Leveraging Multiagent Learning for Automated Vehicles Scheduling at Nonsignalized Intersections

Yunting Xu, *Student Member, IEEE*, Haibo Zhou, *Senior Member, IEEE*, Ting Ma, *Member, IEEE*, Jiwei Zhao, *Student Member, IEEE*, Bo Qian, *Student Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Recent advancements of Vehicle-to-Everything (V2X) communication combined with artificial intelligence (AI) technologies have shown enormous potentials for improving traffic management efficiency and intelligence. To provide innovative and effective data-driven traffic management solution for the coming automated vehicle era, we present a vehicle–road collaboration-enabled nonsignalized intersection management architecture in this paper. First, by dividing the intersection zone into the central section (CS) and the waiting section (WS), a vehicle regulation scheme involved with communication and computation planes is developed for V2X-enabled nonsignalized intersection management. Specifically, in order to guarantee vehicle safety, the definition of no overlapping occupation time in CS and the fastest crossing time point (FCTP) algorithm are employed for vehicle collision avoidance. Second, considering the relative coordination between adjacent intersections, a multiagent-based deep reinforcement learning scheduling (MA-DRLS) algorithm is proposed to realize cooperative multiple intersection management. Through information exchange with different intersection agents, each agent can obtain an optimal scheduling strategy using independent deep reinforcement learning (DRL) network. The features of fixed Q-targets and experience replay are leveraged to improve the reliability of neural network during the training process. Finally, simulation performances in terms of intersection throughput and vehicle waiting time have been provided to validate the effectiveness and demonstrate the superiority of the proposed nonsignalized intersection management solution.**

***Index Terms*—Multiagent, nonsignalized intersection management, reinforcement learning (RL), Vehicle-to-Everything (V2X) communication.**

I. Introduction

With the significant enhancement of wireless communication and intelligent vehicular technologies for autonomous driving, vehicle–road collaboration has become a promising way to provide innovative and effective traffic management solutions within transportation system [1]–[5]. Intersection management is one of the most challenging traffic scenes that has received extensive attention in dealing with severe urban traffic congestion. Current intersection management is mainly taken up by traditional traffic light, however, the scheduling mechanism of regular signal phrase control could hardly alleviate traffic jam and often leads to the drastic congestion-related problem. According to the Urban Mobility Scorecard statement, congestion in intersection areas results in an additional 3.1 billion gallons of fuel consumption and nearly 6.8 billion more hours spent on traveling each year [6]. For further enhancing the performance of traffic light, with sufficient sensor and camera equipment, adaptive signal control is introduced to elevate urban driving experience [7], [8]. Such intelligent traffic lights start intersection regulation change along with the real-time detected traffic status information. But when the traffic becomes more and more heavy on the road, adaptive signal control cannot completely eliminate unnecessary vehicle stops at the intersection, which elucidates that this approach is difficult to meet the requirements of managing the traffic flows for maximum efficiency.

As vehicular *ad hoc* networks (VANETs) and vehicle-to-everything (V2X) technologies remarkably evolve in recent years, automated vehicles equipped with onboard units (OBUs) can enrich their communication connections with each other and proactively exploit different strategies to navigate intersection [9]–[11]. To make full potential use of intersection management for automated vehicles on the road, removing traffic lights has been immensely taken into consideration. Without the rules and restrictions of the traffic light, the intersection management center (IMC) plays the role of mission control to coordinate with each vehicle for the aims of maximizing traffic throughput and safety guarantee [6], [12]. The feature of the nonsignalized scenario enables the improvement of intersection management efficiency by leveraging V2X communication and autonomous driving technologies. However, for most scheduling methods proposed in the existing research, there are plenty of strict limitations on the vehicle motion process which extremely rely on the calculating ability of computational units. Therefore,

Manuscript received August 18, 2020; revised November 26, 2020; accepted January 9, 2021. Date of publication January 26, 2021; date of current version July 7, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61871211; in part by the Innovation and Entrepreneurship of Jiangsu Province High-Level Talent Program; in part by the Summit of the Six Top Talents Program of Jiangsu Province; and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC). *(Corresponding author: Haibo Zhou.)*

Yunting Xu, Haibo Zhou, Ting Ma, Jiwei Zhao, and Bo Qian are with the School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China (e-mail: yuntingxu@smail.nju.edu.cn; haibozhou@nju.edu.cn; majiawan27@163.com; jwzhao@smail.nju.edu.cn; boqian@smail.nju.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/JIOT.2021.3054649

when facing with sophisticated and dynamic nonsignalized intersection environment, it is of significant importance to adopt an innovative approach for dealing with dimensional traffic data and generating optimal management decision for intersection crossing strategies [13].

Recent advancements of artificial intelligence (AI) involved with the data processing framework is widely used for optimizing transportation management [14]–[16]. The state-of-the-art AI technologies that have been applied in complicated traffic environment mainly include machine learning, fuzzy logic, neural network (NN), dynamic programming, etc. Of all these existing approaches, reinforcement learning (RL) is a typical method which is suitable for nonsignalized intersection management [17]. When it comes to the collaborative regulation of the multi-intersection system, RL-based scheduling methodologies can be broadly categorized into two main classifications, that is, centralized and distributed management mechanism [18]. For the centralized mechanism, the whole multiple intersection system is looked upon as an entirety to find global scheduling decision for joint intersections control. However, as the number of intersection increases, the amount of vehicle state will grow exponentially, which means that the centralized mechanism is difficult to get away from the curse of state dimensionality. With regard to the mechanism employing distributed methods, each intersection is modeled as an intelligent agent that can observe its own traffic state and take scheduling strategy independently. To realize cooperative multi-intersection control, different RL algorithms can be applied into the multiagent learning system.

With the above-mentioned considerations, this work is motivated by the superiority of the nonsignalized scenario and by the intensive research effort in multiple intersection system. We first construct a general collision-free nonsignalized architecture for intersection management. IMC is set up to enable V2X communication and to compute optimal crossing time for vehicle passing through intersection. After that, considering the relative coordination between adjacent intersections, each intersection is modeled as an intelligent agent so as to formulate a multiagent system. In multiple intersection scenario, agents can exchange their messages so that global traffic information of the whole system can be obtained. Each intersection maintains an independent decision-making scheme to seek out scheduling solution through the deep RL (DRL) network. During the training process, the features of fixed Q-targets and experience replay are leveraged to improve the reliability of NN. Fixed Q-targets refer to two NNs with the same structure but different parameters, which can cut off the correlation between the target value and the evaluated value. Experience replay means creating a memory bank to store certain state–action pairs for batch size training. In the light of the previous literature, we highlight three main contributions of this paper in the following.

1) We present a vehicle–road collaboration-enabled nonsignalized intersection architecture to provide safety and efficient crossing solution for the automated vehicle. In the context of existing nonsignalized methods, this architecture is designed to have the features of a simplified vehicle control mechanism, low computation cost, and less communication overhead.
2) We propose the fastest crossing time point (FCTP) algorithm to provide fine-grained vehicle crossing decision and multiagent-based DRL scheduling (MA-DRLS) algorithm to realize cooperative multiple nonsignalized intersection management, respectively, which will open up new horizons for innovative data-driven nonsignalized intersections management solution.
3) Compared with the traffic light we currently deploy for intersection management, simulation results in terms of intersection throughput and vehicle waiting time have demonstrated the tremendous superiority of nonsignalized intersection management, which can help to promote the engineering applications of the intelligent transport system.

The remainder of this paper is organized as follows. In Section II, we introduce the related work. Section III presents the system model, including nonsignalized intersection architecture and the introduction of RL with the multiagent system. Section IV gives the problem formulation and scheduling solution of FCTP and MA-DRLS algorithms for automated vehicles. Section V shows the performance evaluation results of the proposed algorithm. Finally, the conclusion of this paper is presented in Section VI.

II. Related Work

A. Development of Intersection Management

The developmental feature of the intersection management is a progress from stationary regulation to progressively real-time scheduling. The simplest traffic light operates on fixed signal time, which is inefficient for traffic regulation. To enhance the capability of the traffic light, adaptive signal control has been extensively studied in the past decade. Younis and Moayeri [19] used a sensor network to detect traffic data and proposed a dynamic signal framework for the road system. Wu *et al.* [20] presented a delay-based signal control scheme to optimize intersection throughput. With the recent development of AI technologies, data-driven solutions have been applied into intersection management [21]. Wan and Hwang [22] leveraged a value-based DRL method for adaptive signal control. Simulation results showed that it can reduce total system delay by 20%. Li *et al.* [23] used a long short-term memory network to provide long-term planning. Although these approaches can improve regulation performance to a certain extent, they cannot completely eliminate the essential deficiency of traffic light. In [24], it had been investigated that the optimization for adaptive signal control often leads to the results that each signal phrase possesses the shortest possible time period. Therefore, a more fine-grained scheduling strategy should be developed to fully utilize the potential of intersection management.

Advanced V2X communication technologies have enabled convenient vehicle–road collaboration under a complicated traffic environment. As automated vehicles become more intelligent to adapt to the changing traffic, nonsignalized intersection management has received considerable research

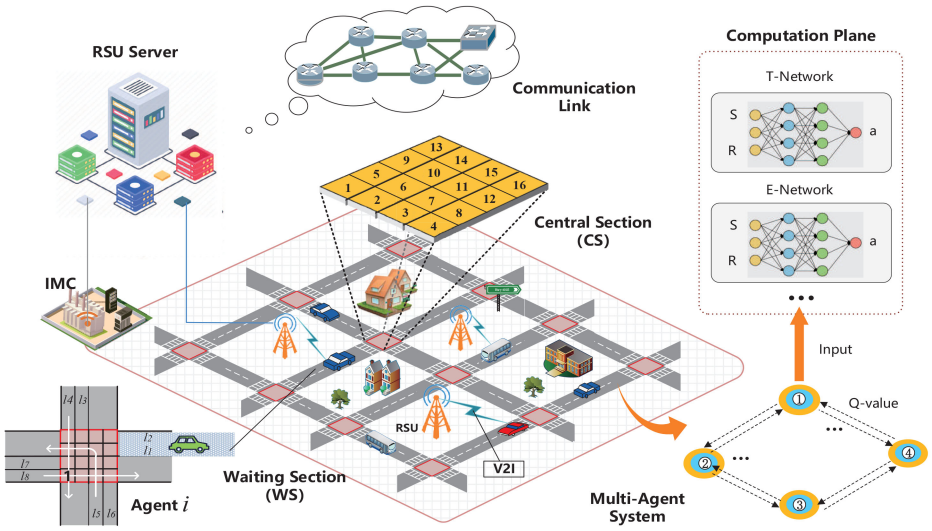

Fig. 1. Vehicle–road collaboration-enabled nonsignalized intersection architecture.

attention for improving scheduling efficiency. Kamal *et al.* [25] defined the concept of collision point (CP) and prevented vehicles from arriving CP at the same time to guarantee vehicle safety. Zhang *et al.* [26] created a state-driven scheduling mechanism for automated vehicles. A new priority-based scheduling algorithm was proposed to decide the sequence order for crossing intersection. Both [25] and [26] considered the safety problems of nonsignalized intersection, while some other researchers focused on particular vehicle features during the navigating process. Dai *et al.* [27] designed a novel smoothness metric to quantitatively capture the travel experience quality. Sayin *et al.* [28] developed an information-driven control scheme and charged vehicle with their impact on other drivers. To simplify the scheduling mechanism, Qian *et al.* [29] proposed an optimal entering time algorithm to pass through intersection and Wang *et al.* [30] exploited detecting zone and control zone for vehicle motion control. These technical solutions to nonsignalized intersection management have shown better intersection performance compared with traffic light regulation.

B. Multiple Intersection System

In respect of multi-intersection scheduling, how to realize the collaborative regulation among intersections has posed great challenges for vehicle crossing strategy. Some researchers investigated multi-intersection management under specifically defined rules. Younes and Boukerche [31] introduced an arterial traffic light mechanism based on real-time information. Greater priority and preference were allocated to the vehicles on arterial lanes. Hou *et al.* [32] considered upstream and downstream flows to guarantee fairness in a multi-intersection system. Other studies looked into the environmentally friendly issue during scheduling. Yang *et al.* [33] developed an eco-driving framework utilizing timing data, which was proved to reduce 13.8% fuel consumption. Lin *et al.* [34] analyzed the arbitrary section of two red-signalized intersections and used Legendre pseudospectral for fuel-optimal operation.

As for the machine learning technologies leveraged for multiple intersection management, centralized and distributed mechanisms are commonly used in dealing with the dynamic environment. In a centralized mechanism, Lee *et al.* [35] employed the RL algorithm with convolutional NN (CNN) to recognize the entire traffic state and jointly control traffic signals of all intersections. The structure of NN was redesigned so that global actions can be outputted by limited parameters. In a distributed mechanism, Liu *et al.* [36] presented a clustering-based RL scheme in the V2X network. Each intersection used the function approximation approach to fit the actual state–action value. Ge *et al.* [18] proposed a cooperative deep Q-network with Q-value transfer (QT-CDQN) algorithm for multiagent system. The Q-values from neighboring agents were transferred to the loss function of the CNN network. However, for the existing scheduling works, most of them only focused on the research of adaptive signal control, without considering the superiority of the nonsignalized scenario.

To take the advantage of aforementioned methodologies, in this paper, we adopt nonsignalized intersection management to provide innovative traffic regulation solution for automated vehicles. Considering multidimensional traffic state, the distributed mechanism is leveraged for realizing the coordination between neighboring intersections. From the development perspective of traffic management, this work brings the nonsignalized scenario into multi-intersection management, which opens up pioneering horizons for the intelligent transportation system.

III. System Model

In this section, we first present a general vehicle–road collaboration-enabled nonsignalized management architecture for the multi-intersection scenario. Then, the RL method together with multiagent system is introduced to perform cooperative multiple intersection management.

A. Nonsignalized Intersection Management Architecture

In the architecture of nonsignalized intersection, the junction of the intersection area is subdivided into two new parts as

TABLE I
SUMMARY OF IMPORTANT SYMBOLS

$\boldsymbol{L}_{\boldsymbol{m}}^{i}$	Lane order $(l_1^i, \cdots, l_8^i)$ of intersection i.
$\zeta_{m,n}^{i}$	The occupation time period for vehicle in m-th lane and grid n.
V_{cs}^{i}	The constant velocity of central section at i-th intersection.
$\boldsymbol{X^i}$	The crossing time point $(x_1^i, \cdots, x_8^i)'$ for each lane's vehicle.
V_{ws}^{i}	The instantaneous velocity of vehicle when it reaches the WS of intersection i.
T_{ws}^{i}	The instantaneous time point of vehicle when it reaches the WS of intersection i.
K_m^i	The number of vehicles that have been allocated crossing time in lane $\boldsymbol{L^i}$.
T_s	Vehicle safety time calculated by vehicle safety distance.
$Q_\pi(s,a)$	The value of specific state-action pair under policy π.
γ	Discounted factor to decide the importance of future reward.
$P_{ss'}^{a}$	Transition probability from state s to s' after taking acion a.
$\pi(a\|s)$	The probability of taking action a in state s under policy π.

illustrated in Fig. 1. The middle square part of the intersection is defined as the central section (CS) which is partitioned into small identical grids for clearly presenting the motion process of vehicle. The extension part of the joint junction is the waiting section (WS). IMC is established for communication connection with automated vehicles. In order to simplify the scheduling mechanism and reduce the computation cost, vehicles are only required to send their instantaneous velocity and arriving time to IMC once they reach WS. Subsequently, the computation plane of IMC will immediately return the optimal crossing decisions for these vehicles so that they can adjust their motion statuses in WS. Such interaction mechanism does not require all vehicles to establish real-time and uninterrupted connections, so the communication overhead of intersection management can be greatly reduced.

Under the above architecture, we consider a four-way intersection scenario which consists of a straight and left-turning lane. The right-turning lane is an independent lane and vehicles in this lane will not collide with the vehicles in other lanes. Therefore, there are total eight lanes in the presented four-way intersection scenario. Based on these eight lanes, we divide the CS into 16 identical grids so as to depict the vehicle motion trajectory and calculate the navigating time of the vehicle in the intersection. Indicate $\boldsymbol{L_m^i} = (l_1^i, \ldots, l_8^i)'$ as the lane order of intersection i and $\zeta_{m,n}^i$ as the occupation time period for vehicle in the mth lane and going to pass through grid n. Combining the particular grids involved with vehicle motion path, the occupation time period of both straight and left-turning trajectories can be calculated. V_{cs}^i represents the vehicle navigating velocity in CS which is set as a constant value for driving stability and better travel experience. If the width of each grid is w, the length of straight trajectory is $4w$ and the occupation time period can be calculated as $T_{\text{straight}} = 4w/V_{cs}^i$. Likewise, the length and time period of left-turning trajectory are $(16+\pi)w/4$ and $T_{\text{turning}} = (16+\pi)w/4V_{cs}^i$, respectively. $\boldsymbol{X^i} = (x_1^i, \ldots, x_8^i)'$ means the crossing time point to CS for each vehicle from l_1^i to l_8^i. A summary of important symbols used in this paper is listed in Table I.

The safety problem of nonsignalized intersection can be regarded as no overlapping occupation time to the same grid for all vehicle trajectories. Take grid 1, for example, $\zeta_{4,1}^i =< x_4^i + 3w/V_{cs}^i, x_4^i + 4w/V_{cs}^i >$ and $\zeta_{8,1}^i =< x_8^i, x_8^i + w/V_{cs}^i >$ denote the occupation time of l_4^i and l_8^i correspondingly. The problem of $\zeta_{8,1}^i \cap \zeta_{4,1}^i = \phi$ can be converted into $|(x_4^i + 7w/2V_{cs}^i) - (x_8^i + w/2V_{cs}^i)| = |x_4^i - x_8^i + 3w/V_{cs}^i| \geq w/V_{cs}^i$ (according to the absolute values of difference between their middle points should not less than half of the total occupation time [29]). In the same way, considering the collision relationship with regard to all the 16 grids, we can derive 20 different inequalities and formulate them into the following matrix:

$$\left|\boldsymbol{\alpha X^i} + \boldsymbol{\beta}\right| \geq \boldsymbol{c} \quad \left(\boldsymbol{\alpha} \in \mathbb{R}^{20\times 8}, \boldsymbol{X^i} \in \mathbb{R}^8, \boldsymbol{\beta}, \boldsymbol{c} \in \mathbb{R}^{20}\right) \quad (1)$$

where $\boldsymbol{\alpha}$ is constituted by the coefficients of $x_1^i, \ldots, x_8^i$ in these inequalities and $\boldsymbol{\beta}$ and $\boldsymbol{c}$ are two 20-D vectors with distinct constant values.

Next, for realizing the objective of maximizing intersection throughput, IMC needs to assign each vehicle the FCTP to enter CS. Once a vehicle reaches the WS, its instantaneous velocity and this time moment is recorded as V_{ws}^i and T_{ws}^i. We specify the maximum velocity $V_{\max}$ of WS and vehicles can adjust their status with the maximal acceleration $a_{\max}^i$ and minimal acceleration $a_{\min}^i$. The least time a vehicle spends in WS can be calculated following the process that the vehicle first accelerates to reach $V_{\max}$ with maximal acceleration $a_{\max}^i$, and then decelerates to reach V_{cs}^i with minimal acceleration $a_{\min}^i$ after navigating $L_{\max}$ distance under $V_{\max}$. It is realistic that the crossing time point to CS x_m^i is equal or longer than T_{ws}^i plus the least WS navigating time, which can be described as

$$\begin{aligned} x_m^i \geq\ & T_{ws}^i + \left(V_{\max} - V_{ws}^i\right)/a_{\max}^i \\ & + L_{\max}/V_{\max} + \left(V_{cs}^i - V_{\max}\right)/a_{\min}^i. \end{aligned} \quad (2)$$

Another factor that will affect the optimal crossing time is the collision relationship between the existing vehicles in CS and the vehicle that is about to pass through the intersection. The number of vehicles that already in CS from lane L^i is parameterized as $K_m^i (m = 1, \ldots, 8)$ and each vehicle's crossing time is recorded as $x_m^{(k)|i} (k \in K_m^i)$. For l_1^i, the collision-free constraint can be expressed in the following:

$$\begin{aligned} |x_1^{(k)|i} - x_4^i + 2w/V_{cs}^i + w\pi/4V_{cs}^i| &\geq w/V_{cs}^i \\ |x_1^{(k)|i} - x_3^i + w/V_{cs}^i + w\pi/4V_{cs}^i| &\geq w/V_{cs}^i \\ |x_1^{(k)|i} - x_5^i + w/2V_{cs}^i + w\pi/8V_{cs}^i| &\geq w/2V_{cs}^i + w\pi/8V_{cs}^i \\ |x_8^i - x_1^{(k)|i} + 2w/V_{cs}^i| &\geq w/V_{cs}^i \\ |x_3^i - x_1^{(k)|i} + w/V_{cs}^i + w\pi/4V_{cs}^i| &\geq w/V_{cs}^i \\ |x_7^i - x_1^{(k)|i} + w/2V_{cs}^i + w\pi/8V_{cs}^i| &\geq w/2V_{cs}^i + w\pi/8V_{cs}^i. \end{aligned}$$

Consequently, for all the lanes L^i in intersection i, we can transform the same constraints into matrix formulation

$$|\boldsymbol{\alpha_m X^i} + \boldsymbol{\beta_m^k}| \geq \boldsymbol{c_m}\ \forall k = 1, \ldots, K_m^i,\ \ m \in \boldsymbol{L^i}. \quad (3)$$

Finally, it is reasonable to consider the safety distance s between two vehicles. In this paper, we use the concept of safety time for simplifying elaboration. The safety time can be represented by $T_s = (s + l_v)/v$, where l_v is the length of vehicle and v is the navigating velocity. Let

$$\boldsymbol{X^{(K)}} = \left[\mathbf{max}\left(x_1^{(k)|i}\right), \ldots, \mathbf{max}\left(x_8^{(k)|i}\right)\right]' \quad (4)$$

the basic safety constraint for adjacent vehicles is as follows:

$$\boldsymbol{X^i} \geq \boldsymbol{X^{(K)}} + (s + l_v)/v. \quad (5)$$

B. Reinforcement Learning With Multiagent System

The RL method is leveraged to deal with nonsignalized intersection management due to the fact that it can make an optimal policy through continuous interaction with traffic environment [37]. IMC plays the role of intersection agent for observing traffic state. In the process of interaction, an agent takes selectable action a_t under state s_t and receives impacts r_t at current step t from the feedback of environment, that is, the general implication of *Reward*. After the specific action is executed, the expectation of the subsequent cumulative rewards is parameterized as the state–action value function

$$Q_\pi(s,a) = E_\pi\Big[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots | a = a_t, s = s_t\Big]. \tag{6}$$

In (6), π is the specific policy agent takes. γ $(0 \le \gamma < 1)$ is defined as the discounted factor and it can avoid the unboundedness of the state–action value caused by directed reward addition. If γ is close to 1, it represents that this agent is looking for long-term reward in the process of learning. The state–action function is also known as the Q-value function which can be transformed into the following expression:

$$Q(s,a) = r(s,a) + \gamma \sum_{s' \in S} P^a_{ss'} \sum_{a' \in A} \pi\big(a'|s'\big) Q\big(s', a'\big). \tag{7}$$

Formula (7) means that $Q(s,a)$ (the value for taking action a in state s) is equal to current reward $(r(s,a))$ plus future reward. Besides, $s' \in S$ (state set) and $a' \in A$ (action set) are the state and action of the next step. The Q-value can reflect the quality of action taken in the current state and needs to be updated until achieving convergence. The update of Q-value at step t is a stochastic incremental transition which can be formulated as

$$\begin{aligned} Q(s_t,a_t) &\leftarrow Q(s_t,a_t) \\ &+ \alpha\Big[r(s_t,a_t) + \gamma \max_{a_{t+1} \in A} Q_t(s_{t+1},a_{t+1}) - Q(s_t,a_t)\Big] \end{aligned} \tag{8}$$

in which $0 < \alpha \le 1$ is called a learning rate, and it determines the update degree from the existing experience to newly emerging knowledge. Among all the RL methods, Q-learning is a typical approach for obtaining an optimal strategy. It maintains a Q-table and incrementally update Q-value to find the best action. This tabular updating-based learning method enables Q-values within limited scale state and action set converging to a stable status, but it is difficult in dealing with complex environment. To solve the dimensional disaster involved with intensive scale states and actions, Q-learning together with NN is adopted in this paper. In such DRL framework, the traffic state is taken as the input of NN and NN will directly generate Q-values for each action rather than recording them in the table. Consequently, the incremental Q-value updating in (8) becomes the training and substitution of NN

$$\begin{aligned} \boldsymbol{New-NN} &\leftarrow \boldsymbol{Old-NN} \\ &+ \alpha\Big[r(s_t,a_t) + \gamma \max_{a_{t+1} \in A} Q_t(s_{t+1},a_{t+1}) - Q(s_t,a_t)\Big]. \end{aligned} \tag{9}$$

When it comes to the application of RL in multiple intersection management, we regard the intersections as intelligent agents to formulate a multiagent system. Each agent exploits an independent DRL framework to realize vehicle scheduling based on both the observation of the traffic environment and the information sent from adjacent agents. For the interaction between the agent and traffic environment as depicted in Fig. 2, the traffic environment will return particular effectors after the agent executing the current action of the scheduling policy. As for the cooperative multiple intersection management, the real-time traffic flow conditions at surrounding intersections have a large impact on the traffic efficiency of the whole road system. Therefore, the collaboration between the neighboring intersection agent is extremely important for optimizing management performance. Based on the output generated by independent NN and the timely information sharing of the traffic condition between adjacent intersections, the coordination of agents in the entire system can be conducted. Finally, during the process of continuous interaction with the environment and other agents, we can obtain an optimal strategy for cooperative multi-intersection management.

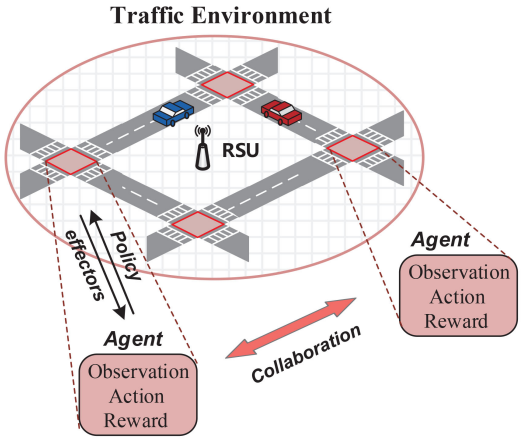

Fig. 2. Multiagent learning scenario for multiple intersection management.

IV. Problem Formulation and Nonsignalized Intersection Management Solution

In this section, to enhance the performance of traffic regulation, we present an FCTP mechanism for each intersection in the whole system. Subsequently, a multiagent-based DRL solution is proposed for cooperative intersection management.

A. Fastest Crossing Time Point Mechanism

The primary objective of nonsignalized intersection is to realize the maximal efficiency of vehicle scheduling, which means that IMC should allocate the fastest crossing time to CS for each vehicle. In the situation of one vehicle about to pass through intersection i at lane l^i_m, the FCTP can be expressed as min x^i_m, and x^i_m should subject to the collision-free constraints mentioned in (1)–(5). For different intersections, they can specify the navigating velocity of CS independently and then calculate the crossing decision for all vehicles reaching WS. For intersection i, the optimal nonsignalized scheduling can be formulated as

$$\begin{aligned} &\min \ x^i_m, \quad m \in \boldsymbol{L^i} \\ &\text{s.t.} \ \ \big|x^i_m + \bar{\beta}_{m^k}\big| \ge \bar{c}_m \quad \forall k = 1, \ldots, K_{\bar{m}^i}, \bar{m} \ne m \end{aligned}$$

Algorithm 1: FCTP to Nonsignalized Intersection

Input: WS velocity V^i_{ws}; WS entry time T^i_{ws}; CS velocity V^i_{cs}
Output: The crossing time point to CS for vehicle x^i_m

If there is one vehicle in intersection i, go to step 3;
If there are multiple vehicles in intersection i, choose one vehicle (in the order from l^i_1 to l^i_8);
Initialize $t_o = \hat{x}_g$;
Denote the dimension of $|x^i_m + \bar{\beta}_{m^k}| \geq \bar{c}_m$ as $\bar{D}_m$; Initialize n=1;
if $1 \leq n \leq \bar{D}_m$ **then**
  if $-\bar{c}_m(n) - \bar{\beta}_{m^k}(n) < t_o$ **then**
    $t_o = \max\{\bar{c}_m(\mathrm{n}) - \bar{\beta}_{m^k}(\mathrm{n}) , t_o\}$
  end
  if $n > 1$ **then**
    for $a = n : 2$ **do**
      if $-\bar{c}_m(a-1) - \bar{\beta}_{m^k}(a-1) < t_o$ && $t_o \leq \bar{c}_m(a-1) - \bar{\beta}_{m^k}(a-1)$ **then**
        $t_o = \bar{c}_m(a-1) - \bar{\beta}_{m^k}(a-1)$
      end
    end
  end
  n=n+1;
end
Return the optimal fastest crossing time point $x^i_m = t_o$;

$$\begin{aligned} x^i_m \geq\ & \mathbf{max}\left(x^{(k)|i}_m\right) + (s + l_v)/v \\ x^i_m \geq\ & T^i_{ws} + \left(V_{\max} - V^i_{ws}\right)/a^i_{\max} \\ & + L_{\max}/V_{\max} + \left(V^i_{cs} - V_{\max}\right)/a^i_{\min} \end{aligned} \tag{10}$$

in which $\bar{\beta}_{m^k}$ and $\bar{c}_m$ can be obtained from the inequalities of no overlapping occupation time to the same grid. To solve the problem (10), we indicate that

$$\begin{aligned} \hat{x}_g = \mathbf{max}\Big\{ & T^i_{ws} + \left(V_{\max} - V^i_{ws}\right)/a^i_{\max} + L_{\max}/V_{\max} \\ & + \left(V^i_{cs} - V_{\max}\right)/a^i_{\min},\ \mathbf{max}\left(x^{(k)|i}_m\right) + (s + l_v)/v \Big\}. \end{aligned} \tag{11}$$

Consequently, the problem (10) can be simplified as

$$\begin{aligned} \min\ & x^i_m, \quad m \in \boldsymbol{L}^i \\ \text{s.t.}\ & \left|x^i_m + \bar{\beta}_{m^k}\right| \geq \bar{c}_m \ \ \&\& \ \ x^i_m \geq \hat{x}_g. \end{aligned} \tag{12}$$

After that, we propose the FCTP algorithm to cope with the vehicle crossing strategy problem. For intersection i, if there is only one vehicle going to pass through the intersection, the FCTP algorithm can calculate the final result to problem (12) immediately. In the case of more than one vehicle at the same time interval, an intersection agent will choose the vehicle from l^i_1 to l^i_8 and calculate the crossing time in order. The details of the FCTP algorithm are shown in Algorithm 1.

Denote t_o as the calculated optimal FCTP for specific vehicle and t_o is initialized as $\hat{x}_g$ due to the second constraint $x^i_m \geq \hat{x}_g$. Let the nth constituent of $\bar{\beta}_{m^k}$ and $\bar{c}_m$ be $\bar{\beta}_{m^k}(n)$ and $\bar{c}_m(n)$, respectively. Therefore, solving the first absolute value constraint $|x^i_m + \bar{\beta}_{m^k}| \geq \bar{c}_m$ means solving the following inequality:

$$\begin{aligned} & x^i_m \geq \bar{c}_m(n) - \bar{\beta}_{m^k}(n) \qquad \text{or} \\ & x^i_m \leq -\bar{c}_m(n) - \bar{\beta}_{m^k}(n) \quad \forall k = 1, \ldots, \bar{D}_m \end{aligned} \tag{13}$$

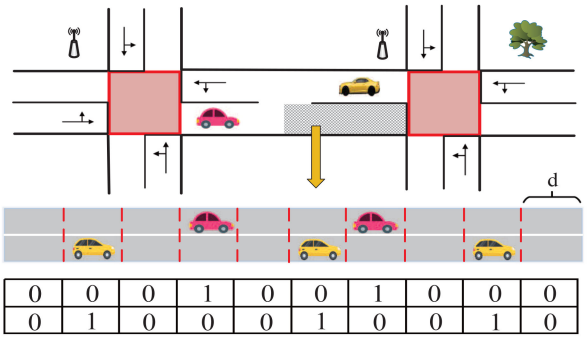

Fig. 3. Intersection traffic state based on segment partition.

where $\bar{D}_m$ is the dimension of $|x^i_m + \bar{\beta}_{m^k}| \geq \bar{c}_m$ constituent. Finally, the result of t_o is the optimal FCTP x^i_m for the coming vehicle in intersection i at lane l^i_m.

B. Multiagent-Based Deep Reinforcement Learning Scheduling for Multi-Intersection

The FCTP algorithm provides a fundamental nonsignalized management solution for an automated vehicle to cross the intersection. When it comes to the situation that multiple vehicles going to cross intersection simultaneously, there are still room for enhancing the performance of intersection management. To this end, we propose the DRL-based management solution to seek out effective scheduling decision for each intersection under a comparatively congested traffic environment. As for multi-intersection management, each intersection agent uses the same DRL structure and also jointly takes the information from adjacent intersections into consideration. The traffic state, crossing action, and reward function are defined as follows.

1) *State Set:* For intersection management, it is essential to observe pivotal vehicle information so as to realize optimal traffic regulation, especially in the environment without traffic lights. There are various vehicle information that can be used in making crossing decision, such as vehicle velocity, current position, driving direction, etc. In order to determine whether the road section is in the congested situation, we perceive the number of vehicles based on the partition of the rectangular segment. As shown in Fig. 3, the extension area of the intersection is divided into several identical rectangular segments. The length of each segment is indicated as d and d is slightly longer than the average vehicle length for clearly identifying the state of the vehicle on the road. If there is a vehicle in the segment, the value of this specific segment is set to 1, otherwise, the value is set as 0. For the ith intersection, the traffic state is expressed as S^i_m, where m is the lane index. Taking Fig. 3 as an example, the gray area contains two lanes and ten divided segments, so the traffic state of this gray area is mathematically expressed as

$$\mathrm{STATE} = \begin{pmatrix} 0\,0\,0\,1\,0\,0\,1\,0\,0\,0 \\ 0\,1\,0\,0\,0\,1\,0\,0\,1\,0 \end{pmatrix}.$$

Correspondingly, we have the following formulation for the intersection state with all eight lanes

$$\text{STATE} = \left(S_1^i, S_2^i, S_3^i, \ldots, S_8^i\right)^T. \tag{14}$$

Assumed that there are n rectangular segment in the extension area, theoretically, the complexity of the state level is $(2^8)^n$ for an isolated intersection with eight lanes. It is obvious that the distribution of the vehicle would lead to a large scale of the traffic state. If a classic tubular Q-learning method is used to learn the optimal policy from the environment, the session of training will cost a lot of computational overhead. Meanwhile, Q-learning is also difficult to make Q-value converge to a stable condition. To address this problem, we leverage the DRL structure to approximate Q-values through NN for each intersection.

2) *Action Set:* Adaptive traffic control usually reallocates the priority order and time duration for different signal phrases. For nonsignalized intersection scenario, vehicles coming from different lanes can directly pass through the intersection so that there is no need to take phrase duration into consideration. In this paper, we prioritize the vehicles in eight lanes at the current dispatch interval and take the scheduling order as executed action. For each traffic state, we manage to seek out optimal action with the aim of enhancing intersection vehicle throughput. Let A_n^i $(0 \le n \le N_t^i!)$ be the nth lane priority order of intersection i and N_t^i is the total number of vehicles at scheduling interval t, which is calculated as

$$N_t^i = n_{l_1^i,t} + n_{l_2^i,t} + \cdots n_{l_8^i,t}, 0 \le N_t^i \le 8, n_{l_m^i,t} \in \{0, 1\}. \tag{15}$$

The value of $n_{l_m^i,t}$ represents whether there is a vehicle needed to be scheduled at current scheduling interval t in lane l_m^i. Therefore, we have $N_t^i!$ actions to choose in one dispatch. If N_t^i is 8, one of the priority action can be expressed as $A_n^i = [l_1^i, l_2^i, \ldots, l_8^i]^T$, which means the scheduling order is from lane 1 to lane 8. Accordingly, the whole action set can be recorded in matrix form

$$\text{ACTION} = \left(A_1^i, A_2^i, A_3^i, \ldots, A_{N_t^i!}^i\right)^T. \tag{16}$$

In the context of cooperative multiple intersection management, if all intersections are looked upon as a whole system, the dimensionality of the action set will grow exponentially as the number of intersection increases. Due to the situation that the DRL network has limited output so as to generate the corresponding value for specific action, the centralized approach is not suitable for the training process. Consequently, we employ a distributed mechanism in which each intersection maintains their own training network. This method can be extended to more intersections without considering the restriction of large action set.

3) *Reward Function:* When entering a new scheduling interval, an intersection agent observes the state s_t of road segment and selects action a_t to regulate vehicle. After that, an agent will receive a measurable feedback from environment. To quantify the effect of environment feedback, parameters like the changes of queued length, vehicle waiting time, and traffic throughput can be taken into the reward function. In our work, we introduce the definition of average delay to calculate particular reward. Once the order of vehicle scheduling is determined, we can get the FCTP t_o according to the FCTP algorithm in Algorithm 1. Assume that the vehicle in WS can navigate with specified maximum acceleration and velocity, the minimum time to crossing intersection for the vehicle is recorded as t_m. Subsequently, delay refers to the time difference between t_o and t_m. The average delay for N_t^i vehicles at intersection i is indicated as

$$\overline{T_d} = \frac{1}{N_t^i}\sum_{k=1}^{N_t^i} \text{delay}(k) = \frac{1}{N_t^i}\sum_{k=1}^{N_t^i} (t_o - t_m). \tag{17}$$

For the implication of delay, the greater value of delay, the more time the vehicle will spend in passing through the intersection. Therefore, we take the negative value of average vehicle delay to represent the reward function which is expressed as follows:

$$r(s, a) = -\overline{T_d} = -\frac{1}{N_t^i}\sum_{k=1}^{N_t^i} \text{delay}(k). \tag{18}$$

Next, the multiagent-based DRL scheduling mechanism is leveraged for the multi-intersection system. Each intersection agent uses independent DRL network for obtaining an optimal scheduling policy in dynamic traffic environment. The feature of fixed Q-targets is exploited in DRL network, i.e., using two NNs with the same structure but different parameters during the training process. The evaluation network that predicts the Q-value of current state–action pair has the latest parameters θ, and the target network that calculates the target Q-value using old parameters θ'. For intersection i, the target Q-value is formulated as

$$y_t = r^i(s_t, a_t) + \gamma \mathbf{max} Q_t^i\left(s_{t+1}, a_{t+1}|\theta_i'\right). \tag{19}$$

Correspondingly, using the framework of the evaluation network and target network, the stochastic incremental updating of the Q-value in (8) can be changed as

$$\begin{aligned} Q_{t+1}^i(s_t, a_t) = {} & Q_t^i\left(s_t, a_t|\theta_i\right) \\ & + \alpha(t)\left[r^i(s_t, a_t) + \gamma \mathbf{max} Q_t^i\left(s_{t+1}, a_{t+1}|\theta_i'\right)\right. \\ & \left. - Q_t^i\left(s_t, a_t|\theta_i\right)\right]. \end{aligned} \tag{20}$$

In order to realize cooperative multi-intersection management, each intersection is modeled as an intelligent agent and they can set up communication connection and exchange information with each other. When current intersection i is about to update its Q-value, neighboring agents will send their latest Q-value and the required traffic information to intersection i. Therefore, the optimal action for intersection i in the current state not only relies on its own Q-value but also affected by the action of adjacent intersections. The Q-value transfer process can be reflected in the Q-value updating of intersection i, so formulation (20) is converted to

$\overline{Q_{t+1}^i}(s_t, a_t)$

$$
\begin{aligned}
&= Q_{t+1}^{i}(s_t, a_t) + \sum\nolimits_{j\in M} \mu(i,j) Q_t^j\left(s_t, a_t|\theta_j\right) \\
&= Q_t^i\left(s_t, a_t|\theta_i\right) + \alpha(t)\big[\, r^i(s_t, a_t) \\
&\qquad + \gamma \mathbf{max} Q_t^i\left(s_{t+1}, a_{t+1}|\theta_i'\right) - Q_t^i(s_t, a_t|\theta_i)\big] \\
&\quad + \sum\nolimits_{j\in M} \mu(i,j) Q_t^j\left(s_t, a_t|\theta_j\right) \qquad (21)
\end{aligned}
$$

where M is the number of adjacent agents for intersection i and $\mu(i,j)$ is the weight factor of Q-value between agent i and agent j. The simplest way to calculated weight $\mu(i,j)$ is to consider the total number of adjacent agents and generate uniform distribution for them, i.e.,

$$
\mu(i,j) = 1/M, \quad \sum\nolimits_{j\in M} \mu(i,j) = 1. \qquad (22)
$$

In this paper, we use the strategy mentioned in [18] to calculate the value of $\mu(i,j)$. Assume that the distance between intersection i and intersection j is D, and traffic density of intersection j is I. $\mu(i,j)$ is set to be negatively correlated with D and positively correlated with I, which means that the closer to intersection i and more vehicles in adjacent intersection j, the greater $\mu(i,j)$ is. Denote $w(D,I)$ as the specific value of $\mu(i,j)$, so we have

$$
\begin{aligned}
\mu(i,j) &= w(D,I), \quad \sum\nolimits_{j\in M} \mu(i,j) = 1 \\
w(D,I) &\propto 1/D \quad \&\& \quad w(D,I) \propto I. \qquad (23)
\end{aligned}
$$

Therefore, after receiving the Q-value together with intersection distance and traffic density from adjacent agent, intersection i can apply all the weighted Q-value $\mu(i,j)Q_t^j(s_t, a_t|\theta_j)$ $(j \in M)$ into its own vehicle scheduling decision-making process so as to realize cooperative multiple intersection management.

With regard to the above-mentioned conformation, we propose an MA-DRLS algorithm in the scenario of multi-intersection. The contents of the MA-DRLS algorithm is shown in Algorithm 2. To update the parameters of the evaluation network and target network, replay memory $\overline{D}$ with capacity $\overline{N}$ is adopted to exploit the old experience during the training process. For an isolated intersection, mean square error (MSE) is taken as a loss function which can use the feature of backpropagation to update NN. The fundamental objective of the loss function is to make target Q-value $r^i + \gamma \max Q_t^i(s_{t+1}, a_{t+1}|\theta_i')$ closer to evaluated Q-value $Q_t^i(s_t, a_t|\theta_i)$. Therefore, the loss function of MSE for each agent is expressed as follows:

$$
\begin{aligned}
\mathrm{MSE}(\theta_i) = 1/K \sum_{k=1}^{K} \big[& r^i(s_t, a_t) + \gamma \mathbf{max} Q_t^i\left(s_{t+1}, a_{t+1}|\theta_i'\right) \\
& - Q_t^i\left(s_t, a_t|\theta_i\right) \big]^2. \qquad (24)
\end{aligned}
$$

Considering the influence of neighboring intersections, the Q-value transfer process from adjacent agents to the current intersection can be embedded into its loss function so that the optimal decision depends on both its own action and adjacent intersections' action. So (24) is redefined as

$$
\begin{aligned}
\mathrm{MSE}(\theta_i) = 1/K \sum_{k=1}^{K} \Big[& r^i(s_t, a_t) + \gamma \mathbf{max} Q_t^i\left(s_{t+1}, a_{t+1}|\theta_i'\right) \\
& + \sum\nolimits_{j\in M} \mu(i,j) Q_t^j\left(s_t, a_t|\theta_j\right) - Q_t^i\left(s_t, a_t|\theta_i\right) \Big]^2. \qquad (25)
\end{aligned}
$$

Algorithm 2: MA-DRLS

for *Each agent (take intersection i for example)* **do**
  Initialize replay memory $\overline{D}$ with capacity $\overline{N}$
  Initialize evaluation network with random weights θ_i target network with θ_i'
  Initialize discounted factor γ, learning rate α, maximum training step $\overline{T}$ and probability ε
end
For each episode, repeat the following procedure:
Observe the initial vehicle state s_t of intersection i
for *step* = 1 *to* $\overline{T}$ **do**
  With probability $1-\varepsilon$ randomly select an action a_t
  Otherwise select action $a_t = \arg\max Q_t^i(s_t, a_t|\theta_i)$
  Implement action a_t, obtain reward $r^i(s_t, a_t)$ and observe next state s_{t+1}
  Receive the Q-value of adjacent agents, and let $\hat{Q}_t^M = \sum_{j\in M} \mu(i,j) Q_t^j(s_t, a_t|\theta_j)$
  Store transition $(s_t, a_t, r^i, s_{t+1}, \hat{Q}_t^M)$ in $\overline{D}$
  if *length*($\overline{D}$) > *capacity* $\overline{N}$ **then**
    Delete the oldest experience
  end
  if *step* > 200 && *step* % 5 == 0 **then**
    Sample random minibatch of transition $(s_t, a_t, r^i, s_{t+1}, \hat{Q}_t^M)$ in $\overline{D}$
    With RMSProp optimizer to update parameter θ_i using the loss function $MSE(\theta_i) = 1/K \sum_{k=1}^{K} [r^i(s_t, a_t) + \gamma \mathbf{max} Q_t^i(s_{t+1}, a_{t+1}|\theta_i') - Q_t^i(s_t, a_t|\theta_i) + \sum_{j\in M} \mu(i,j) Q_t^j(s_t, a_t|\theta_j)]^2$
    Every C updating steps, let $\theta_i' = \theta_i$
  end
  if ε < *max_ε (i.e. 0.9 in this paper)* **then**
    $\varepsilon = \varepsilon + 0.000625$ (ε*_increment*)
  end
end
Return the updated evaluation and target network

In the MA-DRLS algorithm, the concrete network structure for each intersection is shown in Fig. 4. Evaluation network and target network share the same four-layer NN construction. To clearly elaborate the traffic state, 30 road segments of each lane are extracted to observe the coming vehicles. For an isolated intersection with eight lanes, the total 240 state elements constitute a column vector which is taken as the input of NN. After receiving the Q-value of adjacent intersections $\hat{Q}_t^M$ (the definition is in Algorithm 2), we store the transition of $(s_t, a_t, r^i, s_{t+1}, \hat{Q}_t^M)$ in replay memory $\overline{D}$ and randomly sample transitions for minibatch training. The purpose of the random sample process is to cut off the correlation among successive transition sequence. For a certain state, we use the ε-greedy algorithm to choose the current action a_t, that is, with probability ε select the optimal action and with probability $1-\varepsilon$ randomly select an action. The expression of ε-greedy is in the following:

$$
\pi(a_t|s_t) = \begin{cases} (1-\varepsilon)/n + \varepsilon, & a_t = \arg\mathbf{max} Q_t^i\left(s_t, a_t|\theta_i\right) \\ (1-\varepsilon)/n, & a_t \neq \arg\mathbf{max} Q_t^i\left(s_t, a_t|\theta_i\right). \end{cases} \qquad (26)
$$

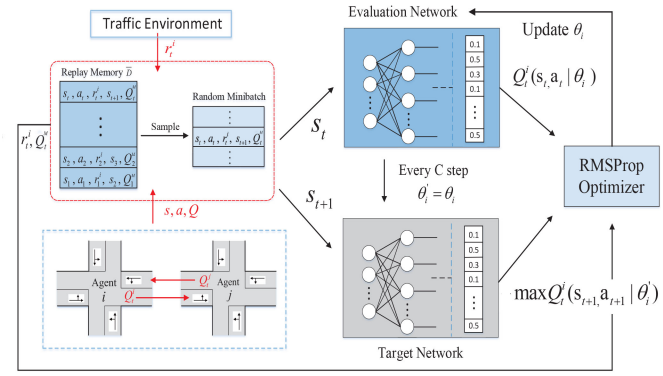

Fig. 4. DRL network for agent *i*.

TABLE II
PARAMETERS SETTINGS

Parameters	Values	Parameters	Values
Width of the CS	50 m	Length of the WS	80 m
Max velocity in CS	18 m/s	Min velocity in CS	10 m/s
Max velocity in WS	20 m/s	Min velocity in WS	0 m/s
Max acceleration	4 m/s^2	Min acceleration	-3 m/s^2
Average vehicle length	4 m	Length of the segment d	5 m
Vehicle safety time	2 s	Scheduling interval	2 s

In order to balance the tradeoff between exploitation and exploration in the learning process, the ε increment strategy is leveraged in this paper. ε is first initialized as 0. If ε in the current step is less than the preset max_ε, an extra value of ε_increment will be added into ε, i.e.,

$$\varepsilon = \varepsilon + \varepsilon\_\text{increment} \quad \text{if } \varepsilon < \max\_\varepsilon \text{ else } \max\_\varepsilon. \tag{27}$$

As for the other parameters setting of NN, the root mean square prop (RMSProp) optimizer is exploited to perform gradient descent on the loss function, which can help to make the loss function fast converge. The discounted factor γ is 0.9 and learning rate α is set as 0.002. In each episode, we first update the parameter θ_i of the evaluation network and fix the parameter θ_i' of the target network. After C training steps, θ_i in the evaluation network is assigned to target network parameters θ_i'. The value of ε_increment in the ε-greedy method is set as 0.000625. For our MA-DRLS algorithm, C is designated as 200. Finally, the output of NN is the specific value of state–action pair and we can obtain an optimal policy from the trained network in the corresponding traffic state.

V. PERFORMANCE EVALUATION

A. Parameter Setting and Evaluation Metrics

To evaluate the performance of FCTP and MA-DRLS algorithms, we carry out the simulation experiments in the PYTHON environment and the module of Tensorflow 1.14.0 is adopted to the construction of NN. Vehicle traffic data is generated by Poisson distribution and the specific parameter setting is shown in Table II.

For each intersection, the width of CS is set as 50 m and the length of WS is 80 m. To make the simulation environment closer to the practical situation, the constant velocity in CS varied from 10 to 18 m/s (i.e., 36–64.8 km/h). Vehicles can adjust their navigating speed according to the preset maximal acceleration 4 m/s^2 and minimal acceleration -3 m/s^2 in WS. As for the traffic state, we indicate the average vehicle length as 4 m and the identical length of the rectangular road segment is 5 m so as to clearly record the location of the vehicle. Vehicle safety time is set to 2 s for collision avoidance between the adjacent vehicles in the same lane. The scheduling interval is also 2 s so that we can ensure there is at most one vehicle in each lane during one dispatch process.

In the scenario of multiple intersection control, we consider a four connected intersection system for simulation conduction, in which vehicles can navigate freely according to their start position and final destination. The distance of two neighboring intersections is set as 500 m. Each intersection agent is under the communication range of adjacent agents, which can provide reliable real-time service for Q-value information exchanging. We give four prime metrics to evaluate the performance of our proposed algorithm and the detailed definition is as follows.

1) *Vehicle Delay:* It refers to the time difference between optimal FCTP t_o (calculated by the FCTP algorithm) and the minimum time t_e to cross intersection (calculated by the specified maximum acceleration and velocity).
2) *Intersection Throughput:* It refers to the number of vehicles that navigate out the management system in a period of statistical time. For an isolated intersection, there are eight lanes to leave the intersection while for four-intersection scenario there are total 16 lanes.
3) *Vehicle Waiting Time:* The vehicle waiting time starts from the time vehicle enters the WS to the time vehicle begins to pass through the intersection.
4) *Computation Time:* The computation time refers to the time cost in NN parameters updating during one training session. Less computation time overhead means greater stability of intersection scheduling.

B. Simulation Performance Comparisons

1) Isolated Intersection: To test the performance of nonsignalized intersection management, we first apply our solution into isolated intersection by combining the FCTP algorithm with the DRL network. Isolated intersection uses the same neural structure in Fig. 4 except for the Q-value coming from adjacent intersections. We compare DRL-based nonsignalized intersection management (DRL-NIM) with stochastic order-based nonsignalized intersection management (SO-NIM) in the context of intersection throughput and vehicle delay. Both DRL-NIM and SO-NIM algorithms exploit the FCTP algorithm to compute the crossing time point for the corresponding vehicle.

Fig. 5 demonstrates typical changing trends for isolated intersection throughput and vehicle delay through DRL training under the traffic density of v/3s (one vehicle coming every 3 s on average). Intersection throughput statistics every 500 s. It can be clearly seen in Fig. 5(a) that intersection throughput increasingly ascends under the continuous training of the DRL network. Average vehicle delay is calculated by the vehicles in the current scheduling interval. The light blue curve

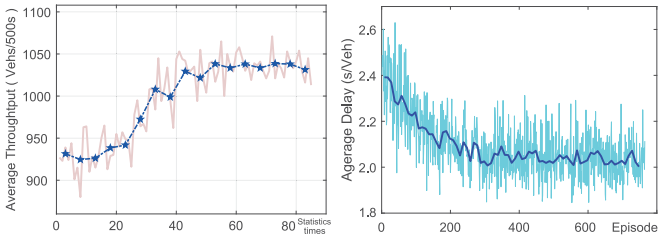

Fig. 5. Throughput and vehicle delay for an isolated intersection. (a) Isolated intersection throughput. (b) Average vehicle delay.

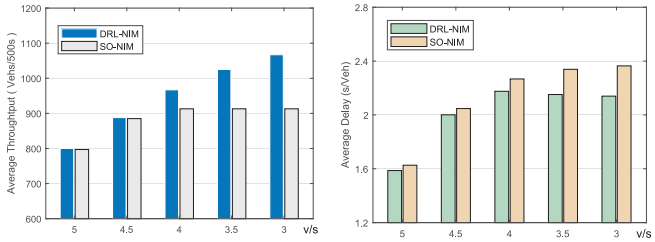

Fig. 6. Average vehicle throughput and delay under different densities. (a) Averaged intersection throughput. (b) Average vehicle delay.

in Fig. 5(b) represents the actual average delay value under each round of scheduling and the dark blue curve shows the mean values of average vehicle delay. From this figure, we can see that the curve of delay gradually decreases and reaches a relatively stable state, which means the proposed DRL method is effective in isolated intersection management. Note that the average vehicle delay and the decreasing value are relatively small, we give the following explanation: in nonsignalized intersection, vehicles in different lanes can cross intersection simultaneously without the constraints of the traffic light. Therefore, for the average delay of vehicle in eight lanes, it ought to possess a relatively small value.

Fig. 6 shows the comparison of DRL-NIM and SO-NIM algorithms under different traffic densities from v/5s to v/3s. The simulation data is gathered into ten independent runs. In Fig. 6(a) and (b), we can know that as the traffic density becomes larger, DRL-NIM has better throughput performance and less vehicle delay than the SO-NIM algorithm. For the SO-NIM algorithm, it can provide basic nonsignalized intersection management under low-density traffic environment while DRL-NIS has the ability in dealing with more congested problems. Note that the average vehicle delay in 3.5 v/s and 3 v/s vehicle density is slightly lower than 4 v/s vehicle density, we give the following explanation: when the traffic density grows to a certain degree, the scheduling priority has a greater influence on the value of average vehicle delay. Due to the situation that the DRL-NIM algorithm can generate a better management policy for scheduling priority when there are more vehicles going to pass through the intersection, it is rational that the average vehicle delay in 3.5 v/s and 3 v/s vehicle density is slightly lower than 4 v/s vehicle density. We choose a comparatively dense traffic density because our previous work [29] has validated the scheduling effectiveness of nonsignalized intersection in low-traffic density, i.e., once there is a coming vehicle, we can immediately dispatch it to cross the intersection. Thus, we do not need to improve the throughput under light traffic conditions.

2) Simulation for Multiple Intersections: In order to demonstrate the performance of the proposed MA-DRLS algorithm in this paper, stochastic order-based multiple nonsignalized intersection management (SO-MNIM) and QT-CDQN algorithm in [18] are introduced. The SO-MNIM algorithm is the extension of the SO-NIM algorithm from isolated intersection to multiple intersections. The QT-CDQN algorithm leverages a multiagent-based DRL network in the scenario of adaptive traffic signal control. In the connected four-intersection system, each intersection maintains a constant CS velocity. Due to the situation that CS velocity can affect the vehicle crossing strategy in nonsignalized intersection, identical CS velocity and different CS velocity cases are taken into the multi-intersection system.

Fig. 7 shows the system throughput performance of MA-DRLS, SO-MNIM, and QT-CDQN algorithms under identical CS velocity (12 m/s), identical CS velocity (15 m/s), and different CS velocity (range from 12 to 15 m/s) in (a)–(c), respectively. For adaptive signalized control, QT-CDQN can increase traffic throughput from around 1600 vehicles every 500 s to around 1850 vehicles every 500 s. The velocity factor of CS does not have too much effect on the QT-CDQN algorithm for the reasons that the throughput of adaptive signal control is mainly decided by the signal sequence and duration time. For the SO-MNIM algorithm, the system throughput will fluctuate as the velocity changes. With greater velocity, vehicles can pass through the intersection more quickly so as to have higher throughput. As for the proposed MA-DRLS algorithm, we can see that vehicle throughput has a significant enhancement in both identical CS velocity and different CS velocity. Meanwhile, the MA-DRLS algorithm has the highest system throughput compared with SO-MNIM and QT-CDQN algorithms in all traffic environment.

Fig. 8 is the cumulative distribution function (CDF) of the vehicle waiting time under different scheduling algorithms. The vertical axis of CDF represents the probability of waiting time which ranges from 0 to 1, and the horizontal axis represents the specific value of waiting time. We can see from these figures that an adaptive signal control QT-CDQN algorithm has the longest waiting time. In almost every situation, the average vehicle waiting time of QT-CDQN is between 60 and 80 s, while for nonsignalized intersection scenario, the waiting time of MA-DRLS and SO-MNIM is basically within 30 s. In Fig. 8(a), the small figure is depicted to show the difference of MA-DRLS and SO-MNIM under identical CS velocity of 12 m/s. It can be clearly seen that the waiting time of MA-DRLS is less than SO-MNIM. In Fig. 8(b), the small figure indicates the actual waiting time value of each vehicle in 15 m/s CS velocity. The purple line represents each vehicle waiting time value of SO-MNIM and the blue line represents the MA-DRLS algorithm. We can intuitively know that the purple line has a greater value than the blue line. For the case of different CS velocities, we plot the waiting time CDF of MA-DRLS and SO-MNIM under 12 and 15 m/s in the small figure of Fig. 8(c). All of them

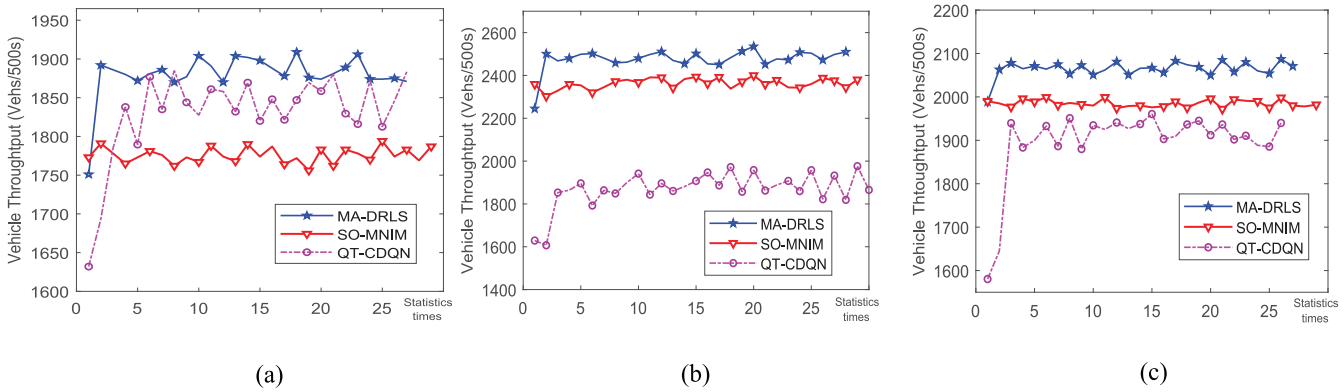

Fig. 7. Throughput of multi-intersection system under different CS velocity situations. (a) Identical CS velocity (12 m/s). (b) Identical CS velocity (15 m/s). (c) Different CS velocity (12 and 15 m/s).

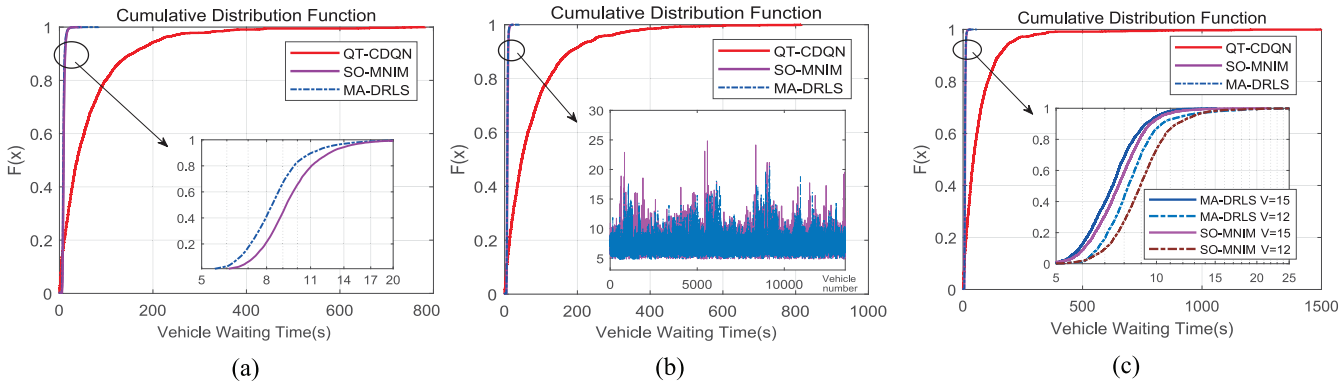

Fig. 8. CDFs of vehicle waiting time under different CS velocity situations. (a) Identical CS velocity (12 m/s). (b) Identical CS velocity (15 m/s). (c) Different CS velocity (12 and 15 m/s).

have shown that the MA-DRLS algorithm slightly outperforms the SO-MNIM algorithm. The situation that vehicle waiting time of MA-DRLS and SO-MNIM possess small value is also due to the efficient scheduling mechanism of nonsignalized intersection. In conclusion, from the simulation demonstration of multi-intersection throughput and vehicle waiting time, we have verified that MA-DRLS can significantly improve traffic efficiency and reduce time cost for intersection management. Although QT-CDQN can enhance scheduling performance to a certain degree, there still exists upper limit scheduling capacity for adaptive signal control and it is powerless against solving the fundamental deficiency of traffic light.

3) Multiagent Learning and Independent Learning: To investigate the effectiveness of the multiagent scheduling mechanism, we perform an independent DRL training method at each intersection in the connected four-intersection system. Each intersection obtains the optimal scheduling policy without receiving the Q-value from the adjacent intersection. The independent training uses the same DRL network as the MA-DRLS algorithm.

Fig. 9 shows the intersection throughput comparison for MA-DRLS and independent DRL training under identical CS velocity and different CS velocity situation. The simulation data is also generated by ten separated runs. It can be seen that MA-DRLS has higher intersection throughput than independent DRL training in both of the two situations. For clearly elucidating the specific throughput value, we plot a classical

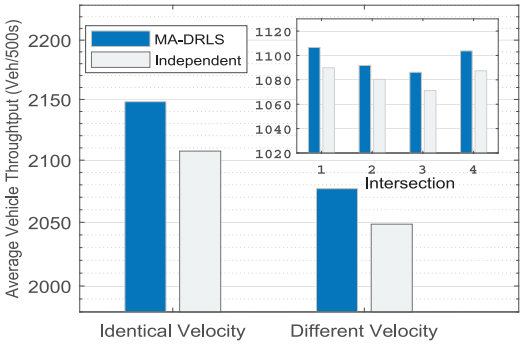

Fig. 9. Comparison of intersection throughput.

comparison of each intersection under identical 12 m/s CS velocity in the small figure of Fig. 9. The throughput of each intersection varies from each other but MA-DRLS has a better throughput performance in overall statistics.

With traffic efficiency guaranteed, we then investigate the computation overhead of proposed algorithms. Fig. 10 shows the average computation time of total four intersection during one training session. For MA-DRLS and independent DRL method, the computation time is calculated by 10^{-3} (s) magnitude, which means that our solution will not bring much effect on vehicle navigation and can provide reliable service for intersection management. The concrete computation time value of the four intersection is specified in Fig. 10 and the small figure shows the practical time value of one intersection

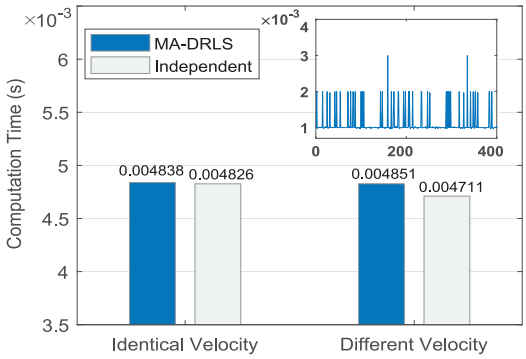

Fig. 10. Comparison of computation overhead.

under each train step. We can know that the computation time of MA-DRLS and independent DRL method is almost the same for both identical velocity and different velocity situation. Combined with the throughput performance in Fig. 9, we have the conclusion that the MA-DRLS algorithm has a better intersection throughput without additional computational overhead, which is effective in realizing cooperative multi-intersection scheduling.

VI. Conclusion and Future Work

In this paper, we have presented a vehicle–road collaboration-enabled nonsignalized intersection management architecture to provide safety and efficient crossing solution for automated vehicles. In order to realize cooperative multi-intersection management, FCTP and MA-DRLS algorithms have been proposed to enhance the efficiency of traffic management. Extensive simulations have demonstrated the superiority of nonsignalized intersection management solution. This research work will be beneficial for promoting the engineering applications of the modern intelligent transportation system. For our future work, we will further investigate the coordination between adjacent intersections and extend optimal scheduling solution for larger scale nonsignalized intersection management.

References

[1] Z. MacHardy, A. Khan, K. Obana, and S. Iwashina, "V2X access technologies: Regulation, research, and remaining challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1858–1877, 3rd Quart., 2018.

[2] H. Zhou, W. Xu, J. Chen, and W. Wang, "Evolutionary V2X technologies toward the Internet of Vehicles: Challenges and opportunities," *Proc. IEEE*, vol. 108, no. 2, pp. 308–323, Feb. 2020.

[3] X. Liang, T. Yan, J. Lee, and G. Wang, "A distributed intersection management protocol for safety, efficiency, and driver's comfort," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1924–1935, Jun. 2018.

[4] A. P. Chouhan and G. Banda, "Autonomous intersection management: A heuristic approach," *IEEE Access*, vol. 6, pp. 53287–53295, 2018.

[5] H. Peng *et al.*, "Resource allocation for cellular-based inter-vehicle communications in autonomous multiplatoons," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 11249–11263, Dec. 2017.

[6] J. Rios-Torres and A. A. Malikopoulos, "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1066–1077, May 2017.

[7] S. Chen and D. J. Sun, "An improved adaptive signal control method for isolated signalized intersection based on dynamic programming," *IEEE Intell. Transp. Syst. Mag.*, vol. 8, no. 4, pp. 4–14, Oct. 2016.

[8] Y. Ren, Y. Wang, G. Yu, H. Liu, and L. Xiao, "An adaptive signal control scheme to prevent intersection traffic blockage," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1519–1528, Jun. 2017.

[9] X. Peng, H. Zhou, B. Qian, K. Yu, F. Lyu, and W. Xu, "Enabling security-aware D2D spectrum resource sharing for connected autonomous vehicles," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3799–3811, May 2020.

[10] S. Park, B. Kim, H. Yoon, and S. Choi, "Ra-eV2V: Relaying systems for LTE-V2V communications," *J. Commun. Netw.*, vol. 20, no. 4, pp. 396–405, Aug. 2018.

[11] N. Cheng *et al.*, "Big data driven vehicular networks," *IEEE Netw.*, vol. 32, no. 6, pp. 160–167, Nov./Dec. 2018.

[12] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 289–299, Aug. 2014.

[13] Y. Meng, L. Li, F.-Y. Wang, K. Li, and Z. Li, "Analysis of cooperative driving strategies for nonsignalized intersections," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 2900–2911, Apr. 2018.

[14] M. S. Shirazi and B. T. Morris, "Looking at intersections: A survey of intersection monitoring, behavior and safety analysis of recent studies," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 1, pp. 4–24, Jan. 2017.

[15] W. Xu *et al.*, "Internet of Vehicles in big data era," *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

[16] M. Chen, T. Wang, K. Ota, M. Dong, M. Zhao, and A. Liu, "Intelligent resource allocation management for vehicles network: An A3C learning approach," *Comput. Commun.*, vol. 151, pp. 485–494, Feb. 2020.

[17] M. Zhou, Y. Yu, and X. Qu, "Development of an efficient driving strategy for connected and automated vehicles at signalized intersections: A reinforcement learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 433–443, Jan. 2020.

[18] H. Ge, Y. Song, C. Wu, J. Ren, and G. Tan, "Cooperative deep Q-learning with Q-value transfer for multi-intersection signal control," *IEEE Access*, vol. 7, pp. 40797–40809, 2019.

[19] O. Younis and N. Moayeri, "Employing cyber-physical systems: Dynamic traffic light control at road intersections," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2286–2296, Dec. 2017.

[20] J. Wu, D. Ghosal, M. Zhang, and C.-N. Chuah, "Delay-based traffic signal control for throughput optimality and fairness at an isolated intersection," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 896–909, Feb. 2018.

[21] S. Araghi, A. Khosravi, and D. Creighton, "A review on computational intelligence methods for controlling traffic signal timing," *Expert Syst. Appl.*, vol. 42, no. 3, pp. 1538–1550, Jan. 2015.

[22] C.-H. Wan and M.-C. Hwang, "Value-based deep reinforcement learning for adaptive isolated intersection signal control," *IET Intell. Transp. Syst.*, vol. 12, no. 9, pp. 1005–1010, Aug. 2018.

[23] S. Li, C. Wei, X. Yan, L. Ma, D. Chen, and Y. Wang, "A deep adaptive traffic signal controller with long-term planning horizon and spatial-temporal state definition under dynamic traffic fluctuations," *IEEE Access*, vol. 8, pp. 37087–37104, 2020.

[24] M. A. S. Kamal, T. Hayakawa, and J.-I. Imura, "Development and evaluation of an adaptive traffic signal control scheme under a mixed-automated traffic scenario," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 590–602, Feb. 2020.

[25] M. A. S. Kamal, J.-I. Imura, T. Hayakawa, A. Ohata, and K. Aihara, "A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 3, pp. 1136–1147, Jun. 2015.

[26] K. Zhang, D. Zhang, A. de La Fortelle, X. Wu, and J. Grégoire, "State-driven priority scheduling mechanisms for driverless vehicles approaching intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2487–2500, Oct. 2015.

[27] P. Dai, K. Liu, Q. Zhuge, E. H.-M. Sha, V. C. S. Lee, and S. H. Son, "Quality-of-experience-oriented autonomous intersection control in vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 1956–1967, Jul. 2016.

[28] M. O. Sayin, C.-W. Lin, S. Shiraishi, J. Shen, and T. Başar, "Information-driven autonomous intersection control via incentive compatible mechanisms," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 912–924, Mar. 2019.

[29] B. Qian, H. Zhou, F. Lyu, J. Li, T. Ma, and F. Hou, "Toward collision-free and efficient coordination for automated vehicles at unsignalized intersection," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10408–10420, Dec. 2019.

[30] J. Wang, X. Zhao, and G. Yin, "Multi-objective optimal cooperative driving for connected and automated vehicles at non-signalised intersection," *IET Intell. Trans. Syst.*, vol. 13, no. 1, pp. 79–89, Jan. 2019.

[31] M. B. Younes and A. Boukerche, "Intelligent traffic light controlling algorithms using vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 5887–5899, Aug. 2016.
[32] Y. Hou, G. Wang, and Y. Zhou, "Virtual-grid based traffic control strategy with multiple intersections collaboration," *IEEE Access*, vol. 6, pp. 40105–40119, 2018.
[33] H. Yang, F. Almutairi, and H. Rakha, "Eco-driving at signalized intersections: A multiple signal optimization approach," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 9, 2020, doi: 10.1109/TITS.2020.2978184.
[34] Q. Lin *et al.*, "Minimize the fuel consumption of connected vehicles between two red-signalized intersections in urban traffic," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9060–9072, Oct. 2018.
[35] S. Lee, M. Younis, A. Murali, and M. Lee, "Dynamic local vehicular flow optimization using real-time traffic conditions at multiple road intersections," *IEEE Access*, vol. 7, pp. 28137–28157, 2019.
[36] W. Liu, G. Qin, Y. He, and F. Jiang, "Distributed cooperative reinforcement learning-based traffic signal control that integrates V2X networks' dynamic clustering," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 8667–8681, Oct. 2017.
[37] H. Li, K. Ota, and M. Dong, "Deep reinforcement scheduling for mobile crowdsensing in fog computing," *ACM Trans. Internet Technol.*, vol. 19, no. 2, pp. 1–18, Apr. 2019.

Yunting Xu (Student Member, IEEE) received the B.S. degree in communication engineering from Nanjing University, Nanjing, China, in 2017, where he is currently pursuing the Ph.D. degree with the School of Electronic Science and Engineering.

He mainly focuses on the dynamic resource management and networking optimization in the field of emerging wireless networks.

Haibo Zhou (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from Shanghai Jiao Tong University, Shanghai, China, in 2014.

From 2014 to 2017, he was a Postdoctoral Fellow with the Broadband Communications Research Group, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. He is currently an Associate Professor with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China. His research interests include resource management and protocol design in vehicular *ad hoc* networks, cognitive networks, and space–air–ground integrated networks.

Dr. Zhou was a recipient of the 2019 IEEE ComSoc Asia–Pacific Outstanding Young Researcher Award. He served as an Invited Track Co-Chair for ICCC'2019 and VTC-Fall'2020 and a TPC Member of many IEEE conferences, including GLOBECOM, ICC, and VTC. He served as an Associate Editor for the IEEE Comsoc Technically Co-Sponsored the *Journal of Communications and Information Networks* from April 2017 to March 2019, and a Guest Editor for the *IEEE Communications Magazine* in 2016, the *International Journal of Distributed Sensor Networks* (Hindawi) in 2017, and *IET Communications* in 2017. He is currently an Associate Editor of the IEEE INTERNET OF THINGS JOURNAL, the *IEEE Network Magazine*, and the IEEE WIRELESS COMMUNICATIONS LETTERS.

Ting Ma (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in statistics from Sichuan University, Chengdu, China, in 2013, 2016, and 2020, respectively.

She is currently a Postdoctoral Fellow with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China. Her current research interests mainly include robust hypothesis testing, space–air–ground integrated network, convex optimization theory, and game theory.

Jiwei Zhao (Student Member, IEEE) received the M.S. degree in information and communication system from Xidian University, Xi'an, China, in 2018. He is currently pursuing the Ph.D. degree with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China.

His current research interests include fully decoupled RAN architecture, coordinated multipoint, and machine learning applications for wireless communication.

Mr. Zhao won the First Prize in the 2016 China Computer Federation China Big Data and Cloud Computing Intelligence Contest.

Bo Qian (Student Member, IEEE) received the B.S. and M.S. degrees in statistics from Sichuan University, Chengdu, China, in 2015 and 2018, respectively. He is currently pursuing the Ph.D. degree with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China.

His current research interests include intelligent transportation systems and vehicular networks, wireless resource management, blockchain, convex optimization theory, and game theory.

Mr. Qian was a recipient of the Best Paper Award from IEEE VTC2020-Fall.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular *ad hoc* and sensor networks.

Prof. Shen received the R. A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) presents in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society, and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013. He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, and IEEE Globecom'07, the Symposia Chair for IEEE ICC'10, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the elected IEEE Communications Society Vice President for Technical and Educational Activities, the Vice President for Publications, the Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a member of the IEEE Fellow Selection Committee. He was/is the Editor-in-Chief of the IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, *IET Communications*, and *Peer-to-Peer Networking and Applications*. He is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.