Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 1820527, 12 pages
https://doi.org/10.1155/2020/1820527

Research Article

Cooperative Multiagent Deep Deterministic Policy Gradient (CoMADDPG) for Intelligent Connected Transportation with Unsignalized Intersection

Tianhao Wu,[1] Mingzhi Jiang,[1] and Lin Zhang [2,1]

[1]*School of Information & Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*Beijing Information Science and Technology University, Beijing 100192, China*

Correspondence should be addressed to Lin Zhang; zhl@bistu.edu.cn

Received 20 April 2020; Revised 9 June 2020; Accepted 23 June 2020; Published 22 July 2020

Guest Editor: Chi-Hua Chen

Unsignalized intersection control is one of the most critical issues in intelligent transportation systems, which requires connected and automated vehicles to support more frequent information interaction and on-board computing. It is very promising to introduce reinforcement learning in the unsignalized intersection control. However, the existing multiagent reinforcement learning algorithms, such as multiagent deep deterministic policy gradient (MADDPG), hardly handle a dynamic number of vehicles, which cannot meet the need of the real road condition. Thus, this paper proposes a Cooperative MADDPG (CoMADDPG) for connected vehicles at unsignalized intersection to solve this problem. Firstly, the scenario of multiple vehicles passing through an unsignalized intersection is formulated as a multiagent reinforcement learning (RL) problem. Secondly, MADDPG is redefined to adapt to the dynamic quantity agents, where each vehicle selects reference vehicles to construct a partial stationary environment, which is necessary for RL. Thirdly, this paper incorporates a novel vehicle selection method, which projects the reference vehicles on a virtual lane and selects the largest impact vehicles to construct the environment. At last, an intersection simulation platform is developed to evaluate the proposed method. According to the simulation result, CoMADDPG can reduce average travel time by 39.28% compared with the other optimization-based methods, which indicates that CoMADDPG has an excellent prospect in dealing with the scenario of unsignalized intersection control.

1. Introduction

In recent years, the development of connected vehicles [1] has prompted the innovation of transportation technology. By introducing the technologies of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication, multivehicle coordination has become possible to improve traffic safety and efficiency. This makes the cooperative intelligent transportation systems (C-ITS) a new research hotspot [2].

As a classic application scenario of Intelligent Connected Transportation, the intersection is more complicated and challenging for the cooperation of intervehicle control than the highway. In the intersection, vehicles will enter from different directions and leave through different exits. The conflict between different vehicles will greatly limit the efficiency and safety of the intersection traffic. Therefore, a flexible and complicated intersection control system is necessary. There have been several related works focusing on traffic signal timing optimization. Ge et al. proposed a cooperative method for multi-intersection signal control based on Q-learning with Q-value transfer [3]. Kim and Jeong proposed a cooperative traffic signal control scheme using traffic flow prediction for a multi-intersection [4]. Kamal et al. concentrated on the mixed manual-automated traffic scenario and developed an adaptive traffic signal control scheme for an isolated intersection [5]. Yu proposed a fuzzy programming-based approach to optimize the signal timing for an isolated intersection, which used operation efficiency, traffic capacity, and exhaust emission as the joint optimization goals [6]. Bai et al. proposed a scenario where trams

crossed intersections without stop and presented a coordinated control model along the tramline to optimize the control of prioritized traffic signals [7]. Xu et al. proposed a game-based policy for the signal control of an isolated intersection together with rerouting of the vehicles using vehicle-to-infrastructure communication [8]. Lu et al. presented a novel speed control method for the successive signalized intersections under connected vehicles environment. With the timing information and vehicle queues at the signalized intersection, vehicle speed was optimized to reduce fuel consumption and emissions [9]. Xu proposed a cooperative method to optimize roadside signal optimization and on-board vehicle speed control at the same time [10]. In addition, with the development of autonomous driving, the mixed traffic of manual-automated driving coexistence is also attracting researchers' attention. Boudaakat et al. established a hydrodynamic model to calculate the total capacity of roundabouts and provided a method to control traffic congestion [11]. Kamal et al. presented a novel adaptive traffic signal control scheme, aiming at minimizing the total crossing time of all vehicles and ensuring comfortable crossing of manually driven vehicles [12]. Gupta et al. proposed a conceptual model for negotiation between self-driving vehicles and pedestrians, which shows an improvement in the overall travel time of the vehicles as compared with the current best practice behaviour, always stop, of autonomous vehicles [13]. In general, traffic signal control schemes should maximize traffic efficiency under the premise of ensuring traffic safety. To make the intersection traffic control more accurate and in real time, many researchers have begun to focus on the study of unsignalized intersection control.

Generally, the researchers divide unsignalized intersection control schemes into two categories, centralized and distributed. Centralized coordination approaches collect the global information of the entire intersection to regulate the vehicles at the intersection. Dai et al. solve the intersection control problem with convex optimization [14]. Guan et al. proposed a centralized conflict-free cooperation method for multiple connected vehicles at unsignalized intersection using model accelerated proximal policy optimization [15]. Qian et al. devised the AIDM algorithm to schedule a vehicle or a platoon to pass the unsignalized intersection [16]. Nevertheless, these centralized schemes are pressed for communication and computation because all vehicles need a central controller to dispatch them.

In decentralized coordination approaches, the centralized controller disappears, and the controller is configured separately in each vehicle to adjust its trajectories considering kinetic information and conflict relationship of adjacent vehicles. Xu et al. introduced a conflict-free geometry topology and designed a distributed controller to stabilize the involved vehicle at the intersection [17]. Bian et al. divided the intersection area into four areas according to the distance and present a process containing observation, optimization, and control [18]. Hadjigeorgiou and Timotheou studied the optimization of the travel time and fuel consumption balance of autonomous vehicles through unsignalized intersection [19]. Belkhouche proposed decentralized multiagent control laws, which are derived for conflict resolution between vehicles [20]. Hsu et al. studied the interaction between vehicles and pedestrians at unsignalized intersections and proposed a decision theory model to represent the interaction [21]. Wang et al. proposed a cooperative algorithm to transform the high-dimensional problem of cooperative driving for multiple vehicles at multiconflict points into the single-dimensional problem of searching the optimal time for vehicles to enter the intersection [22]. Yang and Oguchi developed an advanced vehicle control system with connected vehicles to reduce vehicles delays under a partially connected environment [23]. However, well-designed models and controllers can only show interpretable parts, and more hidden parts become bottlenecks in performance improvement.

One of the most critical goals in artificial intelligence is to obtain a new skill, especially in a multiagent environment. Reinforcement learning (RL) can improve the policy via trial-and-error interaction with the environment, which is analogous to human beings. Recently, RL has taken an essential role in a variety of fields, for example, wireless communication [24] and autonomous driving [25]. Mnih et al. proposed Deep Q-learning Network (DQN) and obtained superhuman performance on Atari video games [26]. Considering that DQN is only applicable to the problem with discrete action spaces, Deep Deterministic Policy Gradient (DDPG) is proposed to solve continuous control problems [27]. When the scenario extends from a single agent to multiple agents, more information can be taken into consideration to improve algorithm performance. Multiagent DDPG (MADDPG) is a multiagent policy gradient algorithm where agents learn a centralized critic based on the observation and actions of all agents [28]. There have already been many applications in the field of intersection control. Liang et al. proposed a double-dueling deep Q-network to control the traffic light cycle [29]. Zhou et al. proposed a car-following model, based on reinforcement learning, to obtain an appropriate driving behaviour to improve travel efficiency, fuel consumption, and safety at signalized intersections [30]. Lee et al. employed reinforcement learning that recognizes an entire traffic state and jointly controls all the traffic signals of multiple intersections [31]. However, the uncertainty of agent number poses a further challenge to address some problems, such as distributed coordination at unsignalized intersection.

This paper proposes a distributed conflict-free cooperation method, Cooperative MADDPG (CoMADDPG), for multiple connected vehicles at unsignalized intersection. The main contributions of this paper are as follows:

(i) This paper formulates the scenario of multiple vehicles passing through an unsignalized intersection as a multiagent reinforcement learning problem.

(ii) The Cooperative MADDPG (CoMADDPG) is proposed, which modifies the classic MADDPG algorithm to adapt the dynamic quantity agents. In CoMADDPG, each vehicle selects reference vehicles to construct a partial stationary environment, which is necessary for the introduction of the RL method.

(iii) This paper also proposes a novel vehicle selection method, which can project all reference vehicles on a virtual lane and selects the largest impact vehicles on the virtual lane. It can assist the CoMADDPG algorithm to converge quickly and avoid collisions effectively.

The rest of this paper is organized as follows. Section 2 illustrates our problem statement and presents the settings of states, actions, and rewards. Section 3 introduces the preliminaries of multiagent reinforcement learning and the workflow of the proposed CoMADDPG algorithm. Section 4 presents the experimental settings and results. Section 5 concludes this work.

2. Problem Statement and Formulation

2.1. Problem Statement. This paper focuses on a 4-direction intersection shown in Figure 1. Each direction denotes the location in the figure, that is, up, down, left, and right, respectively. A certain distance of the intersection is focused on. There are 4 entrances and 4 exits in total, which are unsignalized, and each direction contains only one lane. The vehicles are only allowed to go straight.

As depicted in Figure 1, boxes in different colours represent vehicles in different lanes. The trajectories of the vehicles intersect into 4 conflict points in the merging zone of the intersection. Based on the collected information, each vehicle independently decides its acceleration and deceleration using a policy network. When the vehicles enter the intersection area, the centralized server distributes the newest model to them. During the running process of the vehicles, they produce so-called experience, recording the running data of themselves and reference vehicles at each time step. The reference vehicles are selected via a proposed vehicle selection method. The whole process is continuous, with vehicles entering and leaving the intersection area.

Several assumptions are adopted as follows. Firstly, the kinetic information of vehicles can be measured to support the decision made by each vehicle. Then, it is assumed that all approaching vehicles are connected and automated so that each vehicle can strictly obey the planned acceleration, adjust the velocity, and pass the intersection automatically. Additionally, all vehicles enter the intersection according to the set time, which corresponds to the Poisson process.

2.2. MARL Formulation. The problem is formulated as a multiagent reinforcement learning problem, and each vehicle is treated as an agent by defining state space, action space, and reward function.

2.2.1. State and Action Space. According to the assumptions, each vehicle can obtain others' kinetic information via vehicle-to-vehicle communication. To achieve the aim of collaboration, the state of a vehicle needs to include the dynamics of its adjacent vehicles. However, the dimensions of the state cannot be arbitrarily expanded, so several vehicles with the largest impact are chosen, as is shown in equation (1). The subscript m indicates the maximum number of the largest impact vehicles under consideration. As for the definition of $s^i_{\text{other},j}$, the subscript j represents the j^{th} largest impact vehicle of vehicle i, and it has no relationship with the identity of any vehicle:

$$s_i = \left\{ s^i_{\text{own}}, s^i_{\text{other},1}, \ldots, s^i_{\text{other},j}, \ldots, s^i_{\text{other},m} \right\}, \tag{1}$$

$$s^i_{\text{own}} = \left\{ p^i, v^i, a^i \right\}, \tag{2}$$

$$s^i_{other,*} = \left\{ p^i_*, v^i_*, a^i_* \right\}. \tag{3}$$

In equation (2), s^i_{own} represents the state of vehicle i, including position, velocity, and acceleration. In equation (3), $s^i_{other,*}$ represents the reference vehicles of vehicle i, including position, velocity, and acceleration. Commonly, the position can be formed with Cartesian coordinate, that is, (x, y). However, through the analysis of the task formulation, conflicting vehicles at the same distance from the intersection will have a high correlation. Therefore, polar coordinate (μ, θ) is utilized instead of (x, y). There are only 4 directions in the problem, so the position is denoted by (d, l). Herein, d is the distance from the vehicle to the conflict point, and it is positive when approaching the intersection and negative when leaving. l is the index of lane, that is, $\{1, 2, 3, 4\}$.

2.2.2. Reward Settings. The reward function is designed, as shown in Table 1. Compared with reward settings in [15], the reward is not defined with the final situation, such as vehicle passing or collision but scattered in the running process. Here, the distance difference Diff_D and the time difference Diff_T are used to assist each vehicle to obtain its reward:

$$\text{Diff}_D = \left| p_{\text{cur}} - p_{\text{ref}} \right|, \tag{4}$$

$$\text{Dif}f_T = \frac{p_{\text{cur}} - p_{\text{ref}}}{v_{\text{cur}} - v_{\text{ref}}}. \tag{5}$$

In equations (4) and (5), p_{cur} and p_{ref} represent the positions of the current vehicle and reference vehicle. v_{cur} and v_{ref} are the velocities of the current vehicle and reference vehicle. We set the end of the entering lane to zero. If Diff_T is greater than zero, the current vehicle and reference vehicle are approaching, and vice versa.

It is noted that as the distance difference shrinks, the risk factor does not change linearly, and a logarithmic function is introduced to describe this nonlinear change. On the other hand, the sign of time difference can be a good indicator of whether the distance between the two vehicles is increasing or decreasing. If the distance between vehicles is small enough, the increasing distance will generate positive rewards and vice versa. The transformation of the hyperbolic tangent function is a good description of this change. Moreover, to avoid reward expansion, which is an important issue in RL, we limit the reward value after each calculation and reward is controlled at $[-20, 20]$.

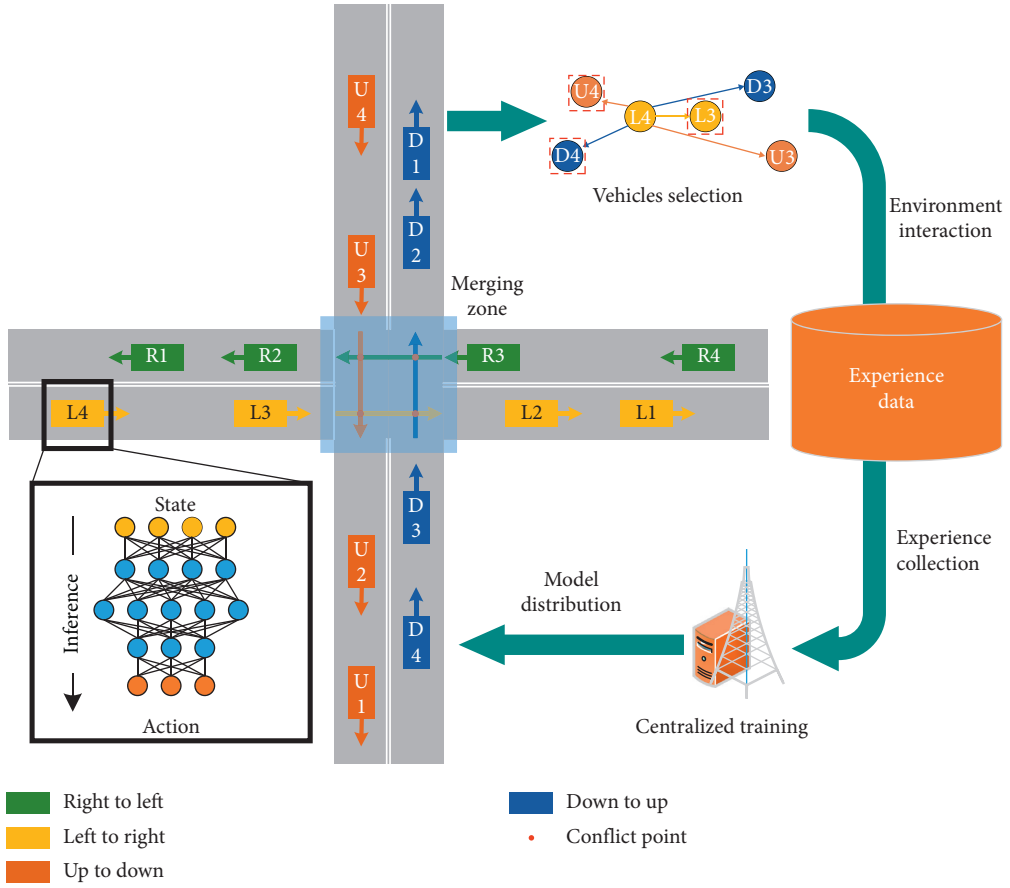

Figure 1: The overall flow diagram of cooperative reinforcement learning integrated with intersection scenario.

Table 1: Reward settings.

Reward items	Reward
Distance difference	$\log(\text{Diff}_D)$
Time difference	$1/\tanh(-\text{Diff}_T)$

3. Cooperative Multiagent Deep Deterministic Policy Gradient

In this paper, partially observable Markov games are considered, constituting a multiagent Markov decision process. The possible state $\mathcal{S}$, a set of actions $\mathcal{A}_1, \ldots, \mathcal{A}_N$, and a set of observations $\mathcal{O}_1, \ldots, \mathcal{O}_N$ jointly describe a Markov game for N agents. To determine the action, each agent utilizes a stochastic policy $\pi_{\theta_i}: \mathcal{O}_i \times \mathcal{A}_i$, which outputs the next state based on the state transition function $\mathcal{T}: \mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_N \mapsto \mathcal{S}$. After interacting with the environment, each agent obtains reward with the function of state and action $\mathcal{S} \times \mathcal{A}_1 \mapsto \mathbb{R}$ and gets a separate observation o_i. The initial state is dependent on a distribution ρ. Each agent runs for maximizing its expected return $R_i = \sum_{t=0}^{T} \gamma^t r_i^t$, where T is the time horizon and γ is a discount factor.

3.1. Multiagent Deep Deterministic Policy Gradient. A significant problem faced by the traditional RL algorithm is that each agent is learning to improve the policy continuously. Thus, from the perspective of each agent, the environment is dynamic, which is not stationary for traditional RL algorithm. To a certain extent, it is impossible to adapt to the dynamic environment by merely changing the agent's policy. Due to the instability of the environment, the critical techniques of DQN-like experience replay cannot be directly used. The policy gradient exacerbates the problem of significant variance due to the increase in the number of agents.

Then MADDPG is an adaptation of actor-critic methods which considers action policies of other agents and can learn policies that require complex multiagent coordination. The algorithm has three characteristics. Firstly, the optimal policy obtained through learning can produce optimal action using only local information. Secondly, there is no need to know the requirements of the dynamic model of the environment and interagent communication. Thirdly, the algorithm can be used in both cooperative and competitive environments.

In the game with N agents, the policies on each agent can be represented as $\mu = \{\mu_1, \ldots, \mu_N\}$, and the gradient of the expected return for agent i, $J(\theta_i) = \mathcal{E}[R_i]$, can be written as

$$\nabla_{\theta_i} J(\boldsymbol{\mu}_i) = E_{x,a\sim\mathcal{D}}\left[\nabla_{\theta_i}\boldsymbol{\mu}_i(a_i \mid o_i)\nabla_{a_i} Q_i^{\boldsymbol{\mu}}(\mathbf{x}, a_1, \ldots, a_N)\big|_{a_i=\mu_i(o_i)}\right], \tag{6}$$

where $\mathcal{D}$ is the experience replay buffer, which contains the tuples (o_i, o_i', a_i, r_i) to record the experience of each agent i.

$Q_i^{\mu}(\mathbf{x}, a_1, \ldots, a_N)$ denotes a centralized action-value function that takes the action of all agents and some state information $\mathbf{x}$ and outputs the Q-value for agent i. In some fundamental cases, $\mathbf{x}$ includes the observation of all agents: $\mathbf{x} = (o_1, \ldots, o_N)$. The centralized action-value function is updated as

$$y^i = r_i^j + \gamma Q_i^{\mu'}\left(x'^j, a'_1, \ldots, a'_N\right)\Big|_{a'_k = \mu'_k\left(o_k^j\right)}, \tag{7}$$

$$\mathscr{L}(\theta_i) = \mathscr{E}_{\mathbf{x},a,r,\mathbf{x}_l}\left(y^j - Q_i^{\boldsymbol{\mu}}\left(x^j, a_1^j, \ldots, a_N^j\right)\right)^2, \tag{8}$$

where $\boldsymbol{\mu}' = \left\{\mu_{\theta'_1}, \ldots, \mu_{\theta'_i}, \ldots, \mu_{\theta'_N}\right\}$ is the target policy set with delayed parameters θ'_i. The main intention of MADDPG is that when the actions produced by all agents are known, the environment is stationary even if the policies vary. Additionally, the algorithm has three techniques. Firstly, actor and critic constitute centralized training, and actor can run only by knowing local information during inference. Secondly, experience replay is improved to apply to a dynamic environment. Thirdly, policy ensemble is utilized to enhance stability and robustness.

3.2. Cooperative MADDPG (CoMADDPG). One of the challenges for multiagent reinforcement learning is when policies of agents are updated, the environment changes, which contradicts existing assumptions of a stationary environment. Accordingly, a primary motivation behind MADDPG is that the actions taken by all agents are known, which makes the environment be considered stationary even as the policies change. However, by default, different observation variables in MADDPG will correspond to the agents one by one, which significantly limits the application scenarios of this algorithm. In this paper, the definition of a stationary environment is extended to suit the situation of more agents.

In the problem of distributed vehicle control at the intersection, the fluent entry and exit of vehicles lead to an uncertain and large number of agents. From the perspective of a stationary environment, it can exist not only globally but also partially. To construct the environment, an agent selects several agents as reference agents. Therefore, the gradient of the expected return can be rewritten as

$$\nabla_{\theta_i} J(\boldsymbol{\mu}_i) = E_{x,a\sim\mathscr{D}}\left[\nabla_{\theta_i}\boldsymbol{\mu}_i\left(a_i \mid \vec{\mathbf{o}_i}\right)\nabla_{a_i} Q_i^{\boldsymbol{\mu}}\Big|\left(\vec{\mathbf{o}_i}, \vec{\mathbf{a}_i}\right)\Big|_{a_i=\mu_i\left(\vec{\mathbf{o}_i}\right)}\right]. \tag{9}$$

In equation (9), $\vec{\mathbf{o}_i} = \left\{o_i, o_{i_1}, \ldots, o_{i_j}, \ldots, o_{i_m}\right\}$ and $\vec{\mathbf{a}_i} = \left\{a_i, a_{i_1}, \ldots, a_{i_j}, \ldots, a_{i_m}\right\}$ denote the set of observations and the set of actions, which come from agent i and reference agents $\{i_1, \ldots, i_j, \ldots, i_m\}$. Note that the entity of each reference agent i_j may change at each time step. For instance, there are four vehicles $\{A, B, C, D\}$ moving in the same lane in sequence. If vehicle A is the current vehicle, the set of observations $\vec{\mathbf{o}_\mathbf{A}}$ can be written as $\{o_A, o_B, o_C, o_D\}$. When vehicle B overtakes vehicle C, the sequence of the vehicles becomes {A, C, B, D}, and the set of observations $\vec{\mathbf{o}_\mathbf{A}}$ becomes $\{o_A, o_C, o_B, o_D\}$. When this data enters the neural network as input, the position of the variables will play an important role. The above operation can decouple the identity and running information of the vehicles so that the proposed CoMADDPG can be adapted to the scenario of more agents.

Note that more agents will not make the decision process more complex. This is because although more agents will expand the length of the input data, those data only need to pass through the same network structure. Moreover, we introduce the virtual lane to propose a vehicle selection method, which makes the current vehicle only care about the largest impact vehicles. In order to ensure the effectiveness of small-scale neural networks, we set an upper limit on the number of reference vehicles. All agents play the same role in cooperation with each other. In the stage of decision-making, the running states of the current vehicle and reference vehicles are combined into a set. Each action takes into account the states and actions of the reference vehicles. The vehicle selection method is explained in detail in the next section.

3.3. Largest Impact Vehicles Selection. The proposed method is built based on a distributed system, and all vehicles are regarded as the equal agents in the system. The process of selecting the largest impact vehicles is a prestep for information gathering. Based on the decoupling of identity and running information of the vehicles, how to select vehicles to obtain running information becomes an issue. There are two main types of vehicle collisions near intersections. One is the longitudinal collision in the lane, and the other is the lateral collision at the merging zone. The longitudinal collisions can be resolved by selecting adjacent vehicles in the same lane. To solve the lateral collision, this paper introduces the concept of the virtual lane.

For clear description, eight vehicles (L3, L4, R3, R4, U3, U4, D3, and D4) are chosen for illustration in Figure 2. At each time step, each vehicle will perform the vehicles selection. Here, vehicle L4 is taken as an example, so the lane from left to right is considered as the baseline of the virtual lane. Then vehicles U3, U4, D3, and D4 are projected onto the virtual lane. Because the up-to-down lane and the down-to-up lane have different conflict point with the left-to-right lane, the vehicles would have different offsets when projected. Since the right-to-left lane does not conflict with the left-to-right lane, R3 and R4 do not need to be projected. In Figure 2, the radius of the arc represents the distance to the conflict points. The dots with different colours represent the projected vehicle on the virtual lane, which is a black arrow pointing forward. In the scenario of this paper, there are four entering lanes, so four virtual lanes take effect. The projected virtual platoon is shown in Figure 3, and different colours and arrows indicate different direction.

In this paper, the selection of the largest impact vehicles can depend on space distance. When L4 is performing the vehicle selection, a star structure is obtained to express the relationship between L4 and other vehicles in Figure 4. Moreover, the length of the link between L4 and other dots means the relationship among vehicles. The shorter the

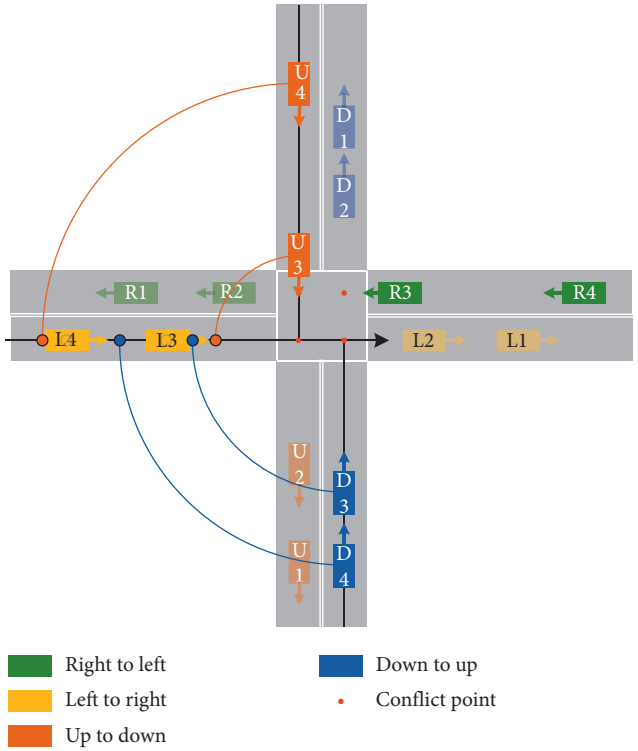

Figure 2: Virtual platoon projection. L3, L4, R3, R4, U3, U4, D3, and D4 are chosen for illustration, and L4 is taken as an example to perform the vehicles selection.

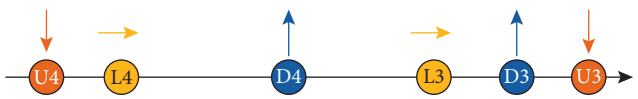

Figure 3: A projected virtual platoon.

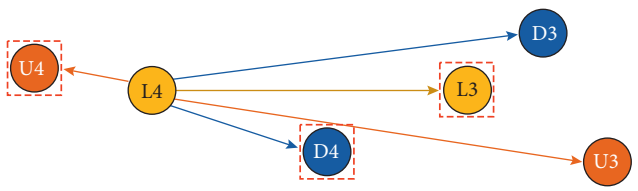

Figure 4: Star structure for L4. The shown dots represent candidate vehicles, and the red-dashed boxes select the vehicles to be considered.

length of the directed line, the larger the impact. Finally, among the linked dots to L4, several largest impact vehicles are selected with red-dashed boxes. The states, including positions and velocities, of these selected vehicles would be taken into the process of model training and inference.

With the largest impact vehicle selection, the proposed CoMADDPG can use a relatively simple network structure to handle a large number of agents.

3.4. Algorithm Architecture. This part illustrates how to apply CoMADDPG algorithm to this distributed control problem at intersection.

A learning algorithm for this distributed RL problem consists of two main parts: CoMADDPG trainer and executor. Figure 5 shows the overall architecture. The executor is applied to get updated policy from the trainer and uses it to collect experience from the simulation environment. Then, the trainer uses experience data from the executor to update policy network, which follows the DDGP model update process. Finally, the trainer distributes the newest model to the executors.

4. Experiments

4.1. Experimental Settings. In this section, CoMADDPG is trained and evaluated in the scenario of the intersection, which contains 4 different directions and allows vehicles to go straight without turning. Therefore, there are four conflicting points in the intersection, and each conflict point corresponds to a virtual lane. Furthermore, there are 4 types of vehicles, and each type possesses the same entry and exit. Vehicles appear at the beginning of each entering lane and follow a Poisson process with different vehicle density. Here, with the predefined arrival time and initial velocity, there is no need to set the distance between vehicles. Geometry and vehicle dynamics parameters are listed in Table 2. As for velocity and initial velocity, m/s is used in the experiment, but, in order to facilitate understanding, km/h is used in this paper. The central server on the intersection can collect experience from the vehicle, update the model, and distribute the newest model to the vehicles entering the lane. After receiving the newest model, the vehicle could determine the action with the model and send the produced experience to the central server. For results, the training processes of CoMADDPG are shown and illustrate our improvement in MADDPG.

4.2. Implementation Details. In CoMADDPG, there are two modules, actor and critic, for inference and training. Each module corresponds to a network structure without shared parameters, which are shown in Figure 6. In Figure 6(a), the actor module inputs state and outputs action, which contains three dense layers and two normalization layers, and chooses ReLU, presented in equation (10), as activation function. On the other hand, the critic module is used to evaluate the actions with Q value in a specific state. The structural difference between critic and actor lies in action set, which is concatenated to the processed state as an additional input. In CoMADDPG, action set contains the actions performed by the current vehicles and reference vehicles:

$$\mathrm{ReLU}(x) = \begin{cases} x, & x > 0, \\ 0, & x \le 0. \end{cases} \tag{10}$$

Furthermore, complete hyperparameters are listed in Table 3.

4.3. Results and Discussion. This section presents the performance of our algorithm at the intersection and analyzes

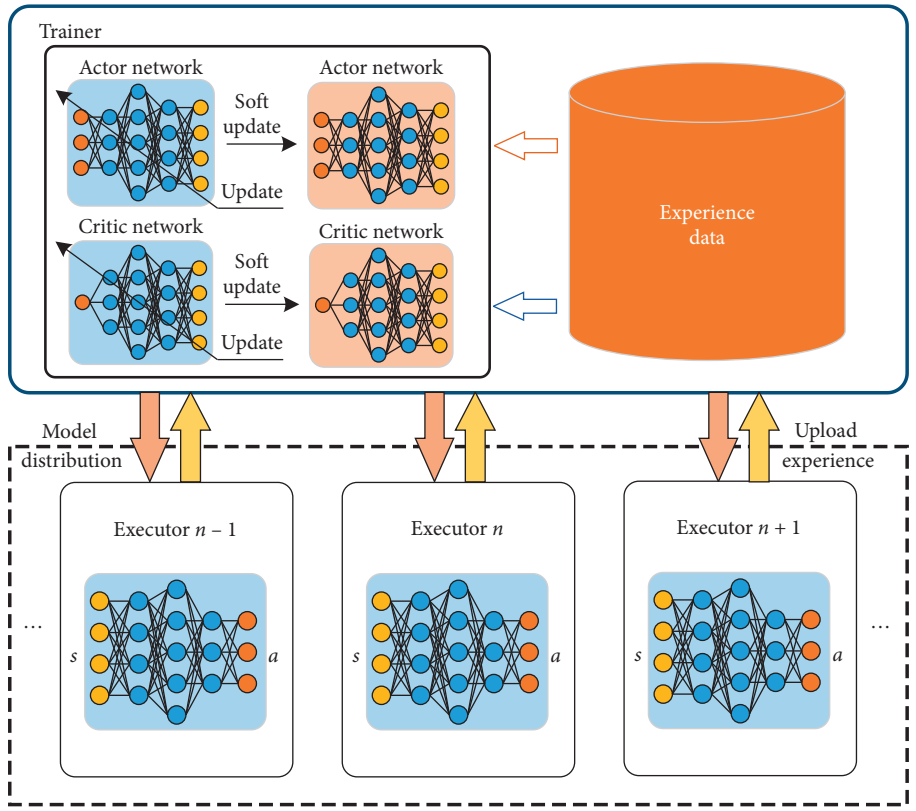

Figure 5: Overall architecture of the algorithm. The new executor, that is, vehicle, downloads the newest network from the centralized trainer. The centralized trainer updates the networks with the experience from executors.

Table 2: Geometry and vehicle dynamics parameters.

Parameter	Value
Lane length (m)	155
Vehicle size (m)	2
Velocity (km/h)	[18,50]
Initial velocity (km/h)	36
Acceleration (m/s^2)	[−3,3]

the empirical results. Firstly, a 3D view is used to visually show the changes in the position and velocity of vehicles as they pass through the intersection. Then, the process of training is observed to verify the convergence of the proposed CoMADDPG. Next, as one of the significant contributions, the virtual lane is compared with actual lanes under different parameters. Moreover, the average travel time is tested with different vehicle densities and different lane lengths. Finally, an optimization-based method is chosen for comparison, and the real-time performance is discussed.

In order to clearly understand the running state, Figure 7 illustrates the position and velocity profiles of approaching vehicles from 4 entrances of the intersection. In Figure 7(a), the approaching vehicles entered the merging zone of intersection orderly. Figure 7(b) presents approaching vehicles to adjust their velocities to achieve collision-free. The speed adjustment is too frequent, which will be optimized to improve the stability of the vehicle speed in our future work.

Firstly, to prove the ability of convergence, the experiment is designed to compare the training process between CoMADDPG and DDPG. From the perspective of reinforcement learning model training, the loss function, reward, and the statistics of the number of collisions are measured. CoMADDPG is the adaptation of MADDPG in the case of dynamic quantity agents, and DDPG is employed as our baseline, which only considers its actions without others' actions in the stage of centralized training. In terms of the mean reward and the loss function of the actor, there is no apparent difference between the two algorithms. In Figure 8(a), the loss functions of actor, which is the decision module in CoMADDPG and DDPG, cross down and tend to be flat, which means both algorithms can converge. During the most steps of the training, DDPG has lower loss than the proposed CoMADDPG, which is due to the simpler state and action input. However, the loss function is only an auxiliary indicator, and the number of collisions in Figure 8(d) is more valuable. In Figure 8(b), DDPG shows a higher loss than CoMADDPG. This is due to the insufficient information obtained by DDPG, which cannot form a stationary partial environment to perform reinforcement learning algorithm. In Figure 8(c), both algorithms have the same trend in the change of the mean reward. Concerning the cumulative number of collisions during training, CoMADDPG is much lower than DDPG in Figure 8(d). Because the CoMADDPG module can still explore during the training process, the cumulative collision curve of

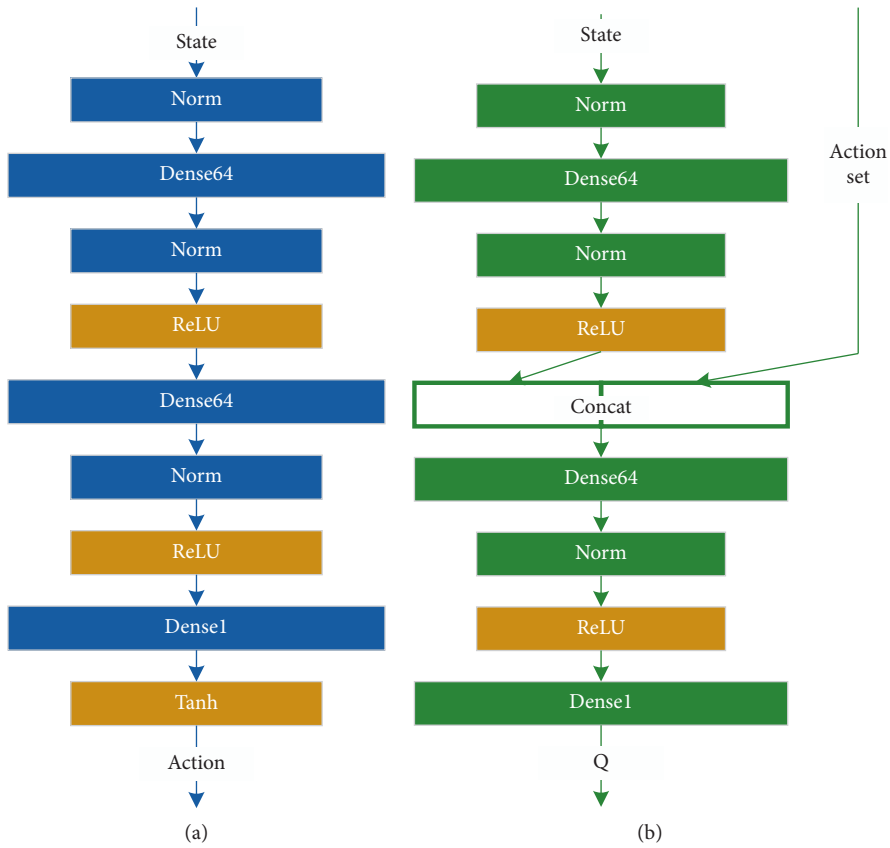

Figure 6: The network structures of actor and critic used in the proposed CoMADDPG. (a) Actor module. (b) Critic module.

Table 3: Hyperparameters of experiment.

Parameter	Value
Discounted factor γ	0.80
Minibatch size T	128
Soft update factor τ	0.998
Epoch U	300
Learning rate-actor	$10^{-4} \longrightarrow 0$
Learning rate-critic	$10^{-3} \longrightarrow 0$
Hidden layers number	2
Hidden units number	64
Optimizer	Adam

CoMADDPG will maintain an upward trend. In the evaluation stage, there is no collision in CoMADDPG, while DDPG still collides frequently.

Secondly, vehicle selection based on virtual lanes is an essential step in the transformation of reinforcement learning methods. To discuss the impact of different vehicle selection strategies on training, the experiment is designed to compare the virtual lane-based method with the actual lane-based method. Figure 9 exhibits the collision performance under virtual lane- and actual lane-based methods with the different number of vehicles considered. The virtual lane-based method is described in Section 3.3, and the actual lane-based method relies on the physical distance to select vehicles. In the experiment, the maximum number of the largest impact vehicles is set to 1, 3, and 6, respectively. Because one vehicle has conflicts with three of the four lanes, three is selected as a parameter. The influence of the front and rear vehicles is also considered here, so 6 is also used as a parameter for the experiment. 1 is set as a control parameter. As displayed in Figure 9, the actual lane-based method hardly achieves no collision, but as more vehicles are considered, the new number of collisions drops significantly. As for virtual lane-based method, it keeps fewer collisions. When considering that the number of vehicles reaches 6, the curve can reach a relatively horizontal state faster. This demonstrates the effectiveness of virtual lane-based vehicle selection in CoMADDPG to achieve no collision at unsignalized intersection.

Thirdly, to observe the influence of various lane lengths on average travel time, the experiment shows vehicles of different densities are running on lanes of different lengths and the average travel time is evaluated. Intuitively, a longer lane would allow vehicles to adjust their velocity to pass the intersection quickly. In Figure 10, there is no noticeable difference among different vehicles densities. The average travel time is proportional to the length of the lane, which means the proposed method can effectively deal with different lane lengths and vehicle densities. Moreover, low-density vehicles perform relatively poorly in long lanes. This is due to the sparse intervehicle spacing, resulting in insufficient coordination among vehicles.

Finally, the trained CoMADDPG is utilized for comparison with the optimization-based approach [18], which is

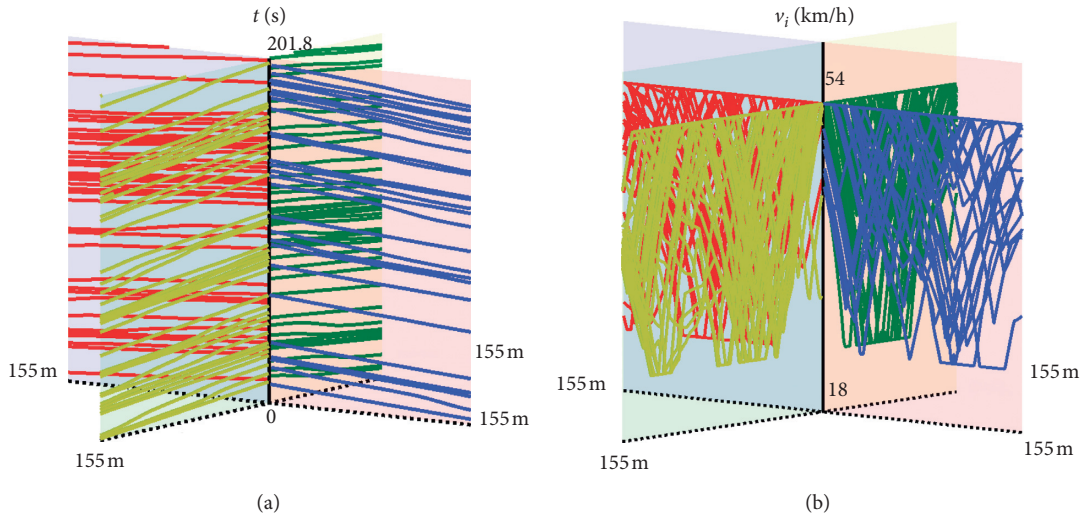

FIGURE 7: Vehicle status profiles in the area of intersection (different colours: vehicles from different entrances). (a) Position profiles. (b) Velocity profiles.

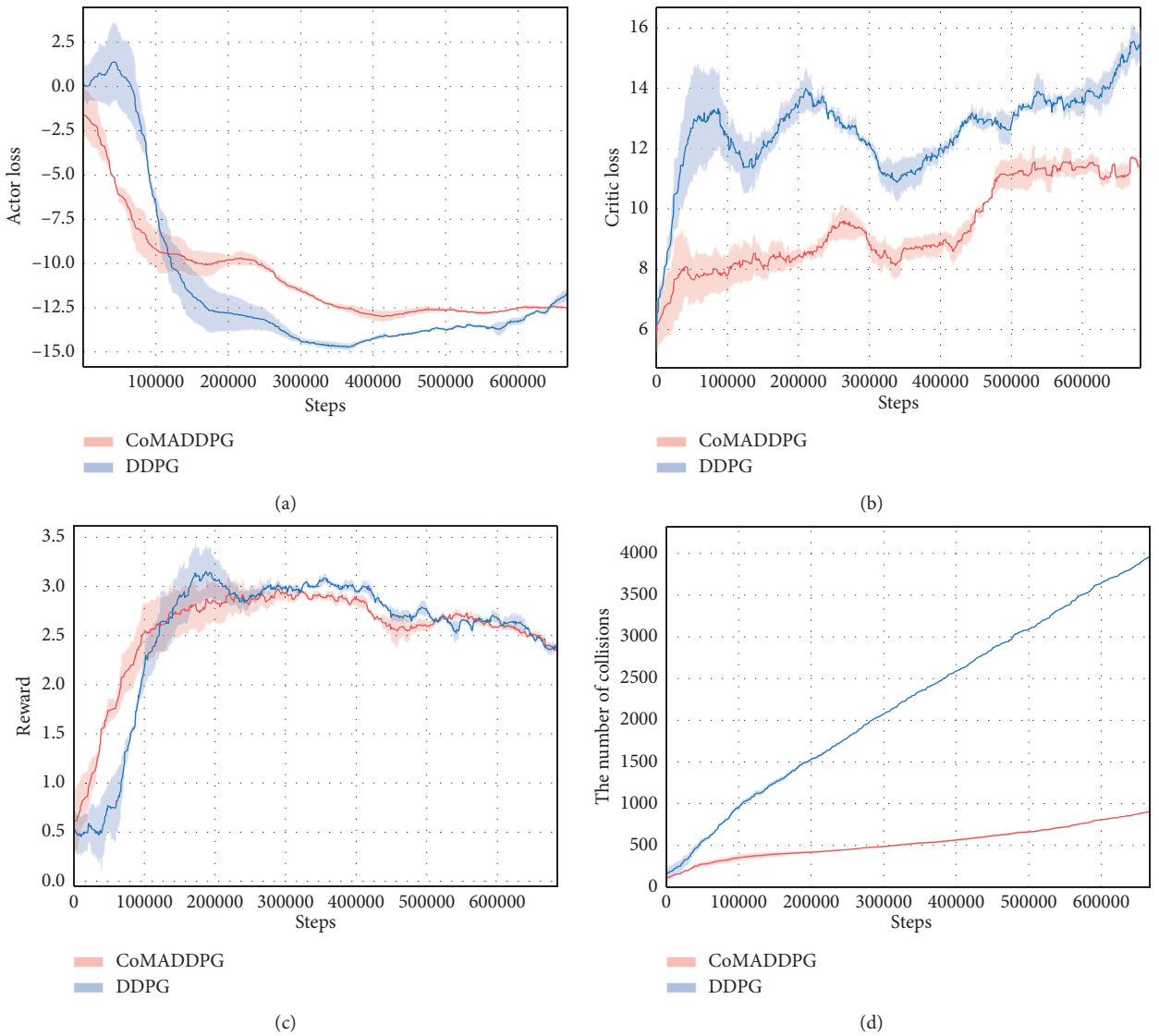

FIGURE 8: Comparison between CoMADDPG and DDPG during the training process. (a) The loss function of the actor. (b) The loss function of the critic. (c) The mean reward. (d) The cumulative number of collisions.

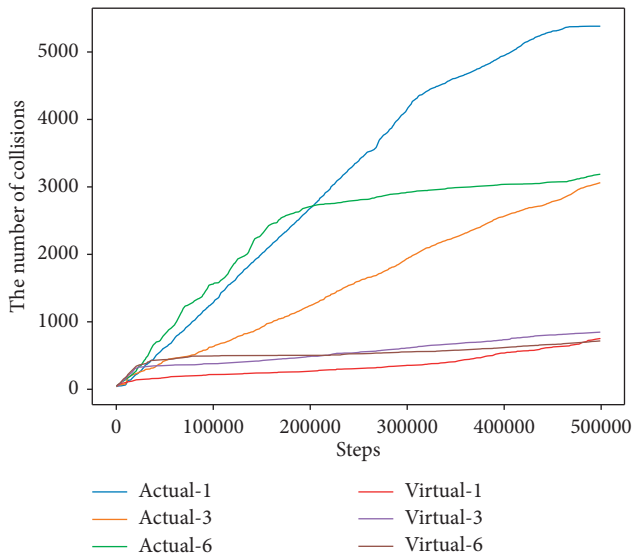

Figure 9: Comparison of cumulative collisions number using different vehicle selection configurations. "Actual" means actual lane-based vehicle selection. "Virtual" means virtual lane-based vehicle selection. The number indicates the maximum number of the largest impact vehicles.

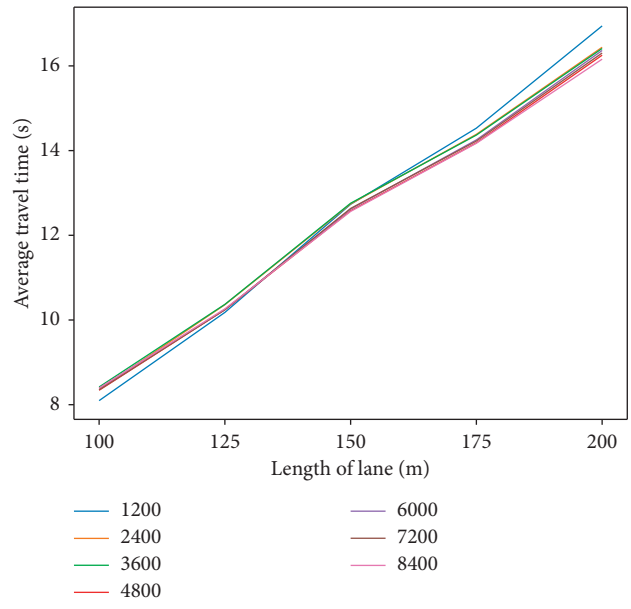

Figure 10: Comparison among various vehicles densities with different lengths of lane. 1200 to 8400 indicate different vehicles densities, whose unit is vehicles/hour.

in the same scenario. The average travel times in 3 different traffic volumes are shown in Table 4. It is observed that the proposed method increases the average travel times by 7.23%~39.28% in medium- and high-traffic volumes. Only in the case of low traffic flow the traffic efficiency is slightly lower than the optimization-based method. In addition, where more density is reached, the optimization-based method is no longer applicable, and the proposed CoMADDPG still works normally. Note that the results are only compared in the speed adjustment area, and the optimization-based method has an extra distance of 100 meters for observation and optimizing the speed. In brief, the results revealed that the proposed CoMADDPG has a slower travel time than the baseline, which can avoid vehicles congested in the entering lane of the intersection.

Table 4: Comparison of average travel times.

Total traffic volume (vehicles/hour)		1200	2400	3600	4800	6000
Travel time (s)	Ref. [18]	12.145	13.773	21.366	—	—
	Proposed	12.385	12.777	12.974	12.580	12.618
Improvement (%)		−1.98	7.23	39.28	—	—

The process of decision is evaluated on a laptop with an Intel CPU (i7-8565U @ 1.8 GHz, 1.9 GHz), 16 GB RAM, and NVIDIA GeForce MX250. The average running time is 0.36 ms. According to the 3 GPP standard [32], TX rate for cooperative collision avoidance between UEs supporting V2X applications is 100 messages/s, which means the message sending interval is 10 ms. Obviously, the proposed CoMADDPG meets the requirement of real time, and there is ample time to support function expansion.

5. Conclusion

In this paper, a multiagent reinforcement learning method is employed to solve distributed cooperation for connected and automated vehicles at unsignalized intersection, which has been regarded as a challenging problem of cooperation among dynamic quantity vehicles. Vehicle selection is incorporated into MADDPG to propose CoMADDPG, which makes it adapt to the dynamic quantity vehicles at the unsignalized intersection. Moreover, the virtual lane-based method enhanced intervehicle cooperation for collision avoidance. A typical 4-direction intersection containing four different types of vehicle is studied. The simulation results demonstrate that the proposed method is efficient. Compared with the existing optimization-based method, up to 39.28% improvement implies that CoMADDPG is worthwhile to handle distributed vehicle control safely and efficiently at unsignalized intersection.

In order to simplify the problem, this paper only studies the case where the single lane only goes straight. The proposed CoMADDPG can also solve the situation of multiple lanes and multiple directions. For multiple lanes, projecting more actual lanes onto virtual lanes requires an appropriate increase in the number of vehicles selected. For multiple directions, the piecewise projection of the collision points on the curve needs to be addressed.

In this paper, traffic safety and efficiency optimizations of passing through unsignalized intersection are researched, but vehicle stability is not considered. The introduction of vehicle stability would limit the exploration ability of the RL algorithm and may fall into the local optimum, which cannot achieve the highest vehicle passing efficiency. In future work, vehicle stability will guide our research as an essential topic.

Data Availability

No data were used to support this study. What we adopted is reinforcement learning, and data are generated from the environment we made in our simulation.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the National Key R&D Program of China (2016YFB0100902).

References

[1] S. Zeadally, J. Guerrero, and J. Contreras, "A tutorial survey on vehicle-to-vehicle communications," *Telecommunication Systems*, vol. 73, no. 3, pp. 469–489, 2020.

[2] L. Chen and C. Englund, "Cooperative intersection management: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 570–586, 2015.

[3] 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.

[4] D. Kim and O. Jeong, "Cooperative traffic signal control with traffic flow prediction in multi–intersection," *Sensors*, vol. 20, no. 1, p. 137, 2020.

[5] 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 Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, 2019.

[6] D. Yu, "Signal timing optimization based on fuzzy compromise programming for isolated signalized intersection," *Mathematical Problems in Engineering*, vol. 2016, Article ID 1682394, 12 pages, 2016.

[7] Y. Bai, J. Li, T. Li, L. Yang, and C. Lyu, "Traffic signal coordination for tramlines with passive priority strategy," *Mathematical Problems in Engineering*, vol. 2018, p. 14, 2018.

[8] Y. Xu, D. Li, and Y. Xi, "A game-based adaptive traffic signal control policy using the vehicle to infrastructure (V2I)," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 9425–9437, 2019.

[9] Y. Lu, X. Xu, C. Ding, and G. Lu, "A speed control method at successive signalized intersections under connected vehicles environment," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 3, pp. 117–128, 2019.

[10] B. Xu, "Cooperative method of traffic signal optimization and speed control of connected vehicles at isolated intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1390–1403, 2018.

[11] S. Boudaakat, R. Ahmed, and B. Omar, "Smart traffic control system for decreasing traffic congestion," in *Proceedings of International Conference on Systems of Collaboration Big Data, Internet of Things & Security (SysCoBIoTS)*, December 2019.

[12] Md 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 Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, 2019.

[13] S. Gupta, M. Vasardani, and S. Winter, "Negotiation between vehicles and pedestrians for the right of way at intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 888–899, 2018.

[14] 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 Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1956–1967, 2016.

[15] Y. Guan, Y. Ren, S. E. Li et al., "Centralized Conflict–free Cooperation for Connected and Automated Vehicles at Intersections by Proximal Policy Optimization," 2019, https://arxiv.org/abs/1912.08410.

[16] 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 of Things Journal*, vol. 6, no. 6, pp. 10408–10420, 2019.

[17] B. Xu, S. E. Li, Y. Bian et al., "Distributed conflict-free cooperation for multiple connected vehicles at unsignalized intersections," *Transportation Research Part C: Emerging Technologies*, vol. 93, pp. 322–334, 2018.

[18] Y. Bian, S. E. Li, W. Ren, J. Wang, K. Li, and H. Liu, "Cooperation of multiple connected vehicles at unsignalized intersections: distributed observation, optimization, and control," *IEEE Transactions on Industrial Electronics*1 pages, 2019.

[19] A. Hadjigeorgiou and S. Timotheou, "Optimizing the trade–off between fuel consumption and travel time in an unsignalized autonomous intersection crossing," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, October 2019.

[20] F. Belkhouche, "Collaboration and optimal conflict resolution at an unsignalized intersection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2301–2312, 2018.

[21] Y.-C. Hsu, S. Gopalswamy, S. Saripalli, and D. A. Shell, "An MDP model of vehicle–pedestrian interaction at an unsignalized intersection," in *Proceedings of IEEE 88th Vehicular Technology Conference (VTC–Fall)*, IEEE, Chicago, IL, USA, Augest 2018.

[22] J. Wang, X. Zhao, and G. Yin, "Multi–objective optimal cooperative driving for connected and automated vehicles at non–signalised intersection," *IET Intelligent Transport Systems*, vol. 13, no. 1, pp. 79–89, 2018.

[23] H. Yang and K. Oguchi, "Intelligent vehicle control at signal–free intersection under mixed connected environment," *IET Intelligent Transport Systems*, vol. 14, no. 2, pp. 82–90, 2019, https://search.crossref.org/?q=Intelligent+vehicle+control+at+signal%E2%80%93free+intersection+under+mixed+connected+environment.+IET+Intelligent+Transport+Systems+14.2+%282019+%29%3A+.

[24] M. Kwon, J. Lee, and H. Park, "Intelligent IoT connectivity: deep reinforcement learning approach," *IEEE Sensors Journal*, vol. 20, no. 5, pp. 2782–2791, 2020.

[25] Y. Zhang, L. Guo, B. Gao, T. Qu, and H. Chen, "Deterministic promotion reinforcement learning applied to longitudinal velocity control for automated vehicles," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 338–348, 2019.

[26] V. Mnih, V. Kavukcuoglu, D. Silver et al., "Human–level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[27] T. P. Lillicrap, "Continuous control with deep reinforcement learning," 2015, https://arxiv.org/abs/1509.02971.

[28] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-

competitive environments," in *Proceedings of Advances in Neural Information Processing Systems. 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, June 2017.

[29] X. Liang, X. Du, G. Wang, and Z. Han, "A deep reinforcement learning network for traffic light cycle control," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1243–1253, 2019.

[30] M. Zhou, Yu Yang, and X. Qu, "Development of an efficient driving strategy for connected and automated vehicles at signalized intersections: a reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, 2019.

[31] J. Lee, J. Chung, and K. Sohn, "Reinforcement learning for joint control of traffic signals in a transportation network," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, 2019.

[32] 3GPP TS22.186, "Enhancement of 3GPP support for 5G V2X services: stage 1," 2019.