Digital Communications and Networks 10 (2024) 1600–1610

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Toward edge-computing-enabled collision-free scheduling management for autonomous vehicles at unsignalized intersections

Ziyi Lu [b], Tianxiong Wu [b], Jinshan Su [a,*], Yunting Xu [b], Bo Qian [c], Tianqi Zhang [b], Haibo Zhou [b,*]

[a] Key Laboratory of Vibration Signal Capture and Intelligent Processing, Yili Normal University, Xinjiang, 835000, China
[b] School of Electronic Science and Engineering, Nanjing University, Nanjing, 210023, China
[c] Department of Mathematics and Theories, Peng Cheng Laboratory, Shenzhen, 518000, China

ARTICLE INFO

Keywords:
Unsignalized intersection
Automatic vehicle scheduling
Edge computing
Communication protocol
Computing power network

ABSTRACT

With the support of Vehicle-to-Everything (V2X) technology and computing power networks, the existing intersection traffic order is expected to benefit from efficiency improvements and energy savings by new schemes such as de-signalization. How to effectively manage autonomous vehicles for traffic control with high throughput at unsignalized intersections while ensuring safety has been a research hotspot. This paper proposes a collision-free autonomous vehicle scheduling framework based on edge-cloud computing power networks for unsignalized intersections where the lanes entering the intersections are undirectional, and designs an efficient communication system and protocol. First, by analyzing the collision point occupation time, this paper formulates an absolute value programming problem. Second, this problem is solved with low complexity by the Edge Intelligence Optimal Entry Time (EI-OET) algorithm based on edge-cloud computing power support. Then, the communication system and protocol are designed for the proposed scheduling scheme to realize efficient and low-latency vehicular communications. Finally, simulation experiments compare the proposed scheduling framework with directional and traditional traffic light scheduling mechanisms, and the experimental results demonstrate its high efficiency, low latency, and low complexity.

1. Introduction

With the rapid development of the Internet of Vehicles (IoV) technology, the automatic driving level of intelligent connected vehicles has gradually improved [1–3]. As the key component of the road system and the bottleneck of the road throughput, the order of Automatic Vehicles (AVs) at the intersection has become a research hotspot in the field of automatic driving [4]. Traditional traffic lights use advanced lane splitting to achieve collision-free intersection access by assigning passing periods to each entering lane. This mechanism is designed to facilitate drivers to understand and comply with traffic lights [5]. However, traditional traffic-light-controlled intersections are characterized by low traffic efficiency and prolonged waiting time due to the inevitable queuing overheads [6]. In addition, traditional traffic light intersections are also high accident locations because of potential driver operation errors. The existing solutions for AVs passing through intersections are mainly based on traffic light recognition. However, existing traffic light recognition schemes face many problems, including computation delay, rear lights of the front AVs, weather interference like rain and snow, weak light conditions, ambient light interference, and low resolution, which make the accidents of AVs at intersections frequent and have become a safety hazard [7].

With the development of Vehicle-to-Everything (V2X) technology, a centralized AV control solution becomes possible for intersections, which enables collision avoidance and cooperation between AVs [8]. Thus in the higher level of automatic driving, there is no need to consider the human-machine interaction or image recognition between AVs and the intersection traffic control system [9]. Therefore, a new AV scheduling system can be established for unsignalized intersections without the participation of human drivers, thus greatly improving the throughput of intersections and reducing the collision rate.

To realize the above system and solve the challenge of AV scheduling at unsignalized intersections, this paper proposes a new low-complexity and high-efficiency AV scheduling scheme and communi-

* Corresponding authors.
E-mail addresses: ziyilu@smail.nju.edu.cn (Z. Lu), tianxiongwu@smail.nju.edu.cn (T. Wu), sqsjs1968@aliyun.com (J. Su), yuntingxu@smail.nju.edu.cn (Y. Xu), boqian@pcl.ac.cn (B. Qian), tianqizhang@smail.nju.edu.cn (T. Zhang), haibozhou@nju.edu.cn (H. Zhou).

https://doi.org/10.1016/j.dcan.2024.03.001
Received 31 March 2023; Received in revised form 27 February 2024; Accepted 3 March 2024
Available online 7 March 2024

cation protocol by replacing traffic lights with real-time access and trajectory planning of arriving AVs and structurally reducing the scheduling delay through a dynamic communication protocol. Our scheduling scheme gives the optimal time to enter the intersection without incurring collisions by analyzing the demand of arriving AVs in real-time, completely removes the waiting lane system for the intersection to distinguish directions, and no longer restricts the direction of each lane, thus greatly increasing the intersection throughput while avoiding collisions [10]. In addition, to reduce the computing delay, we utilize the edge cloud to provide computing power through distributed computing and storage [11–13] and propose the Edge Intelligence Optimal Entry Time (EI-OET) algorithm. To address the emergency situations neglected by existing scheduling schemes, the method proposed in this paper performs a personalized scheduling scheme for AVs that need to be prioritized in emergency situations, enabling them to pass the unsignalized intersection first. Finally, inspired by [14], a communication system and the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) for Vehicular Medium Access Control (CVMAC) protocol are proposed. CVMAC combines the idea of CSMA/CA and the method of Time Division Multiple Access (TDMA) proposed in [15]. Our protocol makes a reasonable switch between these two protocols according to the node density to improve communication efficiency, reduce communication delay, and provide an efficient communication solution for the proposed scheduling scheme.

The main contributions of this study are summarized as follows:

1) We propose an AV scheduling mechanism for collision avoidance at unsignalized intersections, which changes the directional lane of intersections controlled by traffic lights into the lane with variable directions and allocates the entry time for collision avoidance by calculating the trajectory and occupation time of AVs entering the intersection.
2) We design the EI-OET algorithm supported by edge computing to solve the Absolute Value Programming (AVP) problem. It can conduct real-time calculations and feedback instructions for AVs entering the unsignalized intersection with low complexity and high scheduling efficiency.
3) To provide communication support for the scheduling architecture, a communication system using the CVMAC protocol is designed, which can effectively reduce communication delay and improve communication efficiency according to the access node density switching protocol.

The rest of this article is organized as follows. Section 2 introduces the related work. The modeling and problem-solving of the scheduling system are described in detail in Section 3. Section 4 presents the communication system and CVMAC protocol of the scheduling system. Section 5 shows the simulation results and analyses of the proposed scheduling system. Finally, Section 6 concludes this article.

2. Related work

2.1. Development of vehicle scheduling strategies for intersections

To improve the traffic efficiency and safety of intersections, numerous studies were conducted to improve the performance of the traffic light control system and various intelligent scheduling algorithms were developed [16]. Liang et al. [17] designed a double-layer signal system with additional hazard warning lights to reduce traffic congestion caused by accidents. Wang et al. [18] developed a decentralized framework with high communication efficiency for the traffic network with multiple intersections, and improved the traffic efficiency by exchanging traffic statistical information between adjacent intersections. Younes et al. [5] proposed an intelligent traffic light control algorithm by arranging the phase of each independent traffic light. However, these control methods are limited in improving the throughput of the intersection, because the traffic light has the inherent disadvantages of a single direction and long waiting time.

With the development of automatic driving and V2X technologies, the new AV scheduling methods based on unsignalized intersections have become a research focus of scholars. To eliminate traffic lights and ensure safety, many researchers have adopted the idea of cooperation between AVs [19], where AVs can enter the intersection simultaneously from different directions without the risk of collisions. For instance, Bian et al. [20] proposed a systematic method of connecting AVs at unsignalized intersections to enable them to pass in cooperation. Hang et al. [21] proposed a game theory decision-making framework, in which AVs interacted and made collaborative decisions. In view of the priority and interaction of AVs, Wang et al. [22] proposed a collision-free layered control strategy based on game theory for unsignalized intersections to improve traffic efficiency. These methods have effectively achieved collision avoidance but have not optimized the traffic efficiency and throughput of intersections.

In addition, many scholars have investigated the efficiency and throughput improvements of unsignalized intersections utilizing intelligent scheduling algorithms. For instance, Qian et al. [23] proposed an Alternately Iterative Descent Method (AIDM) to solve the scheduling problem by allocating the best entry time for each arriving AV so as to improve the AV throughput at unsignalized intersections. Based on the hierarchy of generalized critical turning points, the decision-making and route planning method was proposed for AVs at unsignalized intersections in [24] to increase traffic efficiency. Hang et al. [25] developed a new decision-making framework using the differential game method to avoid driving conflicts among AVs in unsignalized intersections. These methods have improved the traffic efficiency of the intersection to a certain extent while ensuring the safety of driving. However, existing scheduling methods have high requirements on computing power, communication delay and frequency, and storage support, which also pose high requirements on V2X communications.

In this paper, under the premise and considering the safety of collision-free intersection management, the main optimization goal is to improve the throughput of the intersection given computational complexity and communication requirements, design a low-complexity and efficient AV scheduling strategy for unsignalized intersections, and provide a secure communication system design for this scenario. This goal balances security and efficiency to provide practical and feasible solutions for computing and communication execution.

2.2. Communication protocols of vehicular networks

A commonly used V2X communication technology is Dedicated Short-Range Communications in the Vehicular Ad-hoc Network (VANET). With the development of V2X, academia has done a lot of research on network optimization and VANET communication protocols [26]. Cao and Lee [27] proposed a new TDMA MAC protocol for VANET, which accurately and adaptively optimized the length of each time frame by estimating and predicting the number of vehicles within the coverage of roadside units. Cao et al. [28] proposed a stochastic model to analyze and consider the security issues of malicious dual overhead attacks, and proved that the probability of initiating successful dual overhead attacks is affected by CSMA/CA. Zhang and Zhu [29] proposed an enhanced TDMA-based cooperative MAC protocol termed EVC-TDMA, where AVs dynamically selected relay nodes according to the relative speed and buffer length to transmit.

These MAC protocols are divided into 3 categories, namely, competition-based, competition-free and hybrid. They all aim to improve communication efficiency and reduce delay and error rates. However, none of the existing MAC protocols have been adjusted for the real-time environment. Therefore, the MAC protocol modified for the density of vehicle nodes in this paper can better meet the Quality of Service (QoS) requirements of AVs.

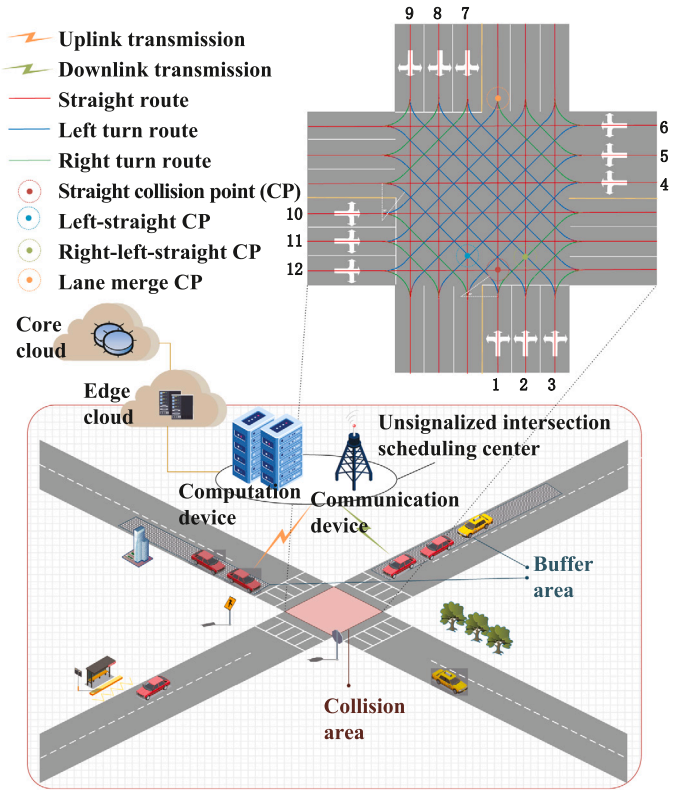

Fig. 1. Intelligent AV scheduling framework at an unsignalized intersection.

3. Automatic vehicle scheduling framework

This section designs an intelligent Automatic Vehicle Scheduling Framework (AVSF) for unsignalized intersections, which includes the mathematical model of AV behaviors when passing through an intersection. In Section 3.1, the system model is described. In Section 3.2, an occupancy time optimization problem is formulated. In Section 3.3, the proposed problem is solved with our EI-OET algorithm.

3.1. System model

The framework proposed in this paper is shown in Fig. 1. The computing and communication functions are mainly implemented by the Unsignalized Intersection Scheduling Center (UISC). An edge cloud mainly provides computing power and storage support, while a core cloud coordinates instructions for multi-intersection collaborations. The road is two-way 6 lanes, where AVs can turn left, go straight and turn right in any lane but can only drive into the corresponding lane. When this type of intersection is widely implemented, each lane can function as a left-turn, straight, or right-turn lane, which circumvents the necessity to change lanes before reaching the scheduling area.

Before entering the Collision Area (CA), an AV enters the Waiting Area (WA), where AVs adjust to a specified speed with constant acceleration. After receiving the instruction to drive in, the AV drives into the CA at the specified speed and on the illustrated track in Fig. 1. The width of the lane is $2a$ and the width of the CA is $12a$. The scheduling instructions are real-time and the scheduling algorithm is applicable to 1 AV. All superscripts L, S, and R in this paper refer to the directions selected by the current AV. Sometimes, superscripts are omitted to indicate general directions.

The Collision Point (CP) covers all areas where an AV may collide with other AVs in a certain direction. There are 12 CPs in any straight direction, 7, 9, or 11 CPs in each of the 3 left-turn directions from the inside of the road to the outside, and 5, 3, or 1 CPs in each of the 3 right-turn directions. The j-th CP on the 3 directions of lane i is marked as $CS_{i,j}^{L}, CS_{i,j}^{S}, CS_{i,j}^{R}$, $i = 1, 2, \cdots, 12$. The radius of the CP is r_0, which exceeds half of the AV length to avoid collisions as well as maximize the throughput.

AVs enter the CA at different speeds in the preset range, which is $v \in [15, 30]$ km/h for the left and right turns, l_i^L, l_i^R and $v \in [15, 60]$ km/h for the straight lanes l_i^S. The maximum driving speed of AVs in the WA is set to v_{max}. The time for AVs to arrive WA of 12 lanes is $t_0 = (t_1^0, \cdots, t_{12}^0)'$, and the entering speed is $v_0 = (v_1^0, \cdots, v_{12}^0)'$. The minimum passing time of AVs in the WA can be calculated as $t_{min} = (t_1^{min}, \cdots, t_{12}^{min})'$ with the maximum acceleration and deceleration $\hat{a}_{WA} \in [\hat{a}_{min}, \hat{a}_{max}]$ in the WA. The length of WA is l_{WA}, so t_i^{min} is to accelerate to v_{max}, maintain, and then decelerate to the CA's entering speed at the maximum deceleration in advance, which is $t_i^{min} = (v_{max} - v_i^0)/\hat{a}_{max} + (v_i - v_{max})/\hat{a}_{min} + l_{WA}/v_{max} - [v_{max}^2 - (v_i^0)^2]/2\hat{a}_{max}v_{max} - (v_i^2 - v_{max}^2)/2\hat{a}_{min}v_{max}$.

Then the UISC will return a speed instruction for the AV to enter CA. Considering the safety of AVs, due to the existence of possible risk factors such as communication errors, brake wear, AV vehicle runaways, there still exists a very low collision probability at the intersection, so reducing loss is also an important consideration. When the traffic increases, the probability of AV collisions increases. To reduce the collision loss, the AV speed should be restricted. When the traffic is low (for example, there are only 1 or 2 AVs), even if there is an accident, the probability of collision is quite low, the expectation of the loss is very small, and the speed can be increased to the upper limit. Therefore, we set the entering speed to $v = (v_1, \cdots, v_{12})'$, which is adjusted and optimized according to queue length in WA $n_0 = (n_1^0, \cdots, n_{12}^0)'$. When the number of AVs in the WA exceeds threshold N_{WA}^{max}, the entering speed will be the minimum value. When it is less than the threshold N_{WA}^{min}, the entry speed will be the maximum value. Otherwise, a subtraction function will be performed on v_i. The AVs enter the CA at time $t = (t_1, \cdots, t_{12})'$.

3.2. Problem formulation based on occupancy time

In our scheduling model, 3 types of occupancy time based constraints are used to avoid collisions, that is, the time constraint of AVs passing WA to enter CA, the safe distance with the previous AV in the current lane, and the collision prevention with the scheduled AVs at the CPs in CA. Therefore, the time constraint of AVs passing WA to enter CA is $t \geq t_0 + t_{min}$. Considering the safe distance with the previous AV in the current lane, there are n_i^T, n_i^S, n_i^R AVs in lane l_i^T, l_i^S, l_i^R, respectively, n_i in total. $l_i^{(j)} \in \{l_i^S, l_i^L, l_i^R\}$ is the lane direction of the j-th AV scheduled in lane l_i, whose entering speed is $v_i^{(j)}$ and the entry time is $t_i^{(j)}$. Then the last AV in the lane enters CA at $t_i^{(N_i)} \triangleq \max\{t_i^{(j)}\}$, $\forall j = 1, \cdots, n_i$. $t^{(N)} \triangleq (t_1^{(N_1)}, \cdots, t_{12}^{(N_{12})})'$ indicates the last scheduled time to enter the CA, and a 12-dimensional vector $t_S \triangleq (t_s, \cdots, t_s)'$ represents the safety time constraint to guarantee the safe distance. The constraint for the safe distance is $t \geq t^{(N)} + t_S$.

Subsequently, we consider collision prevention with the scheduled AVs at CPs. The relationship between the collision lanes and occupancy time at typical CPs is shown in Table 1.

The time occupied by lane l_i^R at $CS_{i,j}^S$ is $\sigma_{iR}^{iS,j}$. The arc center angle of the left and right turning AVs entering and leaving CA is $\pi/5$, so the arc driving time is $T_i^1 = \pi a/2v_i$. The left and right turning AVs take $T_i^2 = a/\sqrt{2}v_i$ to leave the arc to the first CP or to leave the last CP from the arc, and they spend $T_i^3 = \sqrt{2}a/v_i$ to drive between two CPs. The time spent by a straight-driving AV to drive from the start point to the first CP and or between CPs is $T_i^4 = 2a/v_i$. The occupied time period of all AVs within CPs extends forward and backward at the time point in the center of CPs, and the extended length is $T_i^5 = r_0/v_i$. The curve extension length of left or right-turning AVs driving through the first or last CP can be simplified on the premise of safety by adding r_0, so $\sigma_i = [t_i + \sum T_i^k - T_i^5, t_i + \sum T_i^k + T_i^5]$. To prevent collisions, the occupancy

Table 1
Mapping between typical CPs and collision lanes.

CP	Collision lanes	Occupation time
$CS^S_{1,1}$	l^S_1, l^S_{12}	$\sigma^{1S,1}_{1S}, \sigma^{1S,1}_{12S}$
$CS^S_{1,2}$	l^S_1, l^L_2, l^L_4	$\sigma^{1S,2}_{1S}, \sigma^{1S,2}_{2L}, \sigma^{1S,2}_{4L}$
$CS^S_{1,12}$	l^S_1, l^R_4, l^L_{10}	$\sigma^{1S,12}_{1S}, \sigma^{1S,12}_{4R}, \sigma^{1S,12}_{10L}$
$CS^S_{2,2}$	l^R_1, l^S_2, l^L_3	$\sigma^{2S,2}_{1R}, \sigma^{2S,2}_{2S}, \sigma^{2S,2}_{3L}$
$CS^L_{1,1}$	l^L_1, l^L_4, l^R_{12}	$\sigma^{1L,1}_{1L}, \sigma^{1L,1}_{4L}, \sigma^{1L,1}_{12R}$
$CS^L_{1,7}$	l^L_1, l^S_4, l^R_7	$\sigma^{1L,7}_{1L}, \sigma^{1L,7}_{4S}, \sigma^{1L,7}_{7R}$
$CS^L_{2,1}$	l^R_1, l^L_2, l^S_{12}	$\sigma^{2L,1}_{1R}, \sigma^{2L,1}_{2L}, \sigma^{2L,1}_{12S}$
$CS^R_{1,1}$	l^R_1, l^L_2, l^S_{12}	$\sigma^{1R,1}_{1R}, \sigma^{1R,1}_{2L}, \sigma^{1R,1}_{2L}$

time of a CP should not intersect with each other. Take $CS^S_{1,1}$ as an example, $\sigma^{1S,1}_{1S} \cap \sigma^{1S,1}_{12S} = (t_1 + 2a/v_1 - T^5_1, t_1 + a/v_1 + T^5_1] \cap (t_{12} + 8a/v_{12} - t^5_{12}, t_{12} + 8a/v_{12} + T^5_{12}] = \phi$, that is, $|(t_1 + 2a/v_1) - (t_{12} + 8a/v_{12})| = |t_1 - t_{12} + 2a/v_1 - 8a/v_{12}| \geq r_0/v_1 + r_0/v_{12}$. Other CP constraints can be obtained similarly.

The collision prevention constraints are listed for 12 lanes in pairs. Finally, the 234 constraint inequalities are represented by $|At + b| \geq c, A \in \mathbb{R}^{234\times12}$, $t \in \mathbb{R}^{12}$, $b, c \in \mathbb{R}^{234}$, but most of them will not occur in the actual scheduling, because an AV can only choose one of the routes. We replace t_i with $t_i{}^{(j)}$ and remove duplicate constraints to get complete constraints as $|A_i t + b_i{}^{(j)}| \geq c_i$, $\forall j = 1, \cdots, n_i$, $i = 1, \cdots, 12$. Take l^R_2 as an example, $t \triangleq (t_1, t^{(j)}_2, \cdots, t_{12})'$, $j \in \{1, \cdots, n_2\}$, $l_2{}^{(j)} = l^R_2$, and we get (1).

$$
\begin{aligned}
&|t^{(j)}_2 - t_3 + \left(\pi + \sqrt{2}\right) a/2v^{(j)}_2 - \left(\pi + \sqrt{2}\right) a/2v_3| \geq r_0/v^{(j)}_2 + r_0/v_3\\
&or\ |t^{(j)}_2 - t_3 + \left(\pi + 3\sqrt{2}\right) a/2v^{(j)}_2 - 3a/v_3| \geq r_0/v^{(j)}_2 + r_0/v_3\\
&|t^{(j)}_2 - t_8 + \left(\pi + 2\sqrt{2}\right) a/v^{(j)}_2 - \left(\pi + 8\sqrt{2}\right) a/v_8| \geq r_0/v^{(j)}_2 + r_0/v_8\\
&|t^{(j)}_2 - t_9 + \left(\pi + 3\sqrt{2}\right) a/2v^{(j)}_2 - \left(\pi + 19\sqrt{2}\right) a/2v_9| \geq r_0/v^{(j)}_2 + r_0/v_9\\
&|t^{(j)}_2 - t_{11} + \left(\pi + 2\sqrt{2}\right) a/v^{(j)}_2 - 14a/v_{11}| \geq r_0/v^{(j)}_2 + r_0/v_{11}\\
&|t^{(j)}_2 - t_{12} + \left(\pi + \sqrt{2}\right) a/2v^{(j)}_2 - 11a/v_{12}| \geq r_0/v^{(j)}_2 + r_0/v_{12}
\end{aligned}
\tag{1}
$$

For l^R_2, we can get its constraint matrix $|A^R_2 t + b^{(j)}_{2R}| \geq c^R_2$, $j \in \{1, \cdots, n_2\}$, $l^{(j)}_2 = l^R_2$, where

$$
A^R_2 \triangleq \begin{pmatrix}
0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1
\end{pmatrix}
c^R_2 \triangleq \begin{pmatrix}
r_0/v^{(j)}_2 + r_0/v_3\\
r_0/v^{(j)}_2 + r_0/v_8\\
r_0/v^{(j)}_2 + r_0/v_9\\
r_0/v^{(j)}_2 + r_0/v_{11}\\
r_0/v^{(j)}_2 + r_0/v_{12}
\end{pmatrix}
$$

$$
b^{(j)}_{2R} \triangleq \begin{pmatrix}
-t^{(j)}_2 - \left(\pi + \sqrt{2}\right) a/2v^{(j)}_2 + \left(\pi + \sqrt{2}\right) a/2v_3, l_3 = l^L_3\\
or - t^{(j)}_2 - \left(\pi + 3\sqrt{2}\right) a/2v^{(j)}_2 + 3a/2v_3, l_3 = l^S_3\\
-t^{(j)}_2 - \left(\pi + 2\sqrt{2}\right) a/v^{(j)}_2 + \left(\pi + 8\sqrt{2}\right) a/v_8\\
-t^{(j)}_2 - \left(\pi + 3\sqrt{2}\right) a/2v^{(j)}_2 + \left(\pi + 19\sqrt{2}\right) a/2v_9\\
-t^{(j)}_2 - \left(\pi + 2\sqrt{2}\right) a/v^{(j)}_2 + 14a/v_{11}\\
-t^{(j)}_2 - \left(\pi + \sqrt{2}\right) a/2v^{(j)}_2 + 11a/v_{12}
\end{pmatrix}
$$

Similarly, we get the constraints of other lanes. Finally, the optimization problem is expressed as $P1$:

$$P1: min\ e^T t \tag{2a}$$

$$s.t.\ t \geq t_0 + t_{min} \tag{2b}$$

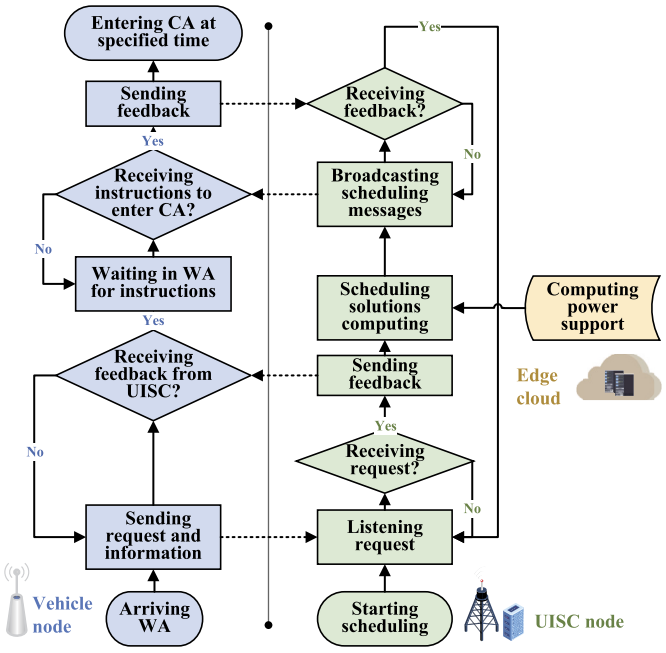

Fig. 2. Workflow of communications between vehicle nodes and UISC nodes.

$$t \geq t^{(N)} + t_S \tag{2c}$$

$$|A_i t + b^{(j)}_i| \geq c_i,\ \forall j = 1, \cdots, n_i, i = 1, \cdots, 12 \tag{2d}$$

where $e = (1, \cdots, 1)' \in \mathbb{R}^{12}$. Optimization objective (2a) is to minimize the time for AVs to enter CA. (2b) is the time constraint of AVs passing WA to enter CA. (2c) is the safe distance from the previous AV in the current lane. (2d) is the collision prevention constraint with the scheduled AVs at the CPs in CA.

3.3. Edge intelligence optimal entry time scheduling system

In our scheduling framework based on occupancy time, the UISC consisting of computing and communication equipment is the key component of the scheduling process. The UISC also helps reduce packet loss rates in cases of poor channel quality to improve algorithm robustness by feedback retransmitting. The computing power and storage required for scheduling are supported by the edge cloud. As shown in Fig. 2, the AV first transmits relevant information, such as arrival time, heading direction, and arrival speed to the roadside node, and then the roadside node communicates with the UISC to request driving instructions until receiving feedback from it. After receiving the request, the UISC computes and returns the feedback to the vehicle node for confirmation in case of packet loss. Then the UISC, whose computing power is supported by the edge cloud, combines the received information to calculate the optimal entry time and then transmits the scheduling instructions via roadside nodes to the AVs until receiving the feedback from them. Finally, the AVs drive into CA according to the instructions. To make the EI-OET algorithm executable, we make the following assumptions:

a) All AVs have communication capabilities and are equipped with an on-board Unit. It is reasonable to ignore the movement within each unit information duration in the transmission process because the messages sent by AVs to the UISC are usually in the order of milliseconds [30].
b) The transmission delay can be ignored due to the maximum AV movement of 0.035 m when the average delay of packet transmission is 1.6 ms [31].
c) The AVs and UISC adopt DSRC for data transmission, which is based on the IEEE 802.11p [32].

The edge intelligence scheduling mechanism is proposed to solve optimization problem $P1$, where for each UISC node, only 1 AV needs to be scheduled for each scheduling slot. To ensure this, each scheduling slot is set to safety time t_s. Since $P1$ is an AVP problem [33], UISC can assign the optimal solution with linear computational complexity.

To reconstruct the constraint matrix, $j_i^S = 1, \cdots, n_i^S$ represents the AV that has been scheduled in l_i^S. Take l_2^R as an example, $t \triangleq (t_1^{(j_1)}, t_2, \cdots, t_8^{(j_8)})'$, $j_i = 1, \cdots, n_i$, $i = 1, \cdots, 12$, and we get (3).

$$
\begin{aligned}
&|t_2 - t_3^{\left(j_3^L\right)} + \left(\pi + \sqrt{2}\right) a/2v_2 - \left(\pi + \sqrt{2}\right) a/2v_3^{\left(j_3^L\right)}| \\
&\quad \geq r_0/v_2 + r_0/v_3^{\left(j_3^L\right)} \\
&|t_2 - t_3^{\left(j_3^S\right)} + \left(\pi + 3\sqrt{2}\right) a/2v_2 - 3a/v_3^{\left(j_3^S\right)}| \geq r_0/v_2 + r_0/v_3^{\left(j_3^S\right)} \\
&|t_2 - t_8^{\left(j_8^L\right)} + \left(\pi + 2\sqrt{2}\right) a/v_2 - \left(\pi + 8\sqrt{2}\right) a/v_8^{\left(j_8^L\right)}| \\
&\quad \geq r_0/v_2 + r_0/v_8^{\left(j_8^L\right)} \\
&|t_2 - t_9^{\left(j_9^L\right)} + \left(\pi + 3\sqrt{2}\right) a/2v_2 - \left(\pi + 19\sqrt{2}\right) a/2v_9^{\left(j_9^L\right)}| \\
&\quad \geq r_0/v_2 + r_0/v_9^{\left(j_9^L\right)} \\
&|t_2 - t_{11}^{\left(j_{11}^S\right)} + \left(\pi + 2\sqrt{2}\right) a/v_2 - 14a/v_{11}^{\left(j_{11}^S\right)}| \geq r_0/v_2 + r_0/v_{11}^{\left(j_{11}^S\right)} \\
&|t_2 - t_{12}^{\left(j_{12}^S\right)} + \left(\pi + \sqrt{2}\right) a/2v_2 - 11a/v_{12}^{\left(j_{12}^S\right)}| \geq r_0/v_2 + r_0/v_{12}^{\left(j_{12}^S\right)}
\end{aligned} \tag{3}
$$

For l_2^R, $m_2^R = n_3^L + n_3^S + n_8^L + n_9^L + n_{11}^S + n_{12}^S$ and $|t_2 e + \bar{b}_2| \geq \bar{c}_2$, where

$$
\bar{b}_2 \triangleq \begin{pmatrix}
-t_3^{\left(j_3^L\right)} + \left(\pi + \sqrt{2}\right) a/2v_2 - \left(\pi + \sqrt{2}\right) a/2v_3^{\left(j_3^L\right)} \\
-t_3^{\left(j_3^S\right)} + \left(\pi + 3\sqrt{2}\right) a/2v_2 - 3a/2v_3^{\left(j_3^S\right)} \\
-t_8^{(j_8)} + \left(\pi + 2\sqrt{2}\right) a/v_2 - \left(\pi + 8\sqrt{2}\right) a/v_8^{\left(j_8^L\right)} \\
-t_9^{(j_9)} + \left(\pi + 3\sqrt{2}\right) a/2v_2 - \left(\pi + 19\sqrt{2}\right) a/2v_9^{\left(j_9^L\right)} \\
-t_{11}^{(j_{11})} + \left(\pi + 2\sqrt{2}\right) a/v_2 - 14a/v_{11}^{\left(j_{11}^S\right)} \\
-t_{12}^{(j_{12})} + \left(\pi + \sqrt{2}\right) a/2v_2 - 11a/v_{12}^{\left(j_{12}^S\right)}
\end{pmatrix}_{m_2^R \times 1}
$$

$$
\bar{c}_2 \triangleq \begin{pmatrix}
r_0/v_2 + r_0/v_3^{(j_3{}^L)} \\
r_0/v_2 + r_0/v_3^{(j_3{}^S)} \\
r_0/v_2 + r_0/v_8^{(j_8{}^L)} \\
r_0/v_2 + r_0/v_9^{(j_9{}^L)} \\
r_0/v_2 + r_0/v_{11}^{(j_{11}{}^S)} \\
r_0/v_2 + r_0/v_{12}^{(j_{12}{}^S)}
\end{pmatrix}_{m_2{}^R \times 1}
$$

Next, we can obtain the complete $|t_i e + \bar{b}_i| \geq \bar{c}_i$, $j = 1, \cdots, n_i$, $i = 1, \cdots, 12$, and transform $P1$ into

$$
\min\ t_i \tag{4a}
$$

$$
s.t.\ t_i \geq t_i^0 + t_i^{min} \tag{4b}
$$

$$
t_i \geq t_i^{(N_i)} + t_s \tag{4c}
$$

$$
|t_i e + \bar{b}_i| \geq \bar{c}_i,\ j = 1, \cdots, n_i,\ i = 1, \cdots, 12 \tag{4d}
$$

To further simplify the problem, we let $\hat{t}_i = \max\{t_i^{(N_i)} + t_s,\ t_i^0 + t_i^{min}\}$, then the optimization problem is expressed as $P2$:

$$
P2\text{: } \min\ t_i \tag{5a}
$$

$$
s.t.\ t_i \geq \hat{t}_i \tag{5b}
$$

$$
|t_i e + \bar{b}_i| \geq \bar{c}_i,\ j = 1, \cdots, n_i,\ i = 1, \cdots, 12 \tag{5c}
$$

where $\bar{b}_i(k)$ is the k-th line of $\bar{b}_i$, $\bar{c}_i(k)$ is the k-th line of $\bar{c}_i$, $k = 1, \cdots, m_i$. (5c) is equivalent to $t_i \geq \bar{c}_i(k) - \bar{b}_i(k)$ or $t_i \leq -\bar{c}_i(k) - \bar{b}_i(k)$, $\forall k = 1, \cdots, m_i$. Our proposed EI-OET is summarized by Algorithm 1.

Algorithm 1: Edge Intelligence Optimal Entering Time Scheduling.

Input: entering lane and direction D, entering time t_0, safety time t_s, entering speed v_i^0, WA memory, time memory, speed memory for scheduled AVs.
Output: entering time t_i, current time vector t and CA entering speed v_i for each AV.

```
while there are still vehicles to be scheduled do
    Update WA memory and time memory;
    if t updates to the new slot then
        Calculate v_i according to traffic load;
    end
    Calculate t_i^(N_i);
    for i = 2 to 12 do
        Input t_i^0, v_i^0 and calculate t̂_i, b̄_i, c̄_i;
        Set t_opt = t̂_i;
        if c̄_i(1) − b̄_i(1) < t̂_i < c̄_i(1) − b̄_i(1) then
            t_opt = c̄_i(1) − b̄_i(1);
        end
        for k = 2 to m_i do
            if −c̄_i(k) − b̄_i(k) < t_opt < c̄_i(k) − b̄_i(k) then
                if −c̄_i(k−1) − b̄_i(k−1) < c̄_i(k) − b̄_i(k) < c̄_i(k−1) − b̄_i(k−1) then
                    t_opt = c̄_i(k−1) − b̄_i(k−1);
                else
                    t_opt = c̄_i(k) − b̄_i(k);
                end
            end
        end
        Update t_i^* = t_opt;
    end
end
```

Further, an Edge Cloud Computing Coordination Protocol (ECCCP) is proposed to realize an edge cloud computing power system that ensures sufficient computing power support and helps reduce packet loss rates in cases of poor channel quality to improve algorithm robustness. The computing and storage resource requirements in each scheduling period are allocated to the edge clouds for distributed storage and computing. Then the core cloud allocates resources for multiple unsignalized intersections, where the scheduling of one intersection is performed by the EI-OET algorithm in the distributed structure. The uniform feedback of the computing results provides UISC with computing power and storage support, thus building the computing power network based on edge clouds. The specific steps are as follows:

a) Collecting messages: The UISC receives messages about AVs entering the WA from roadside nodes, uploads them to an edge cloud, and stores them in the edge cloud memory.
b) Computing data: The edge clouds assign computing tasks and compute the optimal entering time and speed for the newly arriving AVs according to the EI-OET algorithm.
c) Upload data for coordination: Upload the computed instructions to the core cloud for coordination among various edge clouds.
d) Download for instructions: The core cloud confirms the non-confliction between instructions from adjacent edge clouds for different intersections and broadcasts feedback.

e) Transmit instructions to AVs: Edge clouds receive permission from the core cloud and then transmit instructions to AVs via UISC.

3.4. Personalized scheduling considering priority

To meet the emergency requirements during scheduling such as requests from ambulances and fire trucks, high-priority AVs can reserve lanes. After receiving the request, the UISC will instruct AVs in this direction to leave a green lane with high priority in advance, and the high-priority AV will drive into the green lane.

Personalized EI-OET (PEI-OET) is presented in Algorithm 2, where we add the priority request input and the WA driving lane output, and set the WA storage queue to record the lane and sequence of AVs entering the WA. Before starting a scheduling round, we check whether there is any priority adjustment request. If so, priority scheduling is executed for the AVs in the green lane.

Algorithm 2: Personalized Edge Intelligence Optimal Entering Time Scheduling.

```
while there are still vehicles to be schedules do
    if exist request from priority lane l_p then
        Run instructions for existing AVs in WA of lane l_p to switch to
          other lanes;
        Update WA, time and speed memory;
        Calculate v_i according to traffic load and t_i^(N_i);
        while WA[l_p]! = 0 do
            Operate EI-OET for AVs in WA[l_p];
        end
    else
        Run EI-OET;
    end
end
```

4. Communication system and protocol for framework implementation

The AV scheduling process in Section 3 involves various V2I communications. This section proposes a communication system and protocol of the unsignalized intersection, providing a communication implementation for the AV scheduling framework designed in Section 3. This method combines the idea of CSMA/CA with the method of TDMA proposed in [15] to build the CVMAC protocol. The proposed CVMAC protocol dynamically adjusts the working mode according to how busy the channel is to solve the problem of node information access conflicts, which maximizes the efficiency of communications and effectively reduces the communication delay when the node density is low.

4.1. Communication system model

As shown in Fig. 3, the communication system consists of the vehicle nodes, UISC node, and the time slot synchronization node, and completes data transmission according to the CVMAC protocol. The system determines the protocol type according to the estimated node density, uses CSMA/CA when the density is low, and uses Vehicular MAC (VeMAC) for transmission when the density is high. The standard for distinguishing the high and low access densities is determined by the number of AVs to be scheduled when the throughput of the two protocols is equal. When the actual number of access AVs exceeds the standard, the access density is considered high; otherwise, the access density is considered low.

The proposed communication system design is applied to the four-way 12-lane unsignalized intersection scenario in Section 3. The roadside nodes function as AV information collectors, the UISC node serves as the roadside computing and communication unit, and the time slot synchronization node leverages outdoor GPS modules when the actual intersection is in a position where GPS signals are abnormal.

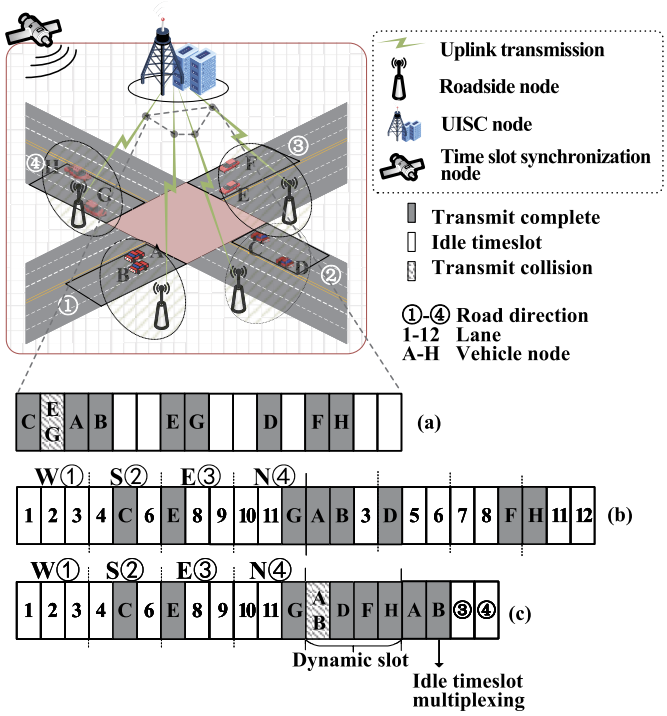

Fig. 3. Communication system structure and time slot design.

The roadside node is equipped with the following processing units:

- A listening unit is used to listen to whether the channel is idle when the scheduling demand is triggered at the unsignalized intersection, analyze the access density from the data packets received, and determine the protocol type according to the density.
- A CSMA/CA scheduling unit is used to send the data packet containing its own AV information and demand information to the UISC node when the access density is low, and receive feedback from the UISC node. When the request fails, a collision backoff algorithm is used to resend the request, and the number of failures is added to the data packet.
- A VeMAC transmission unit is used to receive the idle time slot identification sent by the UISC node when the access density is high, select the suitable time slot according to the direction of its AVs, send information to the UISC node, add the number of the request failure times into the data packet, and feedback it to the UISC node.

The roadside node data frame structure includes the header address (Header_ADDR), which is mainly used to confirm the channel of the data packet and provide transmission authorizations. The AV data packet (AV_Data) is mainly used to carry the information of its associated AVs. In this scenario, the generated AV information, the request data packet (Request_Data), which is mainly the target direction of the AV, and the channel state information generated by the MAC protocol all need to be sent to UISC. In addition, security proof information (Security_Message) is added to prevent external interference.

The UISC node is equipped with the following processing units:

- An access density analysis unit is used to analyze the collision backoff information or channel utilization information from the data packets of the roadside nodes, obtain the access density, send the access density information to the roadside nodes, and guide the roadside nodes to adopt a specific transmission mode.

- A vehicle request processing unit is used to analyze the request information in the data packet of the AV, calculate the time when an AV enters CA, and send it to the waiting AV.

Compared with the roadside node data frame structure, apart from the normal packet transmission time frame, the UISC node also needs to broadcast the current time slot allocation and reservation status. The time synchronization node mainly verifies the header address data, time information packet, and security packet to ensure high-quality real-time synchronization of time slots.

4.2. Communication protocol of roadside nodes and UISC

As shown in Fig. 3, 2 protocol access methods and 1 variant are based on CSMA/CA with binary exponential backoff and VeMAC with TDMA, respectively. Fig. 3(a) shows the CSMA/CA protocol, where AVs in 4 directions are in the arriving sequence. If two AVs, for example, AVs E and G, have access conflicts in the same direction at the same time slot, their access time slots are adjusted according to the backoff rule.

According to VeMAC, Figs. 3(b) and (c) set the time frame in advance. In 4 directions, each lane is set with a time slot of 25 ms. Thus the size of one frame is 300 ms, and the time synchronization signal is checked every 10 frames. AVs from each direction receive channel status information every 3 frames in advance and select the nearest frame and time slot to wait for transmission. In addition, Fig. 3(c) shows the low-density case where access conflicts are infrequent. The time-slot distribution of four-way 12 lanes can be simplified to that of four-way 4 lanes, which can substantially reduce the overheads in large time frames. After the fixed 4-direction slots are allocated, dynamic slots are allocated according to the node density. When the node density is high, to cope with large communication overheads, the number of dynamic time slots is increased. When the density turns medium, the communication overhead declines, and the number of time slots in each direction can be reduced. When the lanes in the same direction compete for time slots, AVs A and B can temporarily occupy the idle time slots in other directions.

To confirm the performance cut-off point between CSMA/CA and VeMAC, we have deduced the mathematical model of the two protocols and found that when the throughput of CSMA/CA and VeMAC is close, the cut-off point appears. When the number of AVs is less than a certain n, CSMA/CA is more suitable; otherwise, the performance of VeMAC is better.

Assuming that n competing AVs can send data at the same time and the probability of data packet transmission p follows a Poisson distribution, the probability of successful transmission P_s and the probability of collision P_c can be expressed as

$$P_s = p*(1-p)^{n-1},\ P_c = 1 - P_s \tag{6}$$

where n represents the number of AVs. The expected number of collisions before a successful transmission, $E[N_c]$, namely the estimation of the number of times to wait for transmission can be calculated as follows

$$E[N_c] = P_c/P_s = 1/[p*(1-p)^{n-1}] - 1 \tag{7}$$

The time involved in each transmission includes Distributed Inter-Frame Spacing (DIFS) time D, which is the response time to detect whether there is an idle time slot, also known as DIFS transmission delay, and file transmission time s. Thus the transmission time of a packet is $T_{trans} = s + D$ normally. T_{prop} indicates the time to wait for a successful transmission before its own turn to start the transmission, which is

$$T_{prop} = E[N_c]*(s+D) \tag{8}$$

At the same time, the average backoff time caused by collisions can be calculated as follows. The average backoff time of retransmission can be calculated by the exponential backoff algorithm. Assuming that the mean value of backoff time is E_t and the mean value of backoff times is m, so the average backoff time of conflict retransmission is

$$T_{queue} = E_t \times (1 + 2 + 2^2 + \cdots + 2^{m-1}) = E_t \times (2^m - 1) \tag{9}$$

Therefore, the total time of a successful transmission can be expressed as

$$T_{total} = T_{prop} + T_{trans} + T_{queue} \tag{10}$$

The throughput of the CSMA/CA protocol is

$$G_{CSMA} = n * S/T_{total} \tag{11}$$

where S represents the size of the packet. In CSMA/CA, the core TDMA idea in VeMAC is also used. The number of lanes is 12 in 4 directions. Therefore, 12 small timeslots from a large timeslot as the transmission interval for AVs coming from different directions. N indicates how many small timeslots a large time frame is divided into, which is set to 12 in this protocol, so the throughput of VeMAC can be expressed as

$$G_{VeMAC} = n * S/N \tag{12}$$

To maximize the protocol performance, we make $G_{CSMA} = G_{VeMAC}$ and derive the number of AVs for switching the protocol:

$$n = \frac{\log(s+d)/[12p - E \times (2^m - 1)p]}{\log(1-p)} \tag{13}$$

To further realize the above protocol switching, in the CSMA/CA scheduling unit of roadside nodes, the binary exponential backoff algorithm is used to deal with access conflicts. When the number request failure times reaches a threshold, the UISC triggers a switch from CSMA/CA to VeMAC. When using VeMAC for transmission, the UISC node carries out a statistical average on the slot occupancy in multiple frame cycles. If the slot occupancy is less than the threshold, UISC switches to CSMA/CA.

The flow chart of the roadside node executing the CVMAC protocol is shown in Fig. 4(a). First, the roadside node triggers the scheduling requirement listening for whether the channel is idle. If the analyzed data in the packet demonstrates the node access density is low, the roadside node sends the data packet containing AV information and request information to the UISC node. A transmission is completed when the ACK feedback from UISC is received. If the request fails for k times, wait $(2^k + T_r)$ ms, where T_r represents a randomly generated period of time within 50 ms if $k \leq 9$. The number of failed requests is set as $k \leq 16$. If k equals 16 for 3 consecutive times, k is cleared for retransmission. If the number of conflicting nodes and the number of nodes with $k = 16$ increase, UISC will calculate the number of AVs and n for protocol switching. Because the transmission probability is affected by the times of transmission back-offs with a range of $0 - 16$, there will be a fixed range of influence on the n value at different times. To determine the number of AVs, n will be recalculated every time when the protocol is switched.

When the analyzed data in the packet demonstrates the node density is high, VeMAC is executed to transmit the data. The second control channel module is opened to receive channel state data packets and select the appropriate time slot according to the direction to send information. The UISC node will take the slot occupancy P_{slot}. $P_{slot} \leq 50\%$ indicates that the current number of AVs is low and the roadside node switches to the low-density protocol. When AVs enter dense traffic, it is necessary to dynamically expand the size of the frame and time slot. In addition, the congestion direction can occupy the idle time slots of other directions. If the occupancy rate of each time slot is high, the directional time slot of the next time frame is selected for collision avoidance. The k value and the frequency of occurrence are recorded as the feedback

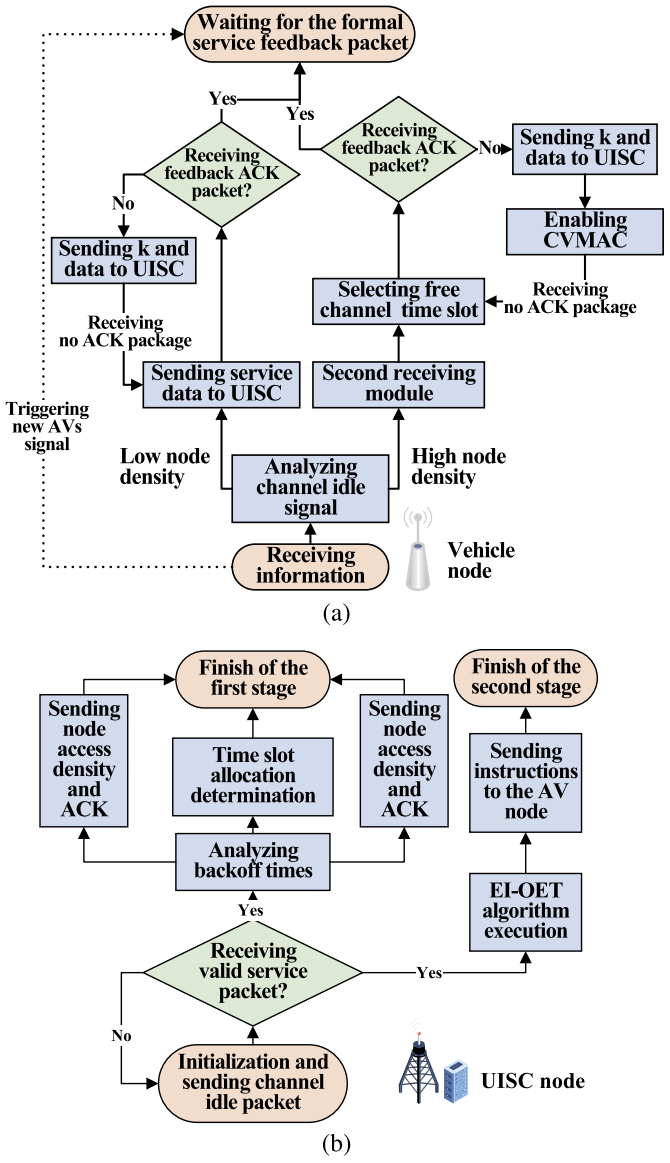

Fig. 4. (a) CVMAC workflow of roadside nodes. (b) CVMAC workflow of UISC nodes.

to UISC, which is the indicator of the congestion degree in the same direction. If AVs in the same direction collide in a time slot due to transmission delay or lane sharing, VeMAC should expand the number of directional time slots in the frame to improve throughput.

The flow chart of the UISC node executing the CVMAC protocol is shown in Fig. 4(b). The UISC node first analyzes the AV information packet data of the roadside node, and obtains the node density by the number of AV backoffs or the slot occupancy. If the density is low, this information will be sent and the roadside nodes are instructed to adopt the corresponding transmission method. Otherwise, the UISC node turns on the second control channel transmission module, uses the VeMAC, and broadcasts the free time slots in all directions and node density information to AVs. Finally, the UISC node analyzes the request data packet of AVs, calculates the time when an AV enters the unsignalized intersection according to the EI-OET algorithm, and feedbacks it to the AV.

5. Experiments

In this section, we mainly carry out simulation experiments on the EI-OET algorithm and describe the environment parameters and system operations. The proposed EI-OET is compared with baselines that distinguish the lane direction with the same scheduling method and a baseline in the traditional traffic light scene. It is confirmed that EI-OET has significant advantages in throughput, delay, and throughput stability when the direction of traffic changes.

Table 2
Key parameters employed in experiments.

Para.	Value	Interpretation
λ	Default	AV flow index.
a	1.5 m	Half lane width
r_0	2 m	Radius of CP
t_s	0.5 s	Time of safety distance
v_{max}	80 km/h	Maximum speed in WA
$\hat{a}_{max}$	4 m/s^2	Maximum acceleration in WA
$\hat{a}_{min}$	-3 m/s^2	Minimum acceleration in WA
l_{WA}	80 m	Length of WA
N_{WA}^{max}	24	High threshold to change speed
N_{WA}^{min}	8	Low threshold to change speed

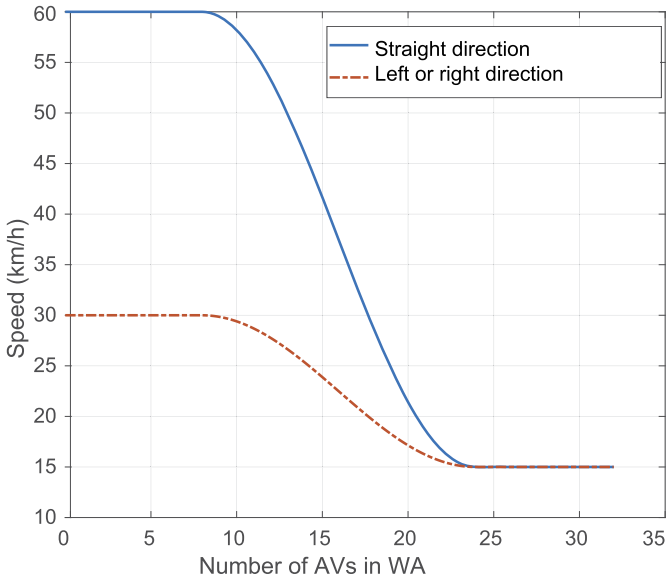

Fig. 5. AV speed assignment function.

5.1. Experimental parameters and implementation settings

The AV flow in the simulation environment is generated by SUMO [34], which includes the trajectory and flow density in real-time. Combined with the data generated by SUMO, the simulations are carried out on an Intel(R) Core(TM) i7-12700H with MATLAB 2022A. The radius of UISC for DSRC is set to 500 m [31]. To make the experiment typical and practical, important parameter values are shown in Table 2.

In addition, due to the different safety speed requirements of countries on AVs turning at intersections and throughput requirements, we adjust the high and low thresholds and the speed assignment function in the simulation to find optimal parameter values, as shown in Fig. 5. The high threshold N_{WA}^{max} of employing the minimum speed 15 km/h is 24 AVs in WA and the low threshold N_{WA}^{min} of employing the maximum speed is 8 AVs in WA. Within the high and low threshold range, we adopt an adjusted cosine function to assign the speed.

To conduct a comprehensive simulation of the model, we make the traffic density and average AV generation interval t_{int} of each lane separately adjustable and follow an exponential distribution with parameter λ. Under the condition of a large traffic volume generated in a lane, assuming that the average length of AVs plus AV spacing is 4.5 m, the average speed of 81 km/h will result in $t_{int} = 0.2$ s. Therefore we change t_{int} from 0.2 s to 10 s through the adjustment of λ in the simulation and

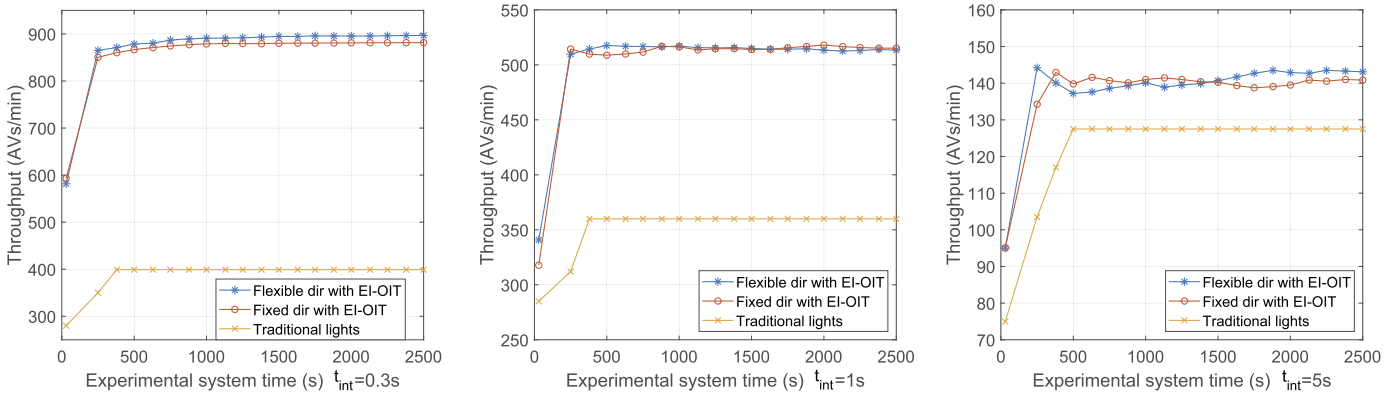

Fig. 6. Throughput over simulation time for different systems. (a) Heavy traffic with $t_{int} = 0.3\ s$. (b) Medium traffic with $t_{int} = 1\ s$. (c) Light traffic with $t_{int} = 5\ s$.

observe the impact of traffic flow density on system throughput. In addition, because the models and algorithms proposed in this paper have improved the flexibility of lane directions, we simulate uneven traffic flow. The traffic flow density of left and right turn is equal, but the proportion between straight-driving and turning is selected as [0.5, 1, 2, 8], that is, AVs generated with straight direction account for 20%, 34%, 50%, and 80%, respectively, and the trends of system performance under different algorithms are observed.

To compare the performance of different algorithms, we choose the intersection controlled by traditional traffic lights [35] and EI-OET with fixed lane directions [36]. For the intersection controlled by traditional traffic lights, we set the red light and green light to last for 30 s respectively, with the yellow light for 5 s between them. For EI-OET with fixed lane directions, it is also a two-way six-lane scenario, and 3 lanes in each direction are set for the left turn, straight, and right turn, respectively but AVs can only drive into the corresponding lane after passing the intersection. The right-turning AVs do not participate in scheduling because they will not collide with other AVs.

5.2. Performance evaluation

The system model adopted in this paper has 3 dimensions of performance evaluation:

- *Throughput*: The system throughput is expressed by the number of scheduled AVs at an intersection in the unit time, which is set to 1 min in this paper. Throughput can directly measure the efficiency of the scheduling system. However, when comparing the efficiency of 2 intersections with different numbers of lanes, we use equivalent throughput instead, that is, the lanes can be extended to a consistent throughput or the proportion of the number of AVs successfully scheduled in a unit time.
- *Delay*: It refers to the average waiting delay of all AVs passing through the intersection in a period of time, expressed as the time interval between reaching the waiting line in the WA and getting the instruction to enter the CA.
- *Computation time*: It refers to the average calculation time required to schedule one AV in a period of time. It is an indicator to measure the computational complexity and the timeliness of the scheduling system.

1) Even densities of traffic flows in all directions

First, we change the traffic density to make the average AV generation time interval of 0.3, 1, and 5 s, respectively. In Fig. 6, as the simulation time goes on, the system throughput rapidly rises and stabilizes near a certain level. Under 3 different levels of traffic density, EI-OET is slightly below or close to OET with fixed directions and is

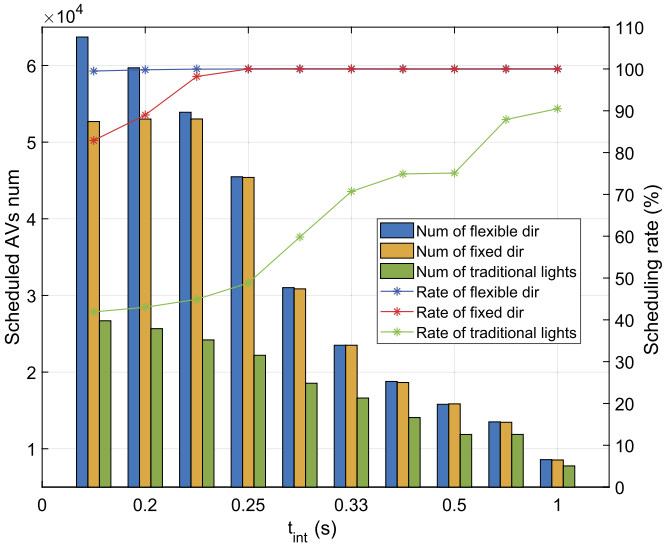

Fig. 7. Scheduling completion rate versus t_{int}.

larger than the traditional traffic light solution. When $t_{int} = 0.3\ s$, the throughput of EI-OET is slightly higher than that of fixed-direction OET, which are 895 AVs/min and 880 AVs/min respectively. When $t_{int} = 1\ s$, the throughput of EI-OET is about 520 AVs/min. When $t_{int} = 5\ s$, the throughput of EI-OET is about 140 AVs/min, while that of the traditional traffic light solution is 300 AVs/min, 360 AVs/min, and 128 AVs/min respectively. In addition, with the traffic density, the advantage of EI-OET is more significant. When $t_{int} = 1\ s$, the throughput of EI-OET is 40% higher than that of the traditional traffic light solution.

To compare the throughput in more detail, we increase t_{int} from 0.2 to 5 s. As shown in Fig. 7, the column represents the number of scheduled AV, and the broken line represents the scheduling completion rate. When $t_{int} < 5\ s$, the scheduling completion rate of fixed-direction OET begins to decline, where it cannot complete all the scheduling tasks of arriving AVs, while EI-OET can always complete all the scheduling tasks. When $t_{int} = 0.2\ s$, the scheduling completion rate of fixed-direction OET is about 82%. For the traditional traffic light solution, the scheduling completion rate drops to 60% when $t_{int} = 1\ s$.

As shown in Fig. 8, we further compare the average delay. When $t_{int} > 0.5\ s$, the average delay of EI-OET and fixed-direction OET is relatively close and both around 1 s, but when $t_{int} < 0.5\ s$, the average delay of fixed-direction OET starts to rise rapidly. When $t_{int} = 0.2\ s$, the average delay of fixed-direction OET is more than 100 s, but the delay of EI-OET is about 0.3 s. The delay is inversely proportional to the pas-

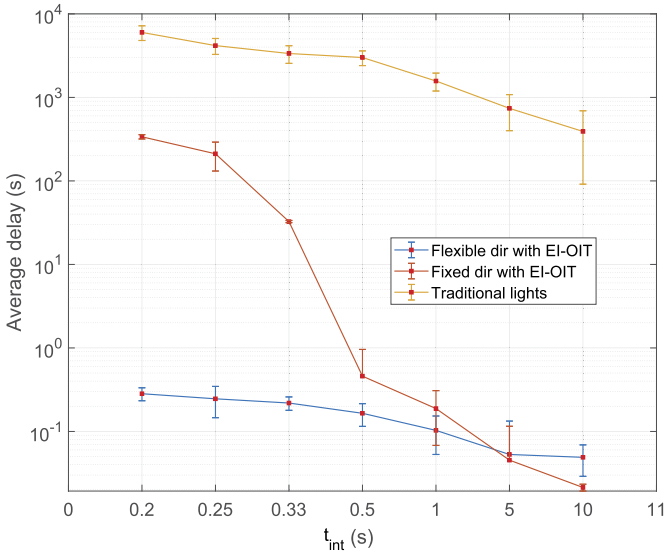

Fig. 8. Average delay versus t_{int}.

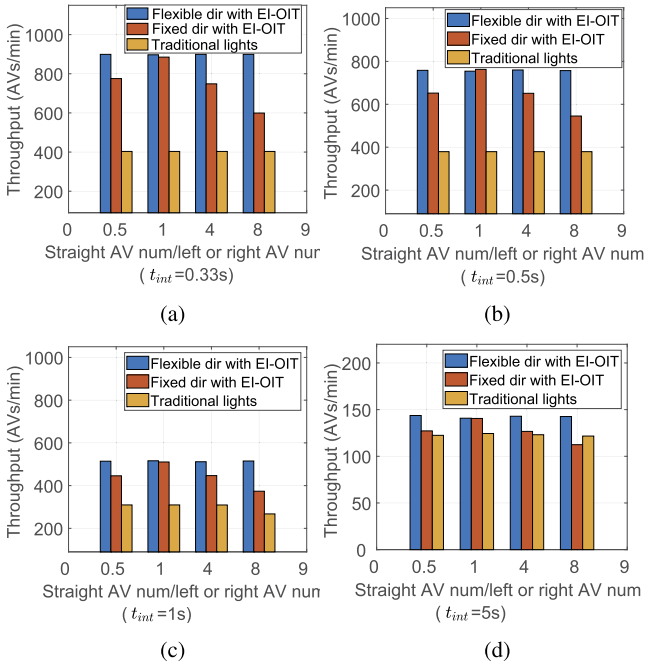

Fig. 9. Throughput versus direction ratio under different t_{int}. (a) $t_{int} = 0.33\ s$. (b) $t_{int} = 0.5\ s$. (c) $t_{int} = 1\ s$. (d) $t_{int} = 5\ s$.

sengers' ride experience. When the traffic is large, EI-OET can greatly reduce the average delay and bring a better ride experience to passengers.

2) Uneven densities of traffic flows in directions

To verify the impact of the proportion of traffic flows in different directions on the system performance, we carry out experiments under different traffic densities and observe the throughput. As shown in Fig. 9, t_{int} is 0.33, 0.5, 1, and 5 s respectively, and the selected independent variable is the number of straight AVs generated in unit time compared with the number of left-turn or right-turn AVs, that is, the proportion of left-turn, straight-driving and right-turn AVs is [0.4, 0.2, 0.4], [0.33, 0.34, 0.33], [0.25, 0.5, 0.25], and [0.1, 0.8, 0.1], respectively.

For EI-OET, the uneven proportion of traffic flows in all directions has no obvious impact on its throughput, but for fixed-direction OET, when the proportion in all directions is equal, it can obtain a close throughput. When the inequality of proportions in all directions increases, the throughput of fixed-direction OET decreases significantly. Compared with the case of equal proportions in all directions, when the proportion in all directions is [0.1, 0.8, 0.1], the throughput decreases by 15.4%, 10.6%, 8.5%, and 0.2% with the change of t_{int}. For the traditional traffic light solution, the traffic density has reached the upper limit of its throughput when $t_{int} < 0.5\ s$, so the proportion of each direction does not change much. However, when $t_{int} > 1\ s$, the uneven traffic flows in different directions also bring about a certain reduction in throughput, which is around 9.7%.

It can be seen that the EI-OET has good stability. When the numbers of AVs to be scheduled in different directions at the intersection are not equal, EI-OET can well maintain the throughput at a high level and does not reduce the throughput due to the restriction of lane directions.

The computational complexity of EI-OET is reflected by the average scheduling time. When $t_{int} = [0.33s, 0.5s, 1s, 5s, 10s]$, the average scheduling time of AVs is in the order of seconds, i.e., 7.29 s, 7.15 s, 5.35 s, 1.42 s, and 1.19 s, respectively. It can be seen that the scheduling time of EI-OET has significant advantages and a relatively low computational complexity.

6. Conclusion

This paper proposed a collision-free scheduling algorithm for unsignalized intersections based on edge clouds to provide computing power and storage support. This EI-OET algorithm has changed the rules of the lane direction and division at traditional intersections. By calculating the occupancy time of AVs at the collision point, the optimal entry time can be allocated to AVs arriving at the intersection, so as to achieve efficient collision-free scheduling at unsignalized intersections. In this paper, EI-OET is used to solve the proposed AVP problem and the scheduling scheme has a low computational complexity. Subsequently, the CVMAC communication protocol and an applicable communication system based on CSMA/CA and VeMAC were proposed as a highly reliable and low-latency communication solution for EI-OET. Finally, the proposed algorithms were verified by simulations to prove their advantages of high throughput, low latency as well as low computational complexity. In the future, we will further study the algorithm in terms of trajectory optimization, optimize EI-OET for more complex traffic scenarios, investigate collaborative scheduling of multiple intersections, and verify the communication implementation and impact of communication conditions on algorithm performance.

CRediT authorship contribution statement

Ziyi Lu: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Tianxiong Wu:** Conceptualization, Data curation, Formal analysis, Methodology, Project administration. **Jinshan Su:** Conceptualization, Funding acquisition, Investigation, Project administration, Resources, Supervision, Validation. **Yunting Xu:** Conceptualization, Funding acquisition, Project administration, Resources. **Bo Qian:** Conceptualization, Funding acquisition, Investigation, Project administration, Resources. **Tianqi Zhang:** Conceptualization, Investigation, Project administration, Resources. **Haibo Zhou:** Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the Natural Science Fund for Distinguished Young Scholars of Jiangsu Province under Grant BK20220067.

References

[1] F. Song, Y. Ma, I. You, H. Zhang, Smart collaborative evolvement for virtual group creation in customized industrial IoT, IEEE Trans. Netw. Sci. Eng. 10 (5) (2022) 2514–2524.
[2] X. Chen, Y. Sun, Y. Ou, X. Zheng, Z. Wang, M. Li, A conflict decision model based on game theory for intelligent vehicles at urban unsignalized intersections, IEEE Access 8 (2020) 189546–189555.
[3] R. Tian, N. Li, I. Kolmanovsky, Y. Yildiz, A.R. Girard, Game-theoretic modeling of traffic in unsignalized intersection network for autonomous vehicle control verification and validation, IEEE Trans. Intell. Transp. Syst. 23 (3) (2022) 2211–2226.
[4] S. Yan, T. Welschehold, D. Büscher, W. Burgard, Courteous behavior of automated vehicles at unsignalized intersections via reinforcement learning, IEEE Robot. Autom. Lett. 7 (1) (2022) 191–198.
[5] M. Bani Younes, A. Boukerche, Intelligent traffic light controlling algorithms using vehicular networks, IEEE Trans. Veh. Technol. 65 (8) (2016) 5887–5899.
[6] C. Suthaputchakun, Z. Sun, A novel traffic light scheduling based on TLVC and vehicles' priority for reducing fuel consumption and CO_2 emission, IEEE Syst. J. 12 (2) (2018) 1230–1238.
[7] J.-G. Wang, L.-B. Zhou, Traffic light recognition with high dynamic range imaging and deep learning, IEEE Trans. Intell. Transp. Syst. 20 (4) (2019) 1341–1352.
[8] C. Chen, Q. Xu, M. Cai, J. Wang, J. Wang, K. Li, Conflict-free cooperation method for connected and automated vehicles at unsignalized intersections: graph-based modeling and optimality analysis, IEEE Trans. Intell. Transp. Syst. 23 (11) (2022) 21897–21914.
[9] C. Chen, B. Wu, L. Xuan, J. Chen, L. Qian, A discrete control method for the unsignalized intersection based on cooperative grouping, IEEE Trans. Veh. Technol. 71 (1) (2022) 123–136.
[10] M. Cai, Q. Xu, C. Chen, J. Wang, K. Li, J. Wang, X. Wu, Multi-lane unsignalized intersection cooperation with flexible lane direction based on multi-vehicle formation control, IEEE Trans. Veh. Technol. 71 (6) (2022) 5787–5798.
[11] Z. Liu, P. Dai, H. Xing, Z. Yu, W. Zhang, A distributed algorithm for task offloading in vehicular networks with hybrid fog/cloud computing, IEEE Trans. Syst. Man Cybern. Syst. 52 (7) (2022) 4388–4401.
[12] Y. Li, S. Xia, M. Zheng, B. Cao, Q. Liu, Lyapunov optimization-based trade-off policy for mobile cloud offloading in heterogeneous wireless networks, IEEE Trans. Cloud Comput. 10 (1) (2022) 491–505.
[13] T.K. Rodrigues, K. Suto, N. Kato, Edge cloud server deployment with transmission power control through machine learning for 6G Internet of Things, IEEE Trans. Emerg. Top. Comput. 9 (4) (2021) 2099–2108.
[14] A.N. Dharsandiya, R.M. Patel, A review on MAC protocols of vehicular ad hoc networks, in: 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2016, pp. 1040–1045.
[15] H.A. Omar, W. Zhuang, L. Li, VeMAC: a TDMA-based MAC protocol for reliable broadcast in VANETs, IEEE Trans. Mob. Comput. 12 (9) (2013) 1724–1736.
[16] F. Song, Z. Ai, Y. Zhou, I. You, K.-K.R. Choo, H. Zhang, Smart collaborative automation for receive buffer control in multipath industrial networks, IEEE Trans. Ind. Inform. 16 (2) (2020) 1385–1394.
[17] L. Qi, M. Zhou, W. Luan, A two-level traffic light control strategy for preventing incident-based urban traffic congestion, IEEE Trans. Intell. Transp. Syst. 19 (1) (2018) 13–24.
[18] Z. Wang, H. Zhu, M. He, Y. Zhou, X. Luo, N. Zhang, GAN and multi-agent DRL based decentralized traffic light signal control, IEEE Trans. Veh. Technol. 71 (2) (2022) 1333–1348.
[19] H. Wu, H. Zhou, J. Zhao, Y. Xu, B. Qian, X. Shen, Deep learning enabled fine-grained path planning for connected vehicular networks, IEEE Trans. Veh. Technol. 71 (10) (2022) 10303–10315.
[20] Y. Bian, S.E. Li, W. Ren, J. Wang, K. Li, H.X. Liu, Cooperation of multiple connected vehicles at unsignalized intersections: distributed observation, optimization, and control, IEEE Trans. Ind. Electron. 67 (12) (2020) 10744–10754.
[21] P. Hang, C. Huang, Z. Hu, C. Lv, Decision making for connected automated vehicles at urban intersections considering social and individual benefits, IEEE Trans. Intell. Transp. Syst. 23 (11) (2022) 22549–22562.
[22] K. Wang, Y. Wang, H. Du, K. Nam, Game-theory-inspired hierarchical distributed control strategy for cooperative intersection considering priority negotiation, IEEE Trans. Veh. Technol. 70 (7) (2021) 6438–6449.
[23] B. Qian, H. Zhou, F. Lyu, J. Li, T. Ma, F. Hou, Toward collision-free and efficient coordination for automated vehicles at unsignalized intersection, IEEE Int. Things J. 6 (6) (2019) 10408–10420.
[24] K. Shu, H. Yu, X. Chen, S. Li, L. Chen, Q. Wang, L. Li, D. Cao, Autonomous driving at intersections: a behavior-oriented critical-turning-point approach for decision making, IEEE/ASME Trans. Mechatron. 27 (1) (2022) 234–244.
[25] P. Hang, C. Huang, Z. Hu, C. Lv, Driving conflict resolution of autonomous vehicles at unsignalized intersections: a differential game approach, IEEE/ASME Trans. Mechatron. 27 (6) (2022) 5136–5146.
[26] F. Song, Z. Ai, H. Zhang, I. You, S. Li, Smart collaborative balancing for dependable network components in cyber-physical systems, IEEE Trans. Ind. Inform. 17 (10) (2021) 6916–6924.
[27] S. Cao, V.C.S. Lee, A novel adaptive TDMA-based MAC protocol for VANETs, IEEE Commun. Lett. 22 (3) (2018) 614–617.
[28] B. Cao, M. Li, L. Zhang, Y. Li, M. Peng, How does CSMA/CA affect the performance and security in wireless blockchain networks, IEEE Trans. Ind. Inform. 16 (6) (2020) 4270–4280.
[29] T. Zhang, Q. Zhu, EVC-TDMA: an enhanced TDMA based cooperative MAC protocol for vehicular networks, J. Commun. Netw. 22 (4) (2020) 316–325.
[30] K. Liu, V.C.S. Lee, J.K.-Y. Ng, J. Chen, S.H. Son, Temporal data dissemination in vehicular cyber-physical systems, IEEE Trans. Intell. Transp. Syst. 15 (6) (2014) 2419–2431.
[31] P. Dai, K. Liu, Q. Zhuge, E.H.-M. Sha, V.C.S. Lee, S.H. Son, Quality-of-experience-oriented autonomous intersection control in vehicular networks, IEEE Trans. Intell. Transp. Syst. 17 (7) (2016) 1956–1967.
[32] H. Zhou, N. Cheng, Q. Yu, X. Sherman Shen, D. Shan, F. Bai, Toward multi-radio vehicular data piping for dynamic DSRC/TVWS spectrum sharing, IEEE J. Sel. Areas Commun. 34 (10) (2016) 2575–2588.
[33] O.L. Mangasarian, Absolute value programming, Comput. Optim. Appl. 36 (1) (2007) 43–53.
[34] P.A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, E. Wiessner, Microscopic traffic simulation using SUMO, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018, pp. 2575–2582.
[35] O. Younis, N. Moayeri, Employing cyber-physical systems: dynamic traffic light control at road intersections, IEEE Int. Things J. 4 (6) (2017) 2286–2296.
[36] Y. Xu, H. Zhou, T. Ma, J. Zhao, B. Qian, X. Shen, Leveraging multiagent learning for automated vehicles scheduling at nonsignalized intersections, IEEE Int. Things J. 8 (14) (2021) 11427–11439.