article ID 1576315, 2 pages, 2021.

[2] M. L. Mouronte-López, "Analysing the vulnerability of public transport networks," *Journal of Advanced Transportation*, vol. 2021, Article ID 5513311, 22 pages, 2021.

[3] N. Tsanakas, J. Ekström, and J. Olstam, "Estimating emissions from static traffic models: problems and solutions," *Journal of Advanced Transportation*, vol. 2020, Article ID 5401792, 17 pages, 2020.

[4] A. Fan, X. Chen, and T. Wan, "How have travelers changed mode choices for first/last mile trips after the introduction of bicycle-sharing systems: an empirical study in Beijing, China," *Journal of Advanced Transportation*, vol. 2019, Article ID 5426080, 16 pages, 2019.

[5] B. Hu, Y. Gao, J. Yan et al., "Understanding the operational efficiency of bicycle-sharing based on the influencing factor analyzes: a case study in nanjing, China," *Journal of Advanced Transportation*, vol. 2021, pp. 1–14, Article ID 8818548, 2021.

[6] L. Caggiani, R. Camporeale, M. Ottomanelli, and W. Y. Szeto, "A modeling framework for the dynamic management of free-floating bike-sharing systems," *Transportation Research Part C: Emerging Technologies*, vol. 87, pp. 159–182, 2018.

[7] A. A. Kadri, I. Kacem, and K. Labadi, "A branch-and-bound algorithm for solving the static rebalancing problem in bicycle-sharing systems," *Computers and Industrial Engineering*, vol. 95, pp. 41–52, 2016.

[8] M. Dell'Amico, E. Hadjicostantinou, M. Iori, and S. Novellani, "The bike sharing rebalancing problem: mathematical formulations and benchmark instances," *Omega*, vol. 45, pp. 7–19, 2014.

[9] S. Yan, J. R. Lin, Y. C. Chen, and F. R. Xie, "Rental bike location and allocation under stochastic demands," *Computers and Industrial Engineering*, vol. 107, pp. 1–11, 2017.

[10] L. You, B. Tuncer, and H. Xing, "Harnessing multi-source data about public sentiments and activities for informed design," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 2, pp. 343–356, 2019.

[11] L. You, J. He, W. Wang, and M. Cai, "Autonomous transportation systems and services enabled by the next-generation network," *IEEE Network*, vol. 36, no. 3, pp. 66–72, 2022.

[12] L. You, B. Tunçer, R. Zhu, H. Xing, and C. Yuen, "A synergetic orchestration of objects, data, and services to enable smart cities," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10496–10507, 2019.

[13] L. You, F. Zhao, L. Cheah, K. Jeong, P. C. Zegras, and M. Ben-Akiva, "A generic future mobility sensing system for travel data collection, management, fusion, and visualization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4149–4160, 2020.

[14] L. Cheng, Z. Mi, D. Coffman, J. Meng, D. Liu, and D. Chang, "The role of bike sharing in promoting transport resilience," *Networks and Spatial Economics*, vol. 22, no. 3, pp. 567–585, 2021.

[15] F. Soriguera and E. Jiménez-Meroño, "A continuous approximation model for the optimal design of public bike-sharing systems," *Sustainable Cities and Society*, vol. 52, Article ID 101826, 2020.

[16] D. Gimon, "Bike commuters contribution to balance shared bike systems during peak load," in *Proceedings of the 2018 IEEE International Smart Cities Conference (ISC2)*, pp. 1-2, Kansas City, MO, USA, September 2018.

[17] Y. Lu, U. Benlic, and Q. Wu, "An effective memetic algorithm for the generalized bike-sharing rebalancing problem," *Engineering Applications of Artificial Intelligence*, vol. 95, Article ID 103890, 2020.

[18] G. Shang, J. Xinzi, and T. Kezong, "Hybrid algorithm combining ant colony optimization algorithm with genetic algorithm," in *Proceedings of the 2007 Chinese Control Conference*, pp. 701–704, Zhangjiajie, China, July 2007.

[19] M. Montazeri, R. Kiani, and S. S. Rastkhadiv, "A new approach to the restart genetic algorithm to solve zero-one knapsack problem," in *Proceedings of the 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI)*, pp. 0050–0053, Tehran, Iran, December 2017.

[20] Y. Wang, J. Chen, and Y. Shen, "A multi-objective optimization model for vrp and vfp based on an improved ant colony algorithm," in *Proceedings of the 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, pp. 777–780, Chongqing, China, October 2019.

[21] D. Zhang, C. Yu, J. Desai, H. Lau, and S. Srivathsan, "A time-space network flow approach to dynamic repositioning in bicycle sharing systems," *Transportation Research Part B: Methodological*, vol. 103, pp. 188–207, 2017.

[22] C. S. Shui and W. Szeto, "Dynamic green bike repositioning problem–A hybrid rolling horizon artificial bee colony algorithm approach," *Transportation Research Part D: Transport and Environment*, vol. 60, pp. 119–136, 2018.

[23] F. Chiariotti, C. Pielli, A. Zanella, and M. Zorzi, "A dynamic approach to rebalancing bike-sharing systems," *Sensors*, vol. 18, no. 2, p. 512, 2018.

[24] M. Lang, "Study on the model and algorithm for multi-depot vehicle scheduling problem," *Journal of Transportation Systems Engineering and Information Technology*, vol. 24, pp. 65–69, 2006.

[25] H. Bae and I. Moon, "Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles," *Applied Mathematical Modelling*, vol. 40, no. 13–14, pp. 6536–6549, 2016.

[26] K. Sakakibara, M. Noishiki, S. Watanabe, I. Nishikawa, and H. Tamaki, "2b2 hierarchical approach with informational feedback for pickup and delivery problems (technical session 2b: vehicle scheduling and communication)," *Proceedings of the International Seaweed Symposium*, vol. 2006, pp. 48–53, 2006.

[27] F. Ni, L. Yan, K. Wu, M. Shi, J. Zhou, and X. Chen, "Hierarchical optimization of electric vehicle system charging plan based on the scheduling priority," *Journal of Circuits, Systems, and Computers*, vol. 28, no. 13, Article ID 1950221, 2019.

[28] Y. Zhang, "Investigation and Analysis of Travel Behaviors of Sharing Bicycles in Wenzhou," *United International Journal for Research & Techonology (UIJRT)*, vol. 02, no. 06, 2021.

[29] Y. Ma, X. Qin, J. Xu, and W. Wang, "A hierarchical public bicycle dispatching policy for dynamic demand," in *Proceedings of the 2016 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, pp. 162–167, Beijing, China, July 2016.

[30] L. You, G. Motta, K. Liu, and T. Ma, "City feed: a pilot system of citizen-sourcing for city issue management," *ACM Transactions on Intelligent Systems and Technology*, vol. 7, no. 4, pp. 1–25, 2016.

[31] A. Kempen, "What 3 Words Easy to use, easy to find any location anywhere," *Servamus Community-Based Saf. Secur. Mag.*vol. 115, no. 2, p. 57, 2022.

[32] M. Muroň, D. Procházka, and F. Dařena, "How do people describe a location: towards automatically generated locations description," in *Proceedings of the 25th European Scientific Conference of Doctoral Students*, p. 85, Czech, Europe, November 2021.

[33] G. M. Morton, *A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing*, International Business Machines Company New York, New York, NY, USA, 1966.

[34] G. Jiang, S.-K. Lam, F. Ning, P. He, and J. Xie, "Peak-hour vehicle routing for first-mile transportation: problem formulation and algorithms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3308–3321, 2020.

[35] L. Gao, W. Xu, and Y. Duan, "Dynamic scheduling based on predicted inventory variation rate for public bicycle system," *Sustainability*, vol. 11, no. 7, p. 1885, 2019.

[36] P. Angeloudis, J. Hu, and M. G. H. Bell, "A strategic repositioning algorithm for bicycle-sharing schemes," *Transportmetrica: Transportation Science*, vol. 10, no. 8, pp. 759–774, 2014.

[37] S. Zhao, C. Li, M. Tian, H. Zhang, Z. Xiao, and R. Hu, "Large-scale scheduling model based on improved ant colony algorithm," *Mobile Information Systems*, vol. 2022, Article ID 9116121, 6 pages, 2022.

Hindawi
Journal of Advanced Transportation
Volume 2022, Article ID 5869101, 12 pages
https://doi.org/10.1155/2022/5869101

Research Article

An Improved Runway Operation Capacity Model for V-Open Multirunway Airports in China

Zhiyuan Shen,[1] **Xinyu Xu**,[1] **Yong He**,[2] **YongGang Yan**,[3] **Lin Zhou**,[4] **and Yingying Hu**[1]

[1]*College of Civil Aviation, Nanjing University of Aeronautics and Astronautics, Nanjing, China*
[2]*East China Air Traffic Management Bureau, Civil Aviation Administration of China, Shanghai, China*
[3]*Central Air Traffic Management Committee, Beijing, China*
[4]*East China Air Traffic Management Bureau at Zhejiang Sub-Bureau, Civil Aviation Administration of China, Hangzhou, China*

Correspondence should be addressed to Zhiyuan Shen; shenzy@nuaa.edu.cn

Received 24 May 2022; Revised 2 August 2022; Accepted 30 September 2022; Published 12 October 2022

Academic Editor: Shao-Wu Cheng

Diverging runways are receiving considerable attention as replacements for parallel runways when constructing three or more runways in China. This has been prompted by the limitations associated with a single-direction airspace and associated regulations when using parallel runways. However, the operating efficiency of diverging runways is still greatly constrained by the separation standard for departing aircraft in CCAR-93TM-R5 and Decree of CAAC NO.123. To address these problems, an improved runway operation capacity model for a V-open multirunway airport is proposed to reduce the separation standard based on the equivalent lateral separation operation (ELSO) standard. A collision risk module is first proposed to assess the feasibility and safety of the proposed policy. Also, a novel runway capacity model for V-open multiple runways is then built to improve the operating efficiency for diverging runway. A series of comparative experiments based on actual flight data from Chengdu Tianfu International Airport demonstrated that reducing the divergence angle from 15° to 10° improved the runway operation capacity by about 8% based on the proposed improved policy. These research results could play a major role in facilitating appropriate amendments to the associated CAAC operation regulations.

1. Introduction

The International Air Transport Association (IATA) has predicted that China will become the largest civil aviation market in the world by around 2025 and involve the flow of 1.6 billion passengers by around 2037 [1]. Then, there is no doubt that COVID-19 significantly impacted civil aviation market and considered as the main factor in much recent studies [2]. Research studies on runway capacity estimation usually focus on the maximum of capacity in peak hour or one day. So, airline and airport pay more attention to theoretical maximum flight number under the priority of operation safety. By reducing the impact of the diverging angle, the departure separation can be decreased. Thereby, the numbers of airport flights permitted to taking-off and landing at peak hour increase. The ticket price at peak hour is usually higher than those at other time. Consequently, it leads to a higher airline revenue.

In the past, large airports at China tended to build additional parallel runways to improve their operation capacity. However, it has been difficult to reach estimated maximum operation capacities due to limitations associated with a single-direction airspace and its associated regulations. The diverging runway has advantages of low building cost and flexible operation procedures; therefore, it is the preferred choice for configuration of large domestic hub airports. It has led to diverging runways being considered for replacing parallel runways when constructing three or more runways in China, such as at Beijing Daxing International Airport (ZBAD) and Chengdu Tianfu International Airport (ZUTF). While configuring runways in different directions facilitates the flexible utilization of airport airspace and

operation procedures, such diverging runways have mainly been used for aircraft takeoffs [3].

However, the departure efficiency for diverging runways is still greatly affected by the separation standard for departing aircraft in CCAR-93TM-R5 [4] and Decree of CAAC NO.123 [5]. Specifically, aircraft departing simultaneously from parallel or diverging runways need to comply with the separation standard that their takeoff courses diverge by 15° or more immediately after departure. The initial purpose of implementing diverging departures is to improve the operating efficiency and reduce flight delays [6]. However, numerous studies have shown that the strict separation standard greatly restricts the use of the available airspace, especially for multiple runways and diverging runways [7, 8]. For example, a larger diverging airspace is needed when applying the separation standard to three parallel runways, and separated departure flight paths could overlap the diverging runways to result in increased waiting time for aircraft departures or reducing the takeoff runway distance, resulting in a decreased departure capacity. Therefore, many studies have attempted to reduce the divergence angle while still conforming to safety standards [9, 10].

The recent rapid development and application of performance based on navigation (PBN) has significantly improved navigation precision in the terminal area. A novel concept of the equivalent lateral separation operation (ELSO) standard [11] was proposed, which is able to reduce divergence angles while maintaining established minimum lateral spacing between departure paths. A series of related theorical and practical evaluation experiments were conducted at Atlanta Airport over a 1-year period in 2011 [12]. The Federal Aviation Administration (FAA) finally approved for the divergence angle to be reduced to 10° at Atlanta Airport based on the safety of the experimental results, which facilitated the development of a new departure procedure. Nowadays, any airport in the USA can operate with the standard 10° divergence angle when both aircraft are flying RNAV (area navigation) standard instrument departures (SID) from parallel runways according to the policy in FAAO.7110.65W [13].

All of the international airports in China have been designed and operated under RNAV SID procedures, and more runways are planned to be built in order to meet the increasing number of flights at the busiest Chinese airports (e.g., Shanghai Pudong, Hangzhou Xiaoshan, and Yunnan Changshui), including to stimulate the economy affected by COVID-19 with greater infrastructure investment [14–16]. Reducing delays and improving operation efficiency are two important steps to reduce the impact of COVID-19 on the aviation industry. However, the superiority of the higher navigation precision of RNAV has not been performed to improve the operating efficiency for the applicable standard and additional runways [17, 18]. Most importantly, the existing separation standard, including the minimum requirement for a 15° divergence angle between independent parallel departure operations, significantly constrains the potential of multirunway operations [19].

In accordance with the development strategies of the CAAC and the Central Military Commission (CMC) of ATC that pertain to higher flight punctuality rates and better civil airspace planning [20], based on the actual aircraft departure path data from ZUTF parallel runways, we proposed modifying the regulations associated with the separation standard by referring to the ELSO standard. We further proposed a capacity estimation model for V-open multiple runways and tested the operating efficiency of the proposed new separation standard relative to the conventional one. Specifically, taking ZUTF as an example, two main modules are developed to test the assumption: (1) the collision risk module based on ELSO for assessing the feasibility and safety of the proposed policy, and (2) the V-open multirunway capacity module for measuring the increases in runway capacity. This study provides theoretical support and reference data for the development of future regulations to reduce the divergence angle standard and the operation procedure of V-open runways. All terminology used in this paper is listed in Table 1.

The remainder of this paper is organized as follows: Section 2 describes the basic theory of V-open runway configuration and ELSO, Section 3 details the proposed collision risk module and V-open multirunway capacity module, Section 4 reports on the theoretical analyses and experimental simulations performed to test our proposed framework, and Section 5 summarizes the obtained findings.

2. Basic Theory

The current divergence requirements for departing aircraft are defined in CCAR-93TM-R5 and Decree of CAAC NO.123. There are three key rules involved in separating aircraft departing from the same runway or parallel runways based on using either conventional SID or RNAV SID. In all cases, aircraft must establish radar identification within 1 mile of the end of the takeoff runway and travel along courses that diverge by 15° or more immediately after departure.

2.1. Layout of V-Open Runways. The layout of the runways at two airports (ZBAD and ZUTF) with diverging runways served as the research model of V-open runways in this study. As illustrated in Figure 1, including three runways totally in our study scene, two runways are parallel, and the other runway is V-open with the parallel runways. For parallel runways, we try to use ELSO to reduce the angle of course divergence for improving the efficiency of airspace usage. For the V-open runway, the main effort is to reduce the coverage impact of the parallel runway departure courses on the third runway. The diverging runway in the figure is used only for aircraft departures toward the east direction. One departure course coincides with the runway centerline extension; another one diverged by alpha angle. In this case, pilots are rather easier to master departing course whatever runway is used. On the other hand, the location of the runway direction is significantly affected by the surrounding terrain, so departure courses along the extension line of the runway centerline could minimize obstacle restrictions. Therefore, there is no need to simultaneously consider

Table 1: List of terminology used in this paper.

Terminology	Referred to
ELSO	Equivalent lateral separation operation
CAAC	Civil Aviation Administration of China
IATA	International Air Transport Association
ZBAD	Beijing Daxing International Airport
ZUTF	Chengdu Tianfu International Airport
PBN	Performance based on navigation
FAA	Federal Aviation Administration
RNAV	Area navigation
SID	Standard instrument departures
CMC	Central Military Commission

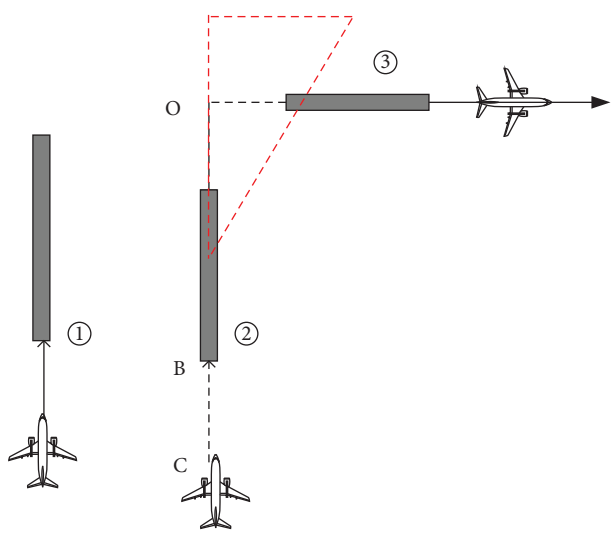


Figure 1: Layout of a V-open runway.

mutual diversion and considering one runway's trajectory divergence could better estimate the runway.

In accordance with the regulations in CCAR-93TM-R5, aircraft can depart from parallel runways simultaneously when the centerlines of the parallel runways are separated by at least 760 m and the courses of the aircraft diverge by 15° or more immediately after departure. In terms of approach operations, correlated parallel approaches and independent parallel approaches are authorized for minimum parallel runway spacing of 915 m and 1035 m, respectively.

The departure course relative to the runway centerline might cover the takeoff zone of the diverging runway in Figure 1. In other words, the departure interval for diverging runways is highly dependent on the time separation between aircraft flying over the intersection zone. There is no doubt that this situation will adversely affect the airport operating efficiency and runway capacity [21]. The concept of ELSO is introduced to solve this problem in this study.

2.2. Principle of ELSO. ELSO was proposed to meet the minimum spacing of the general divergence standard defined in CCAR-93TM-R5. In other words, the divergence angle can be reduced as long as the established minimum lateral spacing between departure paths is maintained. After a 1-year safety evaluation experiment was conducted at Atlanta Airport in 2011, the FAA authorized that the course divergence angle could be reduced to 10° when both aircraft are departing from parallel runways and are flying RNAV SID.

For the independent parallel departure operation, the general standard defines that the minimum spacing between parallel runways ($r_{\min}$) should be 760 m and that the departure-course divergence ($\alpha_{\min}$) should be a minimum of 15°. Figure 2 illustrates some key parameters of the general departure operation. In the figure, dashed lines represent the nominal departure path and the distance (d) from the runway end will determine nominal spacing ($n(d)$) between the nominal departure path, which can be obtained as follows: [22]

$$n(d) = r_{\min} + d \times \tan(\alpha_{\min}). \tag{1}$$

The actual path deviation due to various factors such as the wind direction is also illustrated in Figure 2. We can assume that the resulting width of the departure path can be estimated by single or multiple standard deviations (σ) of the related dispersion distribution. Hence, $\sigma_{\mathrm{STO},C}$ represents the track width of the conventional straight departure procedure along the centerline of the runway, and $\sigma_{\mathrm{DIV},C}$ indicates the track width of the diverging conventional departure procedure. The lateral spacing of the diverging departure path is given as follows:

$$s(d) = r_{\min} + d \times \tan\left(\alpha_{\min} - \sigma_{\mathrm{DIV},C}\right) - d \times \tan\left(\sigma_{\mathrm{STO},C}\right). \tag{2}$$

Therefore, distance $s(d)$ can be regarded as the minimum lateral spacing between actual departure flight paths under the conventional divergence standard and serves as the reference distance when using the ELSO standard.

When parallel runways are separated by more than 760 m, lateral spacing requirement $s(d)$ can also be achieved for a smaller divergence angle, as shown in Figure 3. In the figure, $\sigma_{\mathrm{STO},R}$ indicates the width of the RNAV departure path following the extended centerline, and the width of the RNAV departure path whose centerline is angled relative to the runway centerline is $\sigma_{\mathrm{DIV},R}$.

Given reference distance $s(d)$ and staggered distance t at the end of the runway, the runway distance can be expressed as follows:

$$r = s(d) + d \cdot \tan\left(\sigma_{\mathrm{STO},R}(d)\right) - (d+t)\tan\left(\beta - \sigma_{\mathrm{DIV},R}(d+t)\right), \tag{3}$$

where $\sigma_{\mathrm{STO},R}(d)$ and $\sigma_{\mathrm{DIV},R}(d+t)$ represent the actual track divergence characteristics of RNAV straight-line departure and offset departure, respectively. Therefore, based on the specific runway configuration, equivalent divergence angle β that can implement independent parallel departures under the ELSO standard is as follows:

$$\beta = \arctan\left\{\left(\frac{1}{d+t}\right) \cdot \left(s(d) - r + d \cdot \tan\left(\sigma_{\mathrm{STO},R}(d)\right)\right)\right\} + \sigma_{\mathrm{DIV},R}(d+t). \tag{4}$$

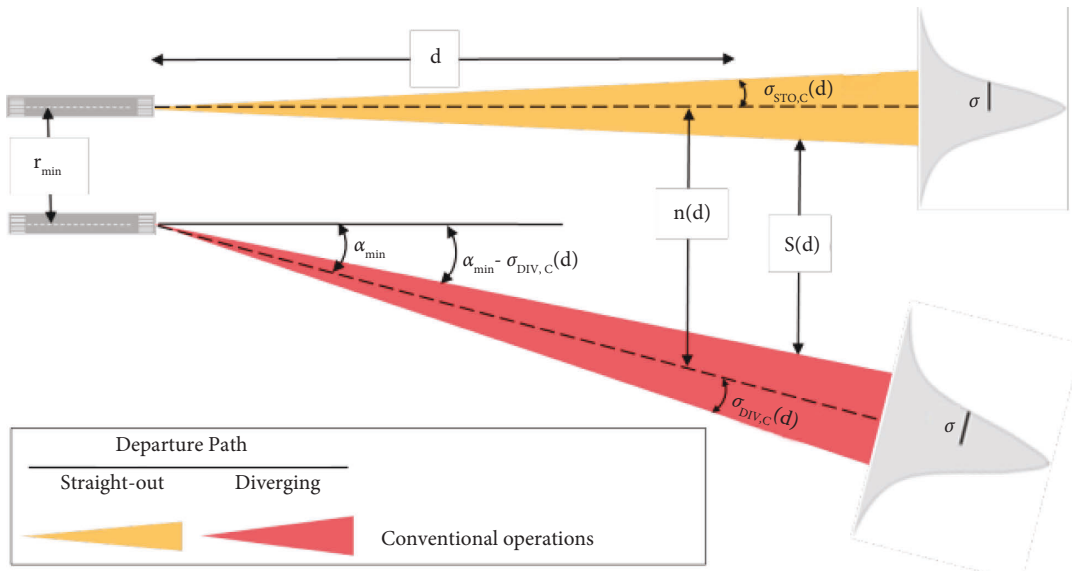


Figure 2: Conventional departure path and associated parameters.

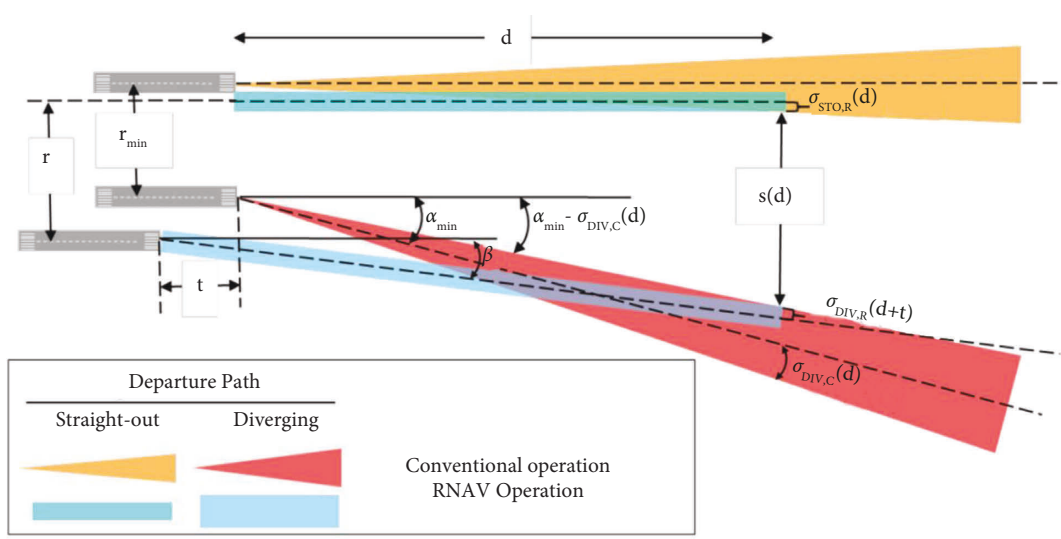


Figure 3: Improved departure path based on ELSO.

Equivalent divergence angle β can ensure that the departure path distance at the end of the runway (d) is the same as or even greater than the reference distance. If the nominal departure path fails to establish a lateral separation of d nautical miles from the end of the runway, the ELSO divergence standard will assume that the departure will be implemented with the traditional 15° divergence angle after this distance.

3. Proposed Modules

A collision risk model was proposed to quantitatively evaluate safety when the divergence angle is reduced based on ELSO. The minimum divergence angle can also be obtained under the safety requirement. In order to further assess the benefits of the new standard, we determined the operating efficiency of a V-open multirunway capacity model.

3.1. Collision Risk Model. The collision risk model mainly considered the path deviation of the actual flight caused by the precision of navigation equipment, which is greatly affected by the aircraft speed. We can assume that both the navigation error and speed error conform to a Gaussian distribution and are given as follows:

$$\varepsilon_{ny} \sim N\left(\mu_n, \sigma_n^2\right), \tag{5}$$

$$\varepsilon_{ay} \sim N\left(\frac{1}{2}\mu_a t^2, \frac{1}{4}\sigma_a^2 t^4\right), \tag{6}$$

where μ_n and μ_a indicate the mean values and σ_n and σ_a are the standard deviations. Subscripts ny and ay denote the aircraft lateral deviation caused by two factors. The lateral collision risk can be computed using equations (7) to (10):

$$\begin{aligned} &\varepsilon_y \sim N\left(\mu_y, \sigma_y^2\right) \\ &\mu_y = \mu_n + \frac{1}{2}\mu_a t^2, \sigma_y^2 = \sigma_n^2 + \frac{1}{4}\sigma_a^2 t^4 \end{aligned}, \tag{7}$$

where ε_y denotes the distribution of the lateral deviation. The actual lateral distance between simultaneously departing aircraft is given as follows:

$$\begin{aligned} D_y'(t) &= L_{2y}'(t) - L_{1y}'(t) = D_y(t) + \left(\varepsilon_{2y} - \varepsilon_{1y}\right) \\ &\sim N\left(D_y(t) + \mu_{2y} - \mu_{1y}, \sigma_{2y}^2 + \sigma_{1y}^2\right) \\ L_{1y}'(t) &= L_{1y}(t) + \varepsilon_{1y}, L_{2y}'(t) = L_{2y}(t) + \varepsilon_{2y} \end{aligned}, \tag{8}$$

where $L_{1y}'(t)$ and $L_{2y}'(t)$ represent the actual path positions of two different aircraft. The probability density function for the lateral direction can be expressed as follows:

$$f(y) = \frac{1}{\sqrt{2\pi\left(\sigma_{2y}^2 + \sigma_{1y}^2\right)}} \exp\left(-\frac{\left(y - \left(D_y(t) + \mu_{2y} - \mu_{1y}\right)\right)^2}{2\left(\sigma_{2y}^2 + \sigma_{1y}^2\right)}\right). \tag{9}$$

The probability density function of the actual distance between the two aircraft is integrated to obtain the lateral collision risk:

$$P_y = p\left\{y_1 < D_y'(t) < y_2\right\} = \int_{y_1}^{y_2} f(y)dy, \tag{10}$$

where y_1 and y_2 represent the minimum and maximum separations between the aircraft, respectively, and can be calculated from the aircraft wing spans. Similarly, the probability density functions in the vertical and horizontal directions are given as follows:

$$f(x) = \frac{1}{\sqrt{2\pi\left(\sigma_{2x}^2 + \sigma_{1x}^2\right)}} \exp\left(-\frac{\left(x - \left(D_x(t) + \mu_{2x} - \mu_{1x}\right)\right)^2}{2\left(\sigma_{2x}^2 + \sigma_{1x}^2\right)}\right), \tag{11}$$

$$f(z) = \frac{1}{\sqrt{2\pi\left(\sigma_{2z}^2 + \sigma_{1z}^2\right)}} \exp\left(-\frac{\left(z - \left(D_z(t) + \mu_{2z} - \mu_{1z}\right)\right)^2}{2\left(\sigma_{2z}^2 + \sigma_{1z}^2\right)}\right), \tag{12}$$

Finally, the collision risk model is obtained as follows:

$$P = \int_{x_1}^{x_2} f(x)dx \int_{y_1}^{y_2} f(y)dy \int_{z_1}^{z_2} f(z)dz. \tag{13}$$

3.2. V-Open Multirunway Capacity Model. Wake turbulence was one of the main factors influencing the design of the V-open multirunway capacity model [23, 24]. At domestic hub airports, the number of medium aircraft accounts for 99% and heavy aircraft accounts for 1% at Chinese domestic hub airports. Therefore, in most cases, it is necessary to consider the wake impact for medium aircraft following heavy or medium aircraft, and detailed wake interval and accounts of different categories are shown in Table 2. Mainly, the wake generated from taking-off aircraft on the diverging runway affects the operation of departing aircraft on parallel runways and vice versa. A typical layout of V-open multiple runways is illustrated in Figure 1, in which there are two parallel runways and one diverging runway. There are generally two approach operations for parallel runways depending on whether or not they have different widths: independent parallel approach and correlated parallel approach [25]. Small airports in China currently use independent parallel approaches due to the high requirements for equipment and working skills. However, another diverging path of 30° for aircraft to safely go around also needs to be considered [26]. Therefore, the capacity evaluation model was proposed here based only on the correlated parallel approach.

An operation schematic diagram of parallel runways using the correlated parallel-approach-independent parallel departure mode is shown in Figure 4. Both runways are used for aircraft arrivals and departures. Aircraft arrive and depart on each runway independently of each other. However, in order to ensure the safe operation of aircraft, the aircraft must land in the same direction on Runways 1 and 2. A safe distance needs to be maintained between aircraft on the parallel runway approaches (δ_a).

Assuming that the distance between the two runways is D, as shown in Figure 4, the distance between two successively approaching aircraft on Runway 1 is the following:

$$l_{\min} = 2\sqrt{(\delta_a)^2 - D^2}. \tag{14}$$

Then, the time interval between two successively approaching aircraft a and aircraft c on Runway 1 is as follows:

$$t_{\min} = \max\left(\frac{l_{\min}}{v}\right). \tag{15}$$

The controller can use the insertion method to arrange for aircraft b to land on Runway 2 between aircraft a and aircraft c, but the time interval between aircraft a and aircraft c must be consistent with the following formula [27]:

$$T_{ac} + B_{ac} \geq t_{\min}. \tag{16}$$

The operation sequence diagram of three runways for Runways 1 and 2 using related parallel-approach-independent parallel departure procedures is shown in Figure 5. Assuming that approaching aircraft a on Runway 1 arrives at the runway threshold at 00:00, then within the interval between aircraft a and aircraft c, the departing aircraft can be released on Runway 1 or the approaching aircraft can be arranged to land on Runway 2. During time gap G_2 of the continuous-approach aircraft on Runway 2, the departing aircraft can continue to be released, but the controller can also release the departing

Table 2: Wake influence parameters.

Category	SH-SH	SH-SM	SM-SH	SM-SM
Wake interval	8 km	6 km	10 km	6 km
Accounts	4%	16%	16%	64%

Remarks: SH/SM denotes heavy and medium aircraft separately.

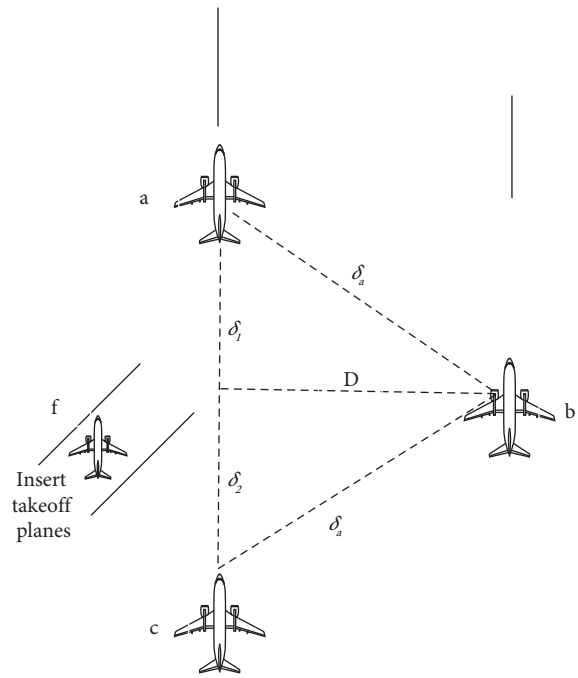


Figure 4: Parallel approach operations.

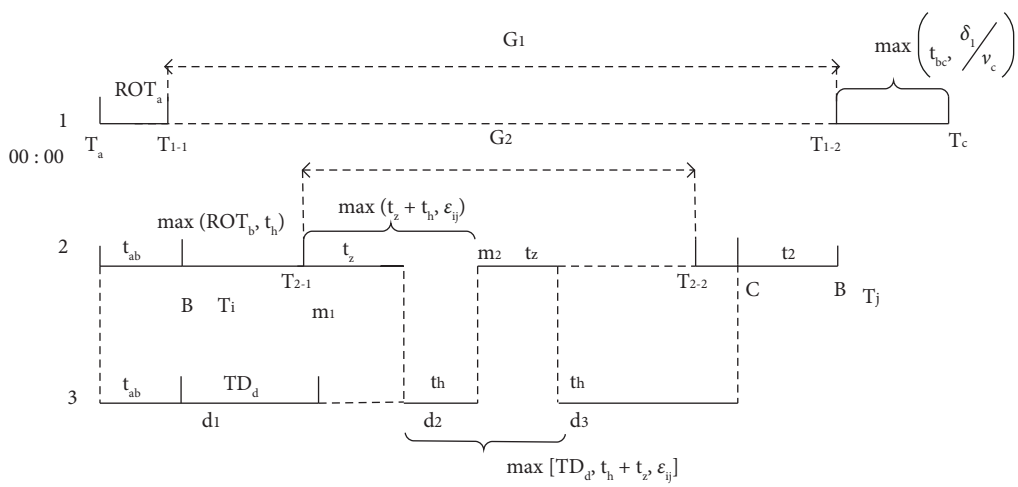


Figure 5: Time sequence model of V-open three-runway operation.

aircraft on Runway 3 after the departing aircraft on Runway 2 has been released for time t_z.

Time interval G_1 between two consecutively approaching aircraft on Runway 1 is as follows:

$$G_1 = T_{1-2} - T_{1-1} = T_{ac} + B_{ac} - ROT_a - \max(t_{bc}, t_{\min}). \tag{17}$$

When $G_1 > 0$ (i.e., when the following conditions are satisfied), n departing aircraft can be inserted in gap G_1:

$$\begin{aligned} E[T_{ac} + B_{ac}] \geq{}& E[ROT_a] + (n-1) \cdot E(\varepsilon_{ac}) \\ &+ E[\max(t_{bc}, t_{\min})] + E(\tau). \end{aligned} \tag{18}$$

Then, the entry and exit capacities of Runway 1 are as follows:

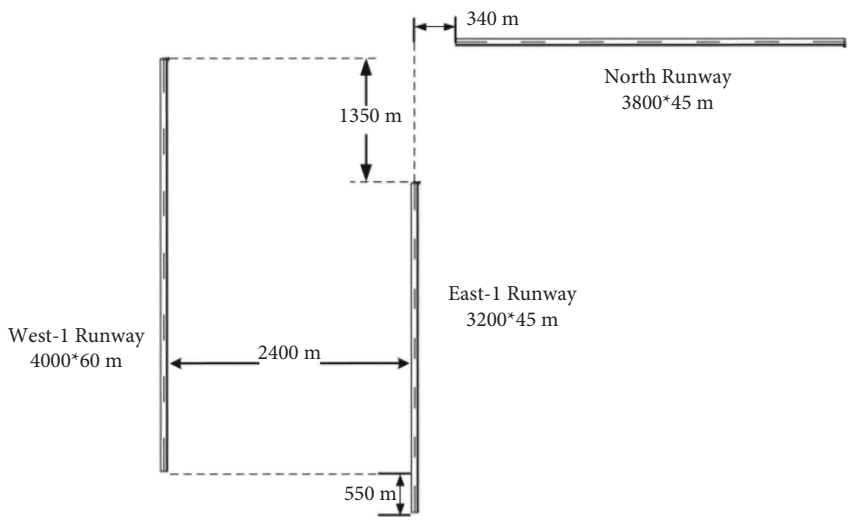


Figure 6: V-open runway layout at ZUTF.

$$C_{1-\text{arrival}} = \frac{3600}{E\left[T_{ac} + B_{ac}\right]}, \tag{19}$$

$$C_{1-\text{departure}} = \sum_{i=1}^{n}\sum_{j=1}^{n} TG_1 \cdot P_{ij} \cdot n_{ij}, \tag{20}$$

where TG_1 is the total number of clearances on Runway 1 within 1 h and n_{ij} is the number of departing aircraft that can be released between two adjacent approaching aircraft N/M on Runway 1.

Time interval G_2 between two successively approaching aircraft on Runway 2 is as follows:

$$G_2 = T_{1-2} - T_{1-1} = T_{ij} + B_{ij} - \max\left(ROT_b, t_h\right) - \max\left(t_2, \frac{\delta}{v_j}\right). \tag{21}$$

When the following conditions are met, k departing aircraft can be inserted in gap G_2:

$$E\left[T_{ij} + B_{ij}\right] \geq E\left[\max\left(ROT_b, t_h\right)\right] + (n-1) \cdot E\left[\max\left(t_z + t_h, \varepsilon_{ij}\right)\right] + E\left[\max\left(t_2, \frac{\delta}{v_j}\right)\right] + E(\tau). \tag{22}$$

Then, the entry and exit capacities of Runway 2 are as follows:

$$C_{2-\text{arrival}} = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{x}^{TG_1} \frac{G_x}{E\left[T_{ij} + B_{ij}\right]}, \tag{23}$$

$$C_{2-\text{departure}} = \sum_{i=1}^{n}\sum_{j=1}^{n} TG_2 \cdot P_{ij} \cdot k_{ij}, \tag{24}$$

where G_x is the time interval between the first successive approach aircraft on Runway 1, TG_1 and TG_2 are the total numbers of clearances on Runways 1 and 2, respectively, within 1 h, and k_{ij} is the number of takeoff and departing aircraft that can be inserted in the time interval between two consecutively arriving aircraft i and j on Runway 2.

In gap G_2 of Runway 2, assuming that Runway 3 can release m aircraft, according to the operation sequence diagram, the following equation can be derived:

$$\left(T_j - t_2\right) - T_{2-1} \geq (m-1) \cdot \max\left[TD_d, t_z + t_h, \varepsilon_{ij}\right] + \tau, \tag{25}$$

Therefore, the departure capacity of Runway 3 is as follows:

$$C_{3-\text{departure}} = \sum_{i=1}^{n}\sum_{j=1}^{n} TG_2 \cdot P_{ij} \cdot m_{ij}, \tag{26}$$

where m_{ij} is the number of takeoff and departing aircraft that can be inserted in the interval between two consecutively arriving aircraft i and j on Runway 2.

4. Experiments and Analysis

This section describes associated experiments and data analysis for the runway layout at ZUTF. As presented in Figure 6, this airport has two parallel runways separated by 2400 m and one diverging runway whose centerline is angled at 90°. The West Runway and East Runway are used for northbound departures and landings, and the North Runway is only used for eastward departures. The collision risk for a reduced divergence angle is evaluated to determine whether it satisfies the minimum requirement. The increase in the runway capacity is determined by a theoretical analysis and further tested using simulation experiments.

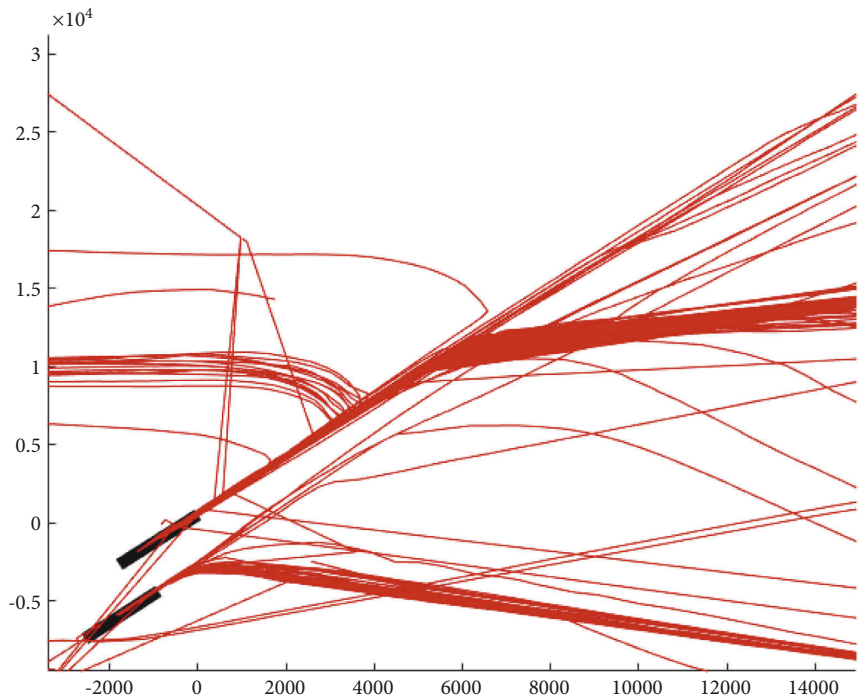


Figure 7: Analysis of actual departure paths at ZUTF.

Table 3: Performance parameters of different aircraft.

Aircraft type	Length of fuselage (m)	Wing span (m)	Height of fuselage (m)	Final approach speed (km/h)	Runway threshold speed (km/h)	Takeoff speed (km/h)	Normal cruising speed (km/h)
Heavy	68.5	64.3	12.55	296	287	293	916
Medium	39.5	34.31	9.7	258	249	269	839
Light	16.55	16.97	4.86	213	204	189	333

4.1. Analysis of Collision Risk. The reference distance from the runway end is generally set as 10 nm (18.6 km) to calculate the reference spacing when applying the ELSO standard. Then, by analyzing the actual departure path at ZUTF (see Figure 7), which contains 998 flight data all departing from parallel runways on December 14 and 15, 2021, the path variance angle is obtained as $\sigma_{\mathrm{STO,C}} = 0.4^{\circ}$ and $\sigma_{\mathrm{DIV,C}} = 0.6^{\circ}$. In this case, minimum lateral spacing $s(d)$ can be easily obtained using equation (2) as the conventional divergence standard.

The width of the parallel runways at ZUTF is 2400 m and the runways are staggered by 1350 m. In this case, equation (4) can be used to evaluate the effect of reducing the divergence angle to 10° according to the ELSO standard.

Experiments on the collision risk were performed for the proposed risk evaluation model using MATLAB software (version 2020). The experimental parameters for aircraft departing from the parallel runways with a separation path angle of 10° and three types of aircraft (heavy, medium, and light) are listed in Table 3.

When the crosswind speed is 6 kt, the aircraft wake moves at 15 ft/s. The time needed from aircraft takeoff to the first navigation point during the departure procedure is about 120 s. Research has shown that the lateral movement distance of the wake at a wind speed of 6 kt is 640 m.

Table 4: Collision risk calculation parameters.

Parameter	Value
μ_n	0
σ_n	3
μ_a	0
σ_a	1.27
R	2400
σ_g	50
$D_z(t)$	300

The parameters of the collision risk model are listed in Table 4. The upper limit of the collision probability integration interval is given by the following equation:

$$y_1 = \frac{\left(\lambda_{1y} + \lambda_{2y}\right)}{2}. \tag{27}$$

The lower limits of the collision probability integration interval in the lateral, longitudinal, and vertical directions are given by the following equation:

$$\begin{cases} y_2 = \min\left(y_1 + 640, 2400 - y_1\right) \\ x_2 = x_1 + \left(5 + \overline{v}_{\mathrm{former.flight}}\right) \times 120 \\ z_2 = \min\{z_1 + (2+2) \times 120, 300 - z_1\} \end{cases}. \tag{28}$$

Table 5: Probabilities of collision risk with different divergence angles.

Divergence angle (degree)	Aircraft type	Side collision risk	Longitudinal collision risk	Vertical collision risk	Total collision risk
14	Heavy-heavy	7.4268×10^{-6}	2.7528×10^{-10}	1.6612×10^{-5}	3.3872×10^{-20}
	Heavy-medium	7.3578×10^{-6}	2.7415×10^{-10}	1.6844×10^{-5}	3.3833×10^{-20}
	Medium-medium	7.1581×10^{-6}	2.6924×10^{-10}	1.5512×10^{-5}	2.9811×10^{-20}
13	Heavy-heavy	9.6952×10^{-6}	2.6852×10^{-10}	1.6512×10^{-5}	4.2849×10^{-20}
	Heavy-medium	9.5586×10^{-6}	2.6964×10^{-10}	1.6300×10^{-5}	4.1873×10^{-20}
	Medium-medium	9.3254×10^{-6}	2.5368×10^{-10}	1.5946×10^{-5}	3.7491×10^{-20}
12	Heavy-heavy	2.7485×10^{-5}	2.7256×10^{-10}	1.6102×10^{-5}	1.1736×10^{-20}
	Heavy-medium	2.5263×10^{-5}	2.7415×10^{-10}	1.5968×10^{-5}	1.0732×10^{-20}
	Medium-medium	2.1452×10^{-5}	2.7238×10^{-10}	1.5317×10^{-5}	8.6751×10^{-19}
11	Heavy-heavy	4.5685×10^{-5}	2.6852×10^{-10}	1.6012×10^{-5}	1.9553×10^{-19}
	Heavy-medium	4.3947×10^{-5}	2.6732×10^{-10}	1.5858×10^{-5}	1.7664×10^{-19}
	Medium-medium	4.0359×10^{-5}	2.5845×10^{-10}	1.5577×10^{-5}	1.6075×10^{-19}
10	Heavy-heavy	7.3195×10^{-5}	2.8601×10^{-10}	1.6302×10^{-5}	3.4127×10^{-19}
	Heavy-medium	7.2873×10^{-5}	2.8586×10^{-10}	1.6400×10^{-5}	3.4163×10^{-19}
	Medium-medium	7.2647×10^{-5}	2.7985×10^{-10}	1.32852×10^{-5}	2.7009×10^{-19}

Table 6: Runway capacity calculation parameters.

Parameter	Value
DD interval	100 s
DA interval	180 s
AA interval	100 s
Minimum distance between landing aircraft and entrance	95 s
Average runway occupation time of approaching aircraft (ROT_i).	50 s
Average runway occupation time of departing aircraft (TD_i).	55 s
Final approach speed of heavy aircraft	300 km/h
Final approach speed of medium aircraft	270 km/h
$\sigma_{V\text{heavy}}$	0.14 (km/h)
$\sigma_{V\text{medium}}$	0.14 (km/h)

Table 7: Calculation results for runway capacity.

Runway capacity (aircraft/h)	Related parallel-approach-independent parallel departure	
	Existing	Optimized
West Runway	32	32
East Runway	26	29
North Runway	30	33
Total capacity	88	94

According to the collision risk model based on the calculation parameters, the collision risk of departing aircraft on the parallel West Runway and East Runway can be calculated as presented in Table 5.

The results in Table 5 indicate that when the West Runway and East Runway at ZUTF are operated in an independent parallel departure mode with a reduced divergence angle from 14 degree to 10°, the probability of aircraft collision risk is all less than the overall safety target level (1.0×10^{-8}). Therefore, it is considered that the operation spacing for the departing aircraft is safe and that the divergence angle for the simultaneous use of the two long-distance West Runway and East Runway at ZUTF can be reduced to 10°.

4.2. V-Open Multirunway Capacity Evaluation

4.2.1. Theoretical Analysis. When the departing aircraft on the East Runway has a divergence angle of 15°, the aircraft departure protection area is 724 m from the North Runway. Therefore, the aircraft on the East Runway will affect the departure of aircraft on the North Runway.

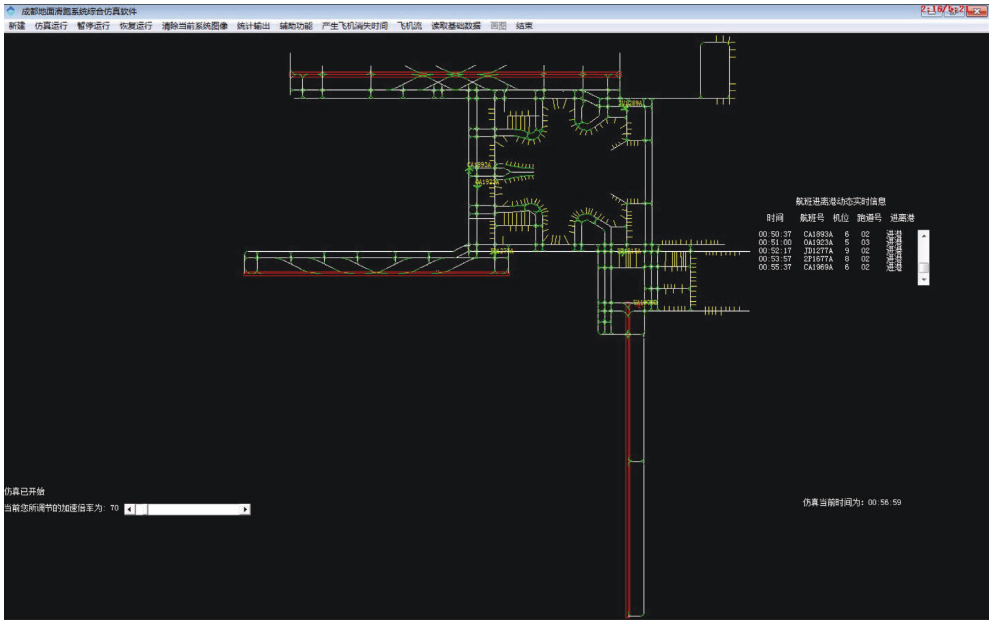


Figure 8: Ground simulation interface.

After the aircraft on the North Runway has taxied for $t_h = 25s$, the air traffic controller can allow the release of the departing aircraft on the East Runway. The departing aircraft on the East Runway is released at $t_z = 70s$, and the departing aircraft on the North Runway can then be aligned to the runway and allowed to taxi.

When applying the ELSO standard, the departure divergence angle is reduced to 10°. In this case, the departure protection zone of the departing aircraft on the East Runway still overlaps with the head of the North Runway, but the overlap distance is reduced to 494 m. Because departing aircraft on the North Runway have a taxiing distance of 600 m behind the runway head, the departing aircraft on East Runway are not limited by the divergence angle of 10°.

The improvements in the runway capacity when using the proposed model and associated parameters are given in Table 6. The capacity results for the conventional standard (divergence angle of 15°) and the ELSO standard (divergence angle of 10°) are presented in Table 7. It is clear that the number of flights per hour can increase by six with the reduced divergence angle.

4.2.2. Simulation Results. In order to test the theoretical results, several simulation experiments were performed for ZUTF based on the general airport simulation software jointly developed using the Visual Basic programming language. This airport surface simulation software has been approved by the ATC office of the CAAC. The airport runway layout was first set to summarize the measurement data of the airport ground flight area, and AutoCAD software was used to build a ground simulation model. After standardizing the positioning point information, two-dimensional coordinate information, taxi-path constraint rules, and parking space information, these data were imported into the ground capacity evaluation system to establish the ground simulation capacity evaluation model at ZUTF. The simulation interface is shown in Figure 8.

In accordance with the regulations for ZUTF operation, the North Runway, East Runway, and West Runway account for 70%, 5%, and 25%, respectively, of all departing flights, while the East Runway and West Runway account for 83% and 17%, respectively, of all arriving flights.

The flight schedule is selected as the data source of the simulation evaluation model, and the ground parameters used in the simulation program are listed in Table 8. Considering the characteristics of runway operation and operating rules, and under the restriction conditions of the percentages of the numbers of arriving and departing flights, the flight flow distribution proportion for each hour, and other characteristics, the original flight schedule of a typical day was randomly selected to generate pressurized flight and aircraft flows for use in the simulations.

Table 9 presents the results from the simulations of the air-ground joint operation capacity of the runway system at ZUTF under the conventional standard (divergence angle of 15°) and the ELSO standard (divergence angle of 10°). The runway capacity under the conventional standard was 74 per hour, which is less than that under the proposed ELSO standard (81 per hour). There were 8.1% more departing flights for the proposed policy during the peak hour and 11.1% more during the peak quarter than the conventional one. Moreover, the average flight delay time reduced significantly by 30% when using the proposed policy. Furthermore, given that 95% of flights involved medium aircraft and 5% involved heavy aircraft with an average passenger load factor of 85%, it can be inferred that passenger turnover at ZUTF would increase by about 6.3%.

The present theoretical analysis and simulations have demonstrated that the efficiency of ZUTF can be markedly

Table 8: Simulation operating parameters.

Simulation parameter	Value
Takeoff runway occupation time	50 s
Landing runway occupation time (fast departure lane)	50 s
Landing runway occupation time (end of runway)	75 s
Minimum time interval between consecutive takeoffs	100 s
Minimum time interval for continuous landings	100 s
Minimum distance between the landing aircraft and the entrance required to issue the takeoff permission	95 s
Time to cross the taxiway starting from standstill	40 s
Runway threshold speed	150 kt
Rapid departure speed	30 kt
Main slide and vertical connecting track speed	20 kt
General taxiway speed	10 kt
Taxi speed in apron area	5 kt
Time from the sliding position withdrawing the wheel to the completion of driving	150 s

Table 9: Simulation results under the conventional standard and the ELSO standard.

Operation mode	Flight number in peak hour	Total flight number	Average delay (min)	Delay in the air (min)	Delay on the ground (min)	Peak hour delay (min)	Flight number in peak quarter
Conventional standard	74	1189	7.9	4.2	4.7	13.9	18
ELSO standard	81	1264	6.8	3.0	3.8	13.6	20
Improvement	+8.1%	+6.3%	−13.9%	−30%	−19%	+2.1%	+11.1%

improved when the divergence angle of aircraft on the East Runway is reduced from 15° to 10°. This results in no overlap in the ground starting area, and the departure operations on the two runways are independent of each other. During operations, it is only necessary to consider that the go-around aircraft on the East Runway will release aircraft on the North Runway before 4.2 km from the runway entrance, which can effectively improve runway capacity by about 10%.

In summary, the proposed policy can improve (1) airport capacity for either parallel runways or diverging runways, (2) the travel experience in civil aviation with a higher flight punctuality rate, and (3) airspace resource optimization with more flexible operation procedures. The present research results can provide theoretical guidance for the future development strategies of the CAAC.

5. Conclusion

Limited by the conventional separation standard of 15 degrees, V-open multiple runways cannot perform its maximum operation function. A collision risk model is proposed to assess the feasibility and safety of the proposed policy at ZUTF airport. A novel capacity estimation model for V-open multiple runways is built to test the operating efficiency of the proposed new separation standard relative to the conventional one. Comparative experiments based on actual flight data from Chengdu Tianfu International Airport showed that the probability of aircraft collision risk is all less than the overall safety target level when divergence angle reduced from 15 degree to 10 degree and new capacity model effectively improves runway capacity by about 10%.

The present research findings can provide theoretical guidance for the CAAC to apply an operation standard to future V-open multiple runways. They might also be useful as a reference for capacity evaluations at airports with diverging runways used for civil aviation. In the future, when divergence angle is reducing from 15 degree to 10 degree, the proposed ELSO standard can also be applied in other airports with three parallel runways to improve the efficiency of airspace in China. Furthermore, the trajectories of go-around aircraft could also cover the diverging runway area for independent parallel approaching operation. Therefore, we will attempt to modify our capacity model with such considerations in our future work.

Data Availability

The trajectory data sampled from the Air Traffic Management Bureau, Civil Aviation Administration of China, used to support the findings of this study, are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest in this paper.

Acknowledgments

The authors acknowledge the financial support from the National Natural Science Foundation of China [grant no. 61773202] and Technology Project of Air Traffic Management, Civil Aviation Administration of China [grant no.TPATM201903].

References

[1] IATA, "IATA forecast Predicts 8.2 billion air Travelers in 2037," IATA, Available at: https://www.iata.org/en/, 2018.

[2] X. Sun, S. Wandelt, and A. Zhang, "Technological and educational challenges towards pandemic-resilient aviation," *Transport Policy*, vol. 114, pp. 104–115, 2021.

[3] H. Xiong, *Research on cross-Runway configuration and Operation mode[D]*, Civil Aviation Flight University of China, Guanghan, 2020.

[4] Civil Aviation Administration of China, *Air Traffic Management Rules of Civil Aviation Administration of China, CCAR-93TM-R5[S]*, Civil Aviation Administration of China, Beijing, 2009.

[5] Parallel Runway Simultaneous, *Parallel Runway Simultaneous Instrument Operation Management Regulations*, Decree of CAAC NO.123, Beijing, 2004.

[6] M. Cheng, Y. Li, and X. Han, "Constructing Scenarios' Network-of-flight conflict in approach of intersecting runway," *Journal of Advanced Transportation*, vol. 2021, no. 6, pp. 1–11, 2021.

[7] A. D. Mundra, H. Bateman, and A. P. Smith, "Converging runway Display Aid in the NAS: challenges, successes and outlook[C]," in *Proceedings of the 2011 IEEE/AIAA 30th Digital Avionics Systems Conference*, IEEE, Seattle, WA, USA, October 2011.

[8] X. Zhou, D. Zhu, and C. Li, "Optimization of MACAD model for intersecting runway capacity estimation[J]," *Aeronautical Computing Technique*, vol. 48, no. 3, pp. 91–94, 2018.

[9] S. Wang, Y. Zhang, H. Yu, and P. Wen, "A fast method to evaluate the runway capacity at the airport based on arrival/departure capacity curve," in *Proceedings of the Automation and Logistics (ICAL)*, IEEE, Chongqing, China, August 2011.

[10] C. Xu and L. Xin, "Cross-runway collision risk assessment under different crossing angles[J]," *Aeronautical Computing Technology*, vol. 48, no. 01, pp. 74–78, 2018.

[11] R. H. Mayer, D. J. Zondervan, and A. A. Herndon, "A standard for equivalent lateral spacing operations–parallel and reduced divergence departures[C]," in *Proceedings of the Ninth USA/EUROPE Air Traffic Management Research and Development Seminar (ATM2011)*, Berlin, Germany, 2011.

[12] S. Neyshabouri and L. Sherry, "Analysis of airport surface operations: a case-study of Atlanta Hartfield airport," in *Proceedings of the Transportation Research Board 93rd Annual Meeting*, pp. 12–16, Washington, DC, USA, 2014.

[13] Federal Aviation Administration, *D10 TRACON Order D10 7110..65, D10 TRACON Air Traffic Control*, Dallas/Fort Worth, TX, 2010.

[14] Y. Li, "Organizational and interorganizational factors affecting safety in the Chinese civil aviation industry," *Social Behavior and Personality: an International Journal*, vol. 47, no. 5, pp. 1–7, 2019.

[15] X. Qiu, "Development in Chinese transportation construction [M]," in *China 40 Years Infrastructure Construction*, pp. 79–98, Springer, Singapore, 2020.

[16] D. Xue, Z. Liu, B. Wang, and J Yang, "Impacts of COVID-19 on aircraft usage and fuel consumption: a case study on four Chinese international airports," *Journal of Air Transport Management*, vol. 95, Article ID 102106, 2021.

[17] D. Guo and D. Huang, "PBN operation advantage analysis over conventional navigation[J]," *Aerospace Systems*, vol. 4, no. 4, pp. 335–343, 2021.

[18] K. R. Sprong, B. M. Haltli, and J. S. DeArmon, "Improving flight efficiency through terminal area RNAV[C]," in *Proceedings of the 6th USA/Europe Air Traffic Management R&D Seminar*, 2005.

[19] X. Zhou, D. Zhu, C. Li, and Ti Wu, "Optimization of MACAD runway capacity evaluation model under cross-runway configuration[J]," *Aeronautical Computing Technology*, vol. 48, no. 03, pp. 91–94, 2018.

[20] T. M. Cheung, "The Riddle in the Middle: China's central military commission in the Twenty-first Century[M]," in *Influence on China's National Security Policymaking*, pp. 84–119, Stanford University Press, 2020.

[21] M. E. Johnson, X. Zhao, B. Faulkner, and J. P Young, "Statistical models of runway Incursions based on runway intersections and Taxiways," *Journal of Aviation Technology and Engineering*, vol. 5, no. 2, p. 3, 2016.

[22] Y. S. Ebrahimi and K. S. Chun, *Parallel Runway Requirement Analysis Study*, NASA, 1993.

[23] Y. Y. Tee and Z. W. Zhong, "Modelling and simulation studies of the runway capacity of Changi Airport," *Aeronautical Journal*, vol. 122, no. 1253, pp. 1022–1037, 2018.

[24] M. Villegas Díaz, F. Gómez Comendador, J. García-Heras Carretero, and R. M Arnaldo Valdes, "Analyzing the departure runway capacity effects of integrating optimized continuous Climb operations," *International Journal of Aerospace Engineering*, vol. 2019, pp. 1–10, 2019.

[25] J. Lei, X. Chong, D. Chen, Z. Chen, and Q. Chen, "A review of airport runway capacity evaluation model[C]//IOP Conference Series: Materials Science and Engineering," *IOP Conference Series: Materials Science and Engineering*, vol. 780, no. 7, Article ID 072019, 2020.

[26] P. D. Mascio, G. Rappoli, and L. Moretti, "Analytical method for calculating Sustainable airport capacity," *Sustainability*, vol. 12, no. 21, p. 9239, 2020.

[27] Z. Zhang, C. Xu, and T. Lu, "Research on the protection area of lateral runway approach based on radar data[J]," *Journal of Safety and Environment*, vol. 20, no. 4, pp. 1391–1396, 2020.