Cooperative Control Model Using Reinforcement Learning for Connected and Automated Vehicles and Traffic Signal Light at Signalized Intersections

Shan Fang, Lan Yang, Wen-long Shang, Xiangmo Zhao, Fengze Li, Washington Ochieng

***Abstract*—Effectively leveraging data and domain knowledge remains a significant challenge in controlling the Internet of Unmanned Agent (IUA). This paper proposes a novel multi-agent deep reinforcement learning-based cooperative control model called MARL-CTV to efficiently control two key IUA agents: Connected and Automated Vehicle (CAV) and controllable Traffic Signal Light (TSL). The CAV agents are controlled by the deep deterministic policy gradient (DDPG) algorithm, and the traffic signal light agent is controlled by a dueling double deep Q-network (D3QN). To reduce the control burden and ensure the cumulative reward converges, the actor and critic networks are pre-trained by the expert dataset, and the expert dataset initializes the experience replay buffer of DDPG. This dataset is generated by multiple velocity profiles derived from a genetic algorithm (GA) based on various random initial states of CAVs. Numerical experiments conducted using a joint simulation platform composed of SUMO and CARLA and real-world data from CitySim demonstrate the effectiveness of MARL-CTV. Specifically, when the market penetration rate (MPR) of CAV is 35%, MARL-CTV enables most CAVs to pass through the signalized intersection without stop-and-go behavior, reducing average travel time by 24.2%, fuel consumption by 22.7%, and the traffic conflicts by 68.3%.**

***Index Terms*—Connected and Automated (CAV), Mixed Traffic Flow, Signalized Intersection, Cooperative Control, Deep Reinforcement Learning**

NOMENCLATURE

Abbreviations

IUA	Internet of Unmanned Agents
CAV	Connected and Automated Vehicle
SPaT	Signal Phase and Timing
GLOSA	Green Light Optimal Speed Advisory
AI	Artificial Intelligence
DQN	Deep Q-Network
DDPG	Deep Deterministic policy gradient
TD3	Twin Delayed Deep Deterministic
SAC	Soft Actor-Critic
HV	Human-driven Vehicle
DRL	Deep Reinforcement Learning
MILP	Mixed-integer Linear Programming
MPC	Model Predictive Control
PPO	proximal policy optimization
D3QN	dueling double deep Q-network
NEMA	National Electrical Manufacturers Association
DNN	Deep Neural Network
ReLU	Rectified Linear Unit
O-U	Ornstein-Uhlenbeck
MPR	Market Penetration Rate
AFC	Average Fuel Consumption
NTC	Number of Traffic Conflicts
AWT	Average Waiting Time
ATT	Average Travel Time
NVS	Number of Vehicle Stops

Symbols

Q	Queue length
T	Simulation duration
$J_1^{i,t}$	Control cost of the *i*-th CAV at the *t*-th time step
j_2^t	Control cost of the signal light at the *t*-th time step
c_v	Control time
I	Total number of CAVs
ϕ_g	Green light timing of one cycle
G_{min}	The shortest green light timing
G_{max}	The longest green light timing
$\boldsymbol{S}$	The stage space of the agent
$\boldsymbol{A}$	The action space of the agent
P	The transition function of the agent
R	The reward function of the agent
γ	The discount factor
$\boldsymbol{s_t^v}$	The state vector of the CAV
p_t^v	The position of the CAV at the t-th time step
v_t^v	The velocity of the CAV at the t-th time step
a_t^v	The acceleration of the CAV at the t-th time step
t_d	The vehicle queue dissipation time
T_G	The remaining green timing
T_R	The remaining red timing
μ	The deterministic policy gradient
θ^μ	The parameter of actor network
θ^Q	The parameter of critic network
θ_e^μ	The parameter of the pre-trained actor network
θ_e^Q	The parameter of the pre-trained critic network
y_i	The output of the target critic network
N	The Batch size
r_t^v	The return calculated by DDPG
r_t^a	fuel consumption at the t-th time step
f_{min}	The minimum fuel consumption
f_{max}	The maximum fuel consumption
r_t^b	The traffic efficiency reward term
r_t^c	The passenger comfort reward term
r_t^d	The velocity reward term
r_t^e	The collision penalty term
r_t^f	The crash penalty term
t_s	Time space headway
r_l	The crash penalty term
$\boldsymbol{D_1}$	The training dataset of the actor network

This research has been supported by the National Natural Science Foundation of China (No.52472446 and No.52202377), Beijing Natural Science Foundation (No. L211027 and No.9232003), the Shaanxi Province Key Research and Development Program (2022GY-300). *(Co-corresponding authors: Wen-Long Shang, Lan Yang)*

Shan Fang, Lan Yang, Xiangmo Zhao and Fengze Li are with the School of Information Engineering, Chang'an University, Xi'an 710064, China (Email: fang6100146@gmail.com; lanyang@chd.edu.cn; xmzhao@chd.end.cn; jasonleee539@gmail.com)

Wen-Long Shang is with the department of Civil and Environmental Engineering, Imperial College London SW7. (Email: wenlong.shang12@imperial.ac.uk.)

Washington Ochieng is with Centre for Transport Studies, Imperial College London, SW7 2AZ London, UK. (Email: w.ochieng@imperial.ac.uk)

$\boldsymbol{D_2}$	The label dataset of the actor network
$\boldsymbol{D_3}$	The training dataset of the critic network
$\boldsymbol{D_4}$	The label dataset of the critic network
$\boldsymbol{D_e}$	The expert experience dataset
U	The uniform distribution
v_{min}	The minimum velocity of vehicle
v_{max}	The maximum velocity of vehicle
a_{min}	The minimum acceleration of vehicle
a_{max}	The maximum acceleration of vehicle
a_t^e	The acceleration generated by genetic algorithm
r^e	The return calculated by genetic algorithm
$\boldsymbol{s_t^l}$	State space of the traffic signal light
X	Number of lanes of one direction
Y	Number of cells of one lane
c_l	Cell length

I. INTRODUCTION

The rapid growth in traffic demand has not only exacerbated congestion but also increased fuel consumption and accident risks in urban areas [1]. According to the 2023 Urban Mobility Report, the cost of congestion surged to $224 billion in 2022 [2]. Signalized intersections, as bottlenecks in urban traffic flow, are particularly vulnerable to these challenges. Enhancing efficiency, fuel consumption, safety, and passenger comfort remains a critical task [4]-[4]. With the development of intelligent transportation systems, two representative Internet of Unmanned Agents (IUAs), including connected and automated vehicle (CAV) and controllable traffic signal light have been widely used to improve efficiency at signalized intersections. Therefore, two main strategies have emerged to improve the capacity of signalized intersections [5]. The first strategy involves controlling CAVs or traffic signals independently, while the second focuses on the joint control of both. Given the extensive research on the first strategy and space limitations, this paper will briefly review the independent control strategy and delve into the joint control approach.

A. CAV Control and Traffic Signal Control

Leveraging the high-precision vehicle state data from CAVs and signal phase and timing (SPaT) data from roadside equipment, various CAV control methods have been developed to manage CAVs upstream of signalized intersections. Notable methods include the Green Light Optimal Speed Advisory (GLOSA) algorithm [6] and the CAV trajectory planning algorithm based on trigonometric functions [7]. Subsequent studies have built upon these methods, incorporating multiple impact factors such as mixed traffic flow [8]-[9], safety [10], turning maneuver [11], and time-variable actuated signal timing prediction [12]. Thanks to cutting-edge artificial intelligence (AI) technology, particularly deep reinforcement learning (DRL), has significantly advanced CAV control. Initial studies typically utilized the deep Q-Network (DQN) algorithm for CAV control [13]-[14], but a significant limitation of DQN-based CAV control is that the acceleration cannot be accurately represented with a finite set of discrete values. To address this problem, the deep deterministic policy gradient (DDPG) algorithm [15], designed for continuous action spaces, has been applied to enhance CAV control accuracy [16]-[17]. Subsequently, the twin delayed deep deterministic (TD3) policy gradient algorithm [18] and the soft actor-critic (SAC) algorithm [19] have also been employed in CAV control, demonstrating their effectiveness in this domain [20]-[23]. To help the single CAV surrounded by the HVs pass the intersection, a hybrid policy based on reinforcement learning and car-following is designed [24]. It should be noted that the above-mentioned literature paid more attention to the longitudinal control strategy of CAVs. However, there may be one more incoming lane that has the same direction. Intersections located in the arterial corridor have three straight lanes. Therefore, it is necessary to make full use of spatiotemporal resources to promote traffic efficiency. For example, the lane-changing of CAVs and the stochasticity of human-driven vehicles (HVs) are considered in the literature[25]-[26]. An eco-driving framework was proposed which integrated the longitudinal car-following control and lateral decisions [27]. Despite these advancements, both approaches rely on fixed traffic light signal timings, highlighting a gap in adaptive signal control strategies that could further optimize traffic flow.

Traffic signal control methods can be categorized into fixed-time control, actuated control, and adaptive control, with adaptive control being the focus of recent studies. Numerous adaptive control approaches have been proposed to enhance lane capacity and travel time based on probe vehicle trajectories [28]-[29]. Similar to CAV control, deep reinforcement learning (DRL) has been extensively applied in traffic signal control. The design of state, action, and reward functions is crucial for the effective implementation of DRL algorithms. Different studies have varied considerations for setting the state, including queue length [30], waiting time [31]-[32], and phase [33]. For the action space, there are two main approaches: the first approach involves the DRL algorithm directly providing the appropriate phase and timing without a cyclic manner [34]-[37], while the second approach allows the DRL algorithm to decide whether to maintain or skip the current phase from the cyclic traffic phases[38]. The reward function is also a critical component of the DRL algorithm, utilizing various metrics to assess the quality of actions at each time step based on the state. Metrics used in reward functions include waiting time [32], queue length [33], and throughput [38]. Recently, some research paid more attention to the traffic signal control under the network rather than the isolated signalized intersection. For instance, the multi-agent reinforcement learning method [39]-[40] and graph neural network [41]-[42] are widely used. In addition, the heterogeneous traffic network composed of signalized and non-signalized intersections was also considered [43]. However, a significant limitation of current traffic light control approaches is that they rarely fully utilized controllable CAVs. Instead, these methods relied on the high-quality vehicle dynamic data provided by CAVs as inputs. Moreover, cooperative control strategies that integrate both CAVs and traffic signals are more effective means for enhancing traffic efficiency, and it is possible to achieve greater improvements in overall traffic flow and safety.

B. Cooperative Control for CAV and Traffic Light Signal

In recent years, the intelligent transportation systems community has increasingly focused on cooperative control at signalized intersections. One of the earliest studies incorporating integrated optimization was proposed in 2014 [44]. Since then, the cooperative control problem has typically been formulated as a bi-level optimization problem, aiming to maximize efficiency and minimize waiting time [45]-[46]. In this framework, the lower level optimizes the trajectory of CAVs to enhance comfort [47]-[49], while the upper level optimizes signal timing or phase [50]-[51]. However, some studies have assumed that all vehicles are controllable CAVs, which limits the applicability and robustness of these methods [46]-[48]. Recent studies have recognized the importance of mixed-traffic flow and the need to account for it. For instance, Yang et al. [52].proposed a cooperative driving framework that employs both centralized and distributed control methods Ying et al. [53] developed an infrastructure-assisted cooperative control model to optimize traffic signal parameters and CAV trajectories, using inverse reinforcement learning, a representative model of imitation learning, for CAV control. Jiang et al. [54] created a joint optimization model to improve CAV-dedicated lane allocation and signal timing. Methodologically, some studies have employed large-scale nonlinear programming [49], mixed-integer linear programming (MILP) [50]-[53], dynamic programming [54], and model predictive control (MPC) [55]. Although MILP can provide global optimal solutions for cooperative control, it is computationally intensive and highly sensitive to problem formulation. Compared to the model-driven approach represented by MILP, the data-driven approach using DRL may better address cooperative control problems in complex traffic scenarios. Guo et al. [56] proposed a DRL-based cooperative control method, using proximal policy optimization (PPO) to help CAVs and traffic signals learn the optimal policy. However, to the best of our knowledge, fewer studies jointly optimize traffic signals and CAVs in mixed-traffic flow due to the difficulty in designing an offline DRL-based control method for the two different agent types [57].

To fill the gaps in the above literature, this paper integrates the CAV control and the traffic signal light control into a cooperative model, defining two types of agents: the CAV agent and the traffic signal light agent. The CAV agent is controlled by the DDPG algorithm, and the traffic signal light agent is controlled by the dueling double deep Q-network (D3QN). The main contributions of this paper are concluded as follows:

1) A cooperative control model using multi-agent deep reinforcement learning for the traffic signal light and CAV (MARL-CTV) is proposed, aiming to improve the multi-objective in terms of efficiency, safety, comfort, and fuel consumption.

2) An expert dataset is constructed to pre-train the actor and critic networks and initialize the experience replay buffer, which can help reduce the control burden and accelerate the cumulative reward convergence of the DRL model.

3) To comprehensively evaluate the performance and robustness of the MRAL-CTV model, two case studies are conducted using the joint simulation platform of SUMO and CARLA. The first case study tests the performance of the MARL-CTV model under high traffic flow conditions, and the second case study validates the robustness of the MARL-CTV model using the real-world signalized intersection data provided by the CitySim dataset.

II. PROBLEM DESCRIPTION

Figure 1 illustrates a typical isolated signalized intersection, where green vehicles represent CAVs, and gray vehicles represent HVs. The area is colored light blue is the control zone. According to the signal timing standards set by the National Electrical Manufacturers Association (NEMA), the traffic signal light is designed with the dual-ring barrier structure.

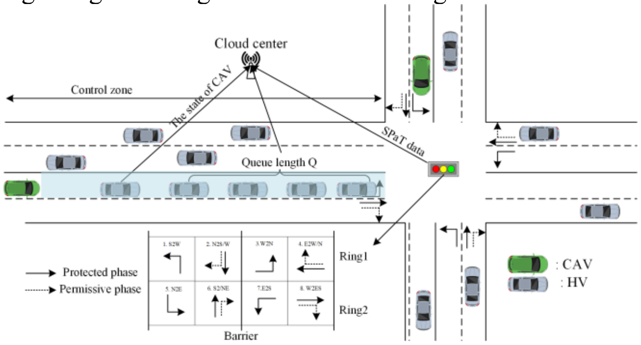


Figure 1. A typical signalized intersection with four arms.

It can be seen from Figure 1 that when the traffic signal is red, both vehicles need to decelerate to idle, forming a queue with length Q. Therefore, the CAVs and traffic signal light should be controlled simultaneously to alleviate congestion and enhance efficiency. The main objective of this optimization model is to minimize total travel time for all vehicles through the cooperative control of CAVs and traffic signal, and it can be formulated as follows:

$$min(\sum_i^I \sum_t^{c_v} J_1^{i,t} + \sum_t^T j_2^t) \tag{1}$$

where the $J_1^{i,t}$ is the control cost of the *i*-th CAV at the *t*-th time step. The c_v is the total control time at which a CAV passes the upstream control zone at the signalized intersection and may differ for each CAV. I is the total number of CAVs. j_2^t is the control cost of the traffic signal light at the *t*-th time step, and T represents the simulation duration. The details of $J_1^{i,t}$ and j_2^t will be illustrated as the section III. In addition, (1) also needs to meet the following signal timing constraints:

$$G_{min} < \phi_g < G_{max} \tag{2}$$

where ϕ_g is the green light timing of one cycle. G_{min} and G_{max} represent the shortest green light timing and the longest green light timing in each signal cycle, respectively. This constraint requires that the green light timing for each phase stay within the G_{min} and G_{max}, preventing a single phase from having an excessively long green period that could reduce available green time for other phases.

III. METHODOLOGY

The proposed MARL-CTV model is illustrated in Figure 2. The environment is established using SUMO [58], and the instantaneous fuel consumption of all vehicles is calculated using the default vehicle emission model HBEFA3/PC_G_EU4 of SUMO. Additionally, the traffic signal light agent uses the discrete position points of all vehicles in the mixed traffic flow as the state space variable. The trajectories of HVs are simulated by the dynamic transformation car-following (DTCF) [59] model. The reward function is designed based on the accumulated total waiting time of vehicles at the signalized intersection, and the action variable space of the traffic light agent is to increase or decrease the green light timing by 0.1

seconds. The CAV agent considers the position, velocity, acceleration of the CAV, and the remaining signal timing as the major state, with acceleration as the action.

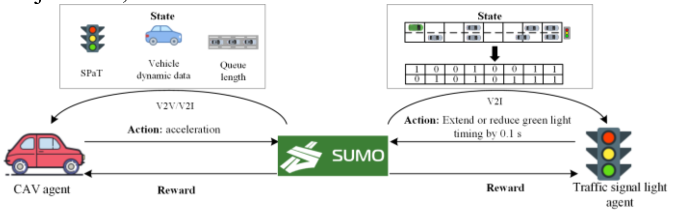


Figure 2. The framework of the MARL-CTV model.

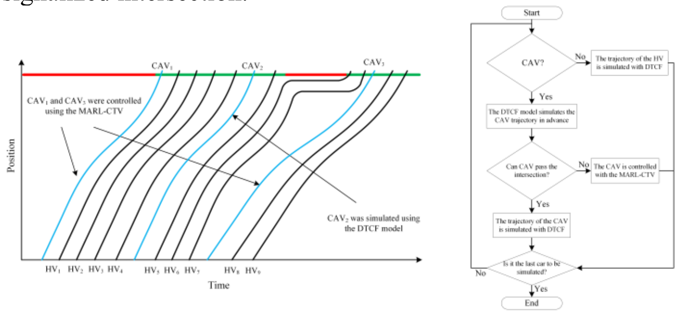
It should be noted that there are numerous CAVs on the signalized intersection, and each agent needs to be initialized and controlled. Furthermore, random vehicle arrivals and initial states can significantly affect the efficiency of multi-agent reinforcement learning, potentially causing non-convergence of cumulative rewards. To address these challenges, two strategies are developed for helping agents discover an optimal policy. Firstly, the CAVs and the traffic signal light are trained separately. More specifically, the CAVs are trained first, and then the traffic signal is trained. This approach allows the CAVs to avoid being re-trained from scratch, thereby reducing the overall training time [60]-[61]. Secondly, the DTCF model is employed to pre-simulate the trajectories of CAVs to determine which ones can follow HVs through the signalized intersection without stop-and-go behavior. As shown in Figure 3, an example is provided below to illustrate the second strategy.

(1) The MARL-CTV method is used to control CAV₁ to pass through the signalized intersection.

(2) The instantaneous velocity and acceleration of CAV₁, along with the state data (collected by loop detectors) for the following HVs entering the upstream control area, serve as inputs to the DTCF model to simulate the trajectories of HV₁-HV₄.

(3) When CAV₂ enters the upstream control zone, the DTCF model simulates its trajectory following HV4 in advance. If the trajectory simulated by the DTCF model indicates that CAV₂ can travel the signalized intersection, CAV₂ is treated as the HV.

(4) Steps 1-3 are repeated: the DTCF model predicts that CAV₃ following HV₇ requires a stop-and-go behavior, so the MARL-CTV model will take over to control CAV3 through the signalized intersection.



(a) Space-time trajectory diagram (b) Flow chart

Figure 3. The diagram of vehicle platoon split in the mixed traffic flow.

A. CAV Agent

The process by which a CAV interacts with the traffic environment and performs action can be described by the Markov decision process $M = \{\boldsymbol{S}, \boldsymbol{A}, P, R, \gamma\}$. Where $\boldsymbol{S}$ is the state space. $\boldsymbol{s}_t^v$ is denoted the state vector of the CAV at the t-th time step, $\boldsymbol{s}_t^v \in \boldsymbol{S}$, and it can be formulated as the (3):

$$\boldsymbol{s}_t^v = \{p_t^v, v_t^v, a_t^v, c_v, t_s, Q, t_d\} \tag{3}$$

where $p_t^v, v_t^v, a_t^v, c_t^v, t_s, Q, t_d$ represent the position, velocity, acceleration, total control time, time gap, vehicle queue length and vehicle queue dissipation time of CAV at the t-th time step, respectively. The vehicle queue length Q and the t_d is calculated by the reference [62]. The action space $\boldsymbol{A}$ is composed of the acceleration values between $-3m/s^2$ and $3m/s^2$. The a_t^v is denoted the action of the CAV at the t-th time step and $a_t^v \in \boldsymbol{A}$. The P is the transition function that represents the probability of the CAV transitioning from $\boldsymbol{s}_t^v$ to $\boldsymbol{s}_{t+1}^v$, and it can be represented as $P(\boldsymbol{s}_{t+1}^v | \boldsymbol{s}_t^v, a_t^v\}$. The R is the reward function $R(\boldsymbol{s}_t^v, a_t, \boldsymbol{s}_{t+1})$, which can calculate the reward value when the CAV taking action a_t based on the state $\boldsymbol{s}_t^v$ to the state $\boldsymbol{s}_{t+1}^v$. γ is the discount factor.

Due to the different CAVs having different initial states, vanilla DDPG may fail to generate an optimal trajectory for every CAV. To address this, an enhanced DDPG (E-DDPG) method has been developed, and the framework of the proposed E-DDPG is shown in Figure 4.

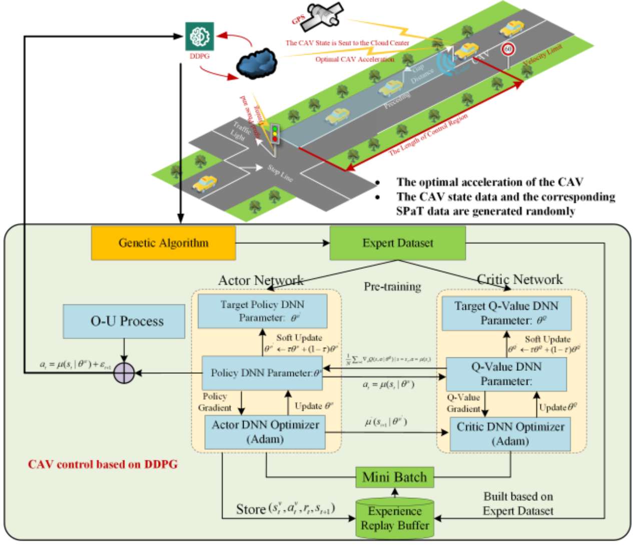

GPS
DDPG
The CAV State is Sent to the Cloud Center
Optimal CAV Acceleration
Gap Distance
Radar
CAV
Velocity Limit
60
Preceding
Traffic Light
Signal Timing
SPaT data
Stop Line
The Length of Control Region
The optimal acceleration of the CAV
The CAV state data and the corresponding SPaT data are generated randomly
Genetic Algorithm
Expert Dataset
Actor Network
Pre-training
Critic Network
O-U Process
Target Policy DNN Parameter: $\theta^{\mu'}$
Soft Update $\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
Target Q-Value DNN Parameter: $\theta^{Q'}$
Soft Update $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$
$a_t = \mu(s_t | \theta^{\mu}) + \varepsilon_{t+1}$
Policy DNN Parameter: θ^{μ}
$\frac{1}{N}\sum_i \nabla_a Q(s,a|\theta^Q)|_{s=s_i, a=\mu(s_i)}$
Q-Value DNN Parameter:
$a_t = \mu(s_t | \theta^{\mu})$
Policy Gradient
Update θ^{μ}
Q-Value Gradient
Update θ^{Q}
Actor DNN Optimizer (Adam)
$\mu'(s_{t+1} | \theta^{\mu'})$
Critic DNN Optimizer (Adam)
CAV control based on DDPG
Mini Batch
Store $(s_t^v, a_t^v, r_t, s_{t+1})$
Experience Replay Buffer
Built based on Expert Dataset

Figure 4. The framework of the E-DDPG.

It can be seen from Figure 4 that the genetic algorithm (GA) calculates the acceleration based on different initial states of CAVs and SPaT data and constructs the expert dataset. The expert dataset is used to pre-train the actor and critic networks of the vanilla DDPG and initialize the experience replay buffer. The CAV agent can then load the pre-trained actor-critic networks and continue training in a multi-agent environment. The details of the E-DDPG approach are provided below.

(1) Actor-Critic Architecture

The actor-critic architecture is the most significant part of the E-DDPG, and it combines the advantages of the policy-based and value-based methods. The actor network means that the deterministic policy gradient method is deployed based on the deep neural network (DNN). Given that the CAV enters the control zone at the t-th time step, the actor network can be formulated with (4):

$$a_t^v = \mu(\boldsymbol{s}_t^v | \theta^{\mu}) \tag{4}$$

where the μ is the actor network represented as DNN, and θ^{μ} is the parameter of the actor network. The structure of the actor

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3541881

network is shown in Figure 5 (a). The batch normalization is introduced to improve the stability of the neural network training stage and to mitigate the negative effects of the internal covariate shifts. Additionally, the Leaky ReLU (rectified linear unit) is selected as the activation function between two fully connected layers to address the problems of dying neurons issue and gradient vanishing.

The critic network means that the state-action value function is built based on the DNN. It can be formulated with (5):

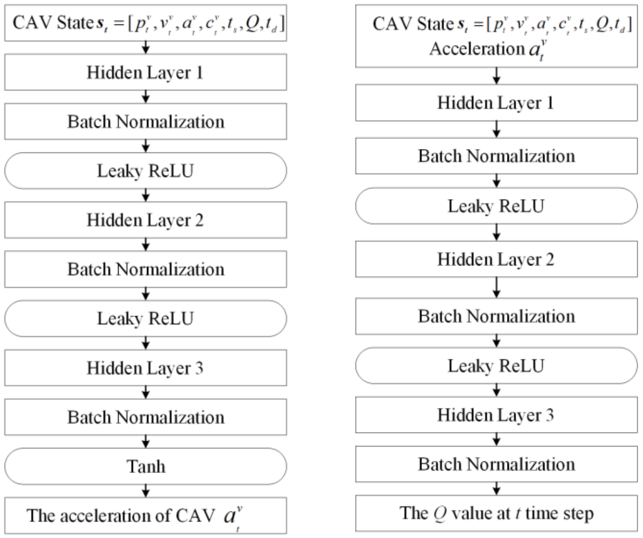
$$Q(\boldsymbol{s_t^v}, a_t|\theta^Q) \tag{5}$$

where the θ^Q is the state-action value function represented as DNN. The structure of the actor network is shown in Figure 5 (b). The θ^μ and the θ^Q are updated by (6)-(7):

$$\nabla_{\Theta^\mu}, j(\Theta) = \frac{1}{N}\sum_i [\nabla_a Q(\boldsymbol{s}, a|\theta^Q|_{s=\boldsymbol{s_i^v}, a=\mu(\boldsymbol{s_i^v})} \nabla_{\Theta^\mu}\mu(\boldsymbol{s}|\theta^\mu)|_{\boldsymbol{s_i^v}}] \tag{6}$$

$$L = \frac{1}{N}\sum_i (y_i - Q(\boldsymbol{s_i^v}, a_i|\theta^Q))^2 \tag{7}$$

where the y_i is the output of the target critic network, and the N means the batch size.



(a) Actor network (b) Critic network

Figure 5. The architecture of actor-critic of E-DDPG.

(2) The Reward Function

The reward function calculates the reward for the CAV acting a_t from the state $\boldsymbol{s_t^v}$ when transitioning to the next state $\boldsymbol{s_{t+1}^v}$. It affects both the agent learning efficiency and the likelihood of risky driving behaviors, such as collisions or running red lights. The reward function can be formulated as follows:

$$r_t^v = r_t^a + r_t^b + r_t^c + r_t^d + r_t^e + r_t^f \tag{8}$$

In (8), the r_t^a represents the fuel consumption penalty term. To avoid the possible negative impact of the different magnitude reward values on the convergence of the cumulative rewards, the fuel consumption of CAV at each time step is normalized. It can be formulated with the (9):

$$r_t^a = -\frac{f_t - f_{min}}{f_{max} - f_{min}} \tag{9}$$

Where the f_{min} and the f_{max} represent the minimum fuel consumption and the maximum fuel consumption, respectively.

The r_t^b is the traffic efficiency reward term. The ratio of the CAV's position and the control zone's length is used as a bonus value for efficiency.

$$r_t^b = \begin{cases} \frac{p_t}{L}, & c_v \le T_G \\ -1, & c_v > T_G \end{cases} \tag{10}$$

It can be seen from (10) that the r_t^b is positively correlated with the position where the CAV is located. The r_t^b is p_t/L when the total control time c_v is less than the remaining green signal timing. The r_t^b is set as -1 when the c_v is greater than the remaining green signal timing. Additionally, when the CAV enters the control zone, and the traffic light is red, running a red light should be avoided. Therefore, (10) should be rewritten as (11):

$$r_t^b = \begin{cases} -1, c_v < T_R \; and \; p_t > L \\ \frac{p_t}{L}, \; r_t \le T_R + T_G \\ -1, \; r_t > T_R + G \end{cases} \tag{11}$$

The r_t^c is the passenger comfort reward term, and it can be defined as (12):

$$r_t^c = -\frac{(\frac{a_t^v - a_{t-1}^v}{\Delta t})^2}{3600} \tag{12}$$

To encourage the CAV to travel at a higher velocity as much as possible while satisfying the constraints, a larger reward should be set. However, it would sometimes lead the vehicle to exceed the limitation and make the CAV agent training process unstable. Consequently, the reward is constant. The velocity reward term r_t^d can be shown in (13):

$$r_t^d = \begin{cases} 0.1, & 0 \le v_t^v \le v_{max} \\ -1, & else \end{cases} \tag{13}$$

In addition, to avoid the CAV performing emergency braking when approaching the signalized intersection when the traffic light is red, the velocity of the CAV in the remaining red signal timing should be less than L/T_R. Therefore, (13) should be rewritten as (14):

$$r_t^d = \begin{cases} 0.1, c_v < T_R \; and \; 0 \le v_t^v \le \frac{L}{T_R} \\ 0.1, \quad 0 \le v_t^v \le v_{max} \\ -1, \quad else \end{cases} \tag{14}$$

r_t^e is the collision penalty. When the minimum time gap t_s is less than 0.8 seconds, the collision avoidance bonus is set to -1, and when the t_s is greater than or equal to 0.8 seconds, the collision avoidance bonus $r_c(t)$ is set to 1, as shown in (15):

$$r_t^e = \begin{cases} 1, & t_s \ge 0.8 \\ -1, & t_s < 0.8 \end{cases} \tag{15}$$

As shown in Figure 6, when the CAV₁ enters the control zone, the queue formed by three HVs has the length Q, and the queue dissipates over a period t_d. Under these conditions, the distance traveled by CAV₁ during $t_R + t_d$ should below $L - Q$, and the corresponding total control time c_v must be less than $t_R + t_d$.

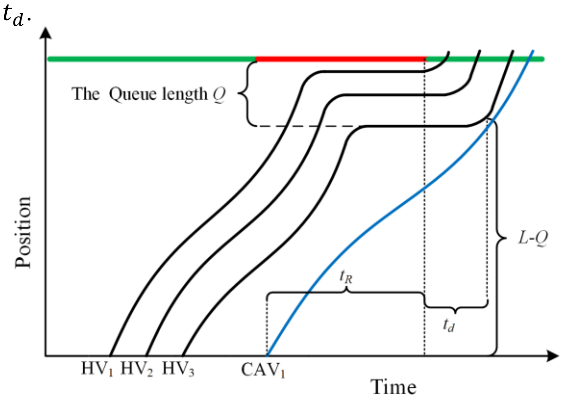


Figure 6. The illustration of queue length.

Thus, r_t^f can be formulated with (16):

$$r_t^f = \begin{cases} 1, & c_v < t_R + t_d and\, p_t^v < L - Q \\ -1, & else \end{cases} \tag{16}$$

When the CAV1 enters the control zone, and the corresponding traffic signal light is green, (16) can be rewritten as (17):

$$r_t^f = \begin{cases} 1, & c_v < t_d and\, p_t^v < L - Q \\ -1, & else \end{cases} \tag{17}$$

(3) The expert dataset construction

The expert dataset construction process is shown in Algorithm 1. Lines 5-9 initialize the CAV position, velocity, acceleration, total control time, and the corresponding remaining green timing. Line 10 outputs the acceleration generated by the GA, and Line 12-23 adds the position, velocity, acceleration, and total control time to the corresponding data vectors.

Algorithm 1 The Construction Process of the Expert Dataset

```
1: **Input:** The initial state of CAV $\boldsymbol{s}_0^v = [p_0^v, v_0^v, a_0^v, c_v]$
2: **Output:** The training dataset $\boldsymbol{D_1}$ and the label dataset $\boldsymbol{D_2}$ of the actor network. The training dataset $\boldsymbol{D_3}$ and the label dataset $\boldsymbol{D_4}$ of the critic network.
3: Initialize $\boldsymbol{D_1}$, $\boldsymbol{D_2}$, $\boldsymbol{D_3}$, $\boldsymbol{D_4}$ and $\boldsymbol{D_5}$ are empty.
4: **for** $i = 1$ **to** 1000 **do**
5:     $p_0^v = 0$
6:     $v_0^v = U(v_{min}, v_{max})$
7:     $a_0^v = U(a_{min}, a_{max})$
8:     $c_v = 0$
9:     $T_G = U(10,30)$
10:    out = GA ($\boldsymbol{s}_0^v$,$T_G$)
11:    L = Length(out)
12:    if L≠0 then
13:      for $t = 1$ to L do
14:        $\boldsymbol{s}_t^v = out[t,:]$
15:        $a_t^v = out[t,2]$
16:        Calculate the reward $r(t)$
17:        The current state $\boldsymbol{s}_t^v$ is added to dataset $\boldsymbol{D_1}$
18:        The current action $a_t^v$ is added to dataset $\boldsymbol{D_2}$
19:        The current state $\boldsymbol{s}_t^v$ and action $a_t^v$ are added to dataset $\boldsymbol{D_3}$
20:        The reward $r(t)$ is added to dataset $\boldsymbol{D_4}$
21:        The state at next time step $\boldsymbol{s}_{t+1}^v$ is added to $\boldsymbol{D_5}$
22:      **end for**
23:    **end if**
24: **end for**
```

The training dataset $\boldsymbol{D_1}$ and the labeled dataset $\boldsymbol{D_2}$ obtained from Algorithm 1 are used to pre-train the actor network. Since the output of the actor network is the acceleration of the CAV, mean square error (MSE) is chosen as the loss function to make the actor network gradually approximate the optimal acceleration corresponding to the state of the CAV at each time step, as shown in (18):

$$L_A = \frac{1}{N}\sum_{t=1}^{N}(a_t^e - a_t)^2 \tag{18}$$

where N denotes the batch size of one epoch, a_t^e is the acceleration of the CAV generated by GA, and a_t is the acceleration of CAV output by the actor network.

The training dataset $\boldsymbol{D_3}$ and the label dataset $\boldsymbol{D_4}$ obtained from Algorithm 1 are used to pre-train the critic network. Since the Q-value of the critic network could not be obtained, the labeled dataset $\boldsymbol{D_4}$ is built on the result of the reward function at each time step. Similar to the actor network, MSE is chosen as the loss function to train the critic network as shown in (19):

$$L_C = \frac{1}{N}\sum_{t=1}^{N}(r^e(t) - q_t)^2 \tag{19}$$

where $r^e(t)$ is the total reward value obtained from the GA and q_t is the Q value of the critic network.

Additionally, the experience replay buffer constructed from the expert dataset is shown in (20):

$$\boldsymbol{D_e} = [\boldsymbol{D_1}, \boldsymbol{D_2}, \boldsymbol{D_3}, \boldsymbol{D_4}, \boldsymbol{D_5}] \tag{20}$$

where $\boldsymbol{D_e}$ is the matrix that is composed of CAV state vector at the next time step.

The CAV acceleration control is used as an example to explain the proposed E-DDPG method in detail, as shown in Algorithm 2. Here are two points that are different from the vanilla DDPG: The expert dataset is used to pre-train the actor network and the critic network. The speed of the cumulative rewards converging to the optimal policy is accelerated in this way, as shown in Algorithm 2, line 3; The experience buffer is also initialized with the expert dataset and is further updated during the training process as shown in Algorithm 2 in line 4 and line 14.

Algorithm 2 The Training Process of CAV Agent

```
1: **Input:** The initial state of CAV $\boldsymbol{s}_0^v = [p_0^v, v_0^v, a_0^v, c_0^v]$, and the remaining green signal timing $T_G$. The $p_0^v$, $v_0^v$ and $a_0^v$ are obtained by SUMO, and $c_0^v$ is 0.
2: **Output:** The position $p_t^v$, the velocity $v_t^v$, the acceleration $a_t^v$
3: The actor network and target actor network load the pre-training parameters $\theta_e^\mu$. The critic network and target critic network load the pre-training parameters $\theta_e^Q$.
4: The dataset $\boldsymbol{D_e}$ is used to initialize the experience replay buffer
5: **for** $episode = 1$ **to** $M$ **do**
6:    Initialize the noise: $\varepsilon_0 = 0.1$
7:    **for** $t = 1$ **to** $T_G$ **do**
8:      Generate the noise $\varepsilon_t$ with the O-U process
9:      The acceleration of the CAV: $a_t^v = \mu(\boldsymbol{s_t}|\theta^\mu) + \varepsilon_t$
10:      $v_{t+1}^v = v_t^v + a_t^v \Delta t$
11:      $p_{t+1}^v = p_t^v + p_t^v \Delta t + \frac{1}{2} a_t^v (\Delta t)^2$
12:     $c_v = c_v + 1$
13:      Calculated the reward $r_t$
14:      The vector $[\boldsymbol{s}_t^v, a_t^v, r_t, s_{t+1}]$ will be storage in $\boldsymbol{D_e}$
15:        N samples $[\boldsymbol{s}_i^v, a_i^v, r_i, s_{i+1}]$ are randomly selected from $\boldsymbol{D_e}$
16:      Calculate the noise based on O-U process:
           $\varepsilon_{t+1} = \varepsilon_t + \theta_t(\mu - \varepsilon_t)\Delta t + \sigma_t\sqrt{\Delta t} \cdot n$
17:      Calculate the target value:
           $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
18:      Update the parameters of the critic network and minimize the loss:
           $L = \frac{1}{N}\sum_{i=1}^{N}(y_i - Q(\boldsymbol{s_i^v}, a_i|\theta^Q))^2$
19:      Update the parameters of the actor network and the DNN:
20:             $\nabla_{\theta^\mu,\theta^\psi} J = \frac{1}{N}\sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \times (\nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s_i} + \nabla_{\theta^\psi}\psi(s|\theta^\psi)|_{s_i})$
```

```
21:        Update the target network:
22:                $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
23:    end for
24: end for
```

B. Traffic Signal Light Agent

The traffic signal light agent is constructed using the D3QN algorithm, which is well-suited for discrete action space. The D3QN algorithm combines the advantages of the DQN algorithm, the Dueling Q Network algorithm, and the Double Q Network algorithm. This combination allows for more effective handling of control problems in high-dimensional, complex state spaces, thereby enhancing the training efficiency and decision-making quality of the traffic signal light agent.

(1) State space

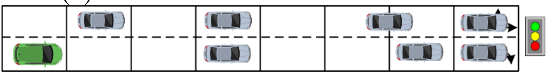
Compared to macroscopic traffic flow parameters such as overall traffic flow and average speed, lane-level traffic congestion is more appropriate as an input variable for the signal light agent. To quantify the degree of congestion in different lanes, the control zone is discretized into multiple cells, as illustrated in Figure 7 (a). The state space $\boldsymbol{s_t^l}$ of the signal light agent at the t-th time step is then defined by (21):

$$\boldsymbol{s_t^l} = \{s_t^l(x,y) | 1 \leqq x \leqq X, 1 \leqq y \leqq Y\} \tag{21}$$

where X represents the number of lanes, and Y can be defined with (22), and c_l is the cell length:

$$Y = \lceil \frac{L}{c_l} \rceil \tag{22}$$

When a vehicle occupies a single cell, the corresponding cell value is set to 1. If a vehicle spans two cells, both corresponding cells are set to 1; otherwise, the cell value is set to 0. Assuming the positions of all vehicles in the control zone at the t-th time step are as shown in Figure 7 (a), the value of X is 2, and the value of Y is 8. The state space $\boldsymbol{s_t^l}$ is then represented as shown in Figure 7 (b).



(a)

0	1	0	1	0	1	1	1
1	0	0	1	0	0	1	1

(b)

Figure 7. The design of the state space of traffic signal light agent.

(2) Action space

The action space of the traffic signal light agent is defined as $a_t^l = [-1,0,1]$, where -1 indicates a reduction of the green light timing by 0.1 seconds at the t-th time step, 0 means the green light timing remains unchanged, and 1 signifies an extension of the green light timing by 0.1 seconds at the t-th time step. However, the process mentioned above does not mean the next green duration would be adjusted at each time step. Once switch to the next green phase, the adjusted green duration based on the traffic signal light control agent will be employed immediately.

(3) Reward function

The accumulated total waiting time w_t^x of each lane is used as the reward function of the traffic light agent, as shown in (23):

$$r_t^l = \sum_x^X \sum_t^T w_t^x \tag{23}$$

where T represents the simulation duration, and I is the number of CAVs, and w_t^x can obtained from SUMO.

(4) D3QN Algorithm

The Q-value of the D3QN is calculated with (24). It consists of the optimal state-value function V_μ and the optimal dominance function A_μ:

$$Q_\mu^l(\boldsymbol{s_t^l}, a_t^l | \gamma, \alpha, \beta) = V_\mu(\boldsymbol{s_t^l} | \gamma, \beta) + A_\mu(\boldsymbol{s_t^l}, a_t^l | \gamma, \alpha) \tag{24}$$

where γ represents the weight of the convolutional neural network (CNN) in the Dueling DQN algorithm, and α and β are the weights of the deep neural network (DNN).

The D3QN algorithm separately estimates the value corresponding to the state of the traffic signal light agent and the advantage of action against other actions in the action space $\boldsymbol{A}$ through (24). It adjusts the weight parameters α, β, and γ during the training process. This allows for the Q-value to be estimated under the condition that the action corresponding to the state does not affect the environment. However, when using the backpropagation algorithm to update γ, the Q-value may correspond to multiple value functions and advantage functions. Therefore, since (24) is not differentiable, it needs to be transformed into (25).

$$Q_\mu^l(\boldsymbol{s_t^l}, a_t^l | \gamma, \alpha, \beta) = V_\mu(\boldsymbol{s_t^l} | \gamma, \beta) + (A_\mu(\boldsymbol{s_t^l}, a_t^l | \gamma, \alpha) - \frac{1}{|\boldsymbol{A}|} \sum_{a_{t+1}^l} A_\mu(\boldsymbol{s_t^l}, a_{t+1}^l | \gamma, \alpha)) \tag{25}$$

The loss function of the Dueling DQN model is further modified to the Double DQN model, resulting in the following loss function for the D3QN model:

$$Loss = E[\left(Y_t - Q_\mu^l(\boldsymbol{s_{t+1}^l}, a_t^l | \gamma, \alpha, \beta)\right)^2] \tag{26}$$

where Y_t is the target value, as shown in (27):

$$Y_t = R_{t+1} + \gamma Q_\mu^l(\boldsymbol{s_{t+1}^l}, argmax_a Q_\mu^l(\boldsymbol{s_{t+1}^l}, a_t^l | \gamma, \alpha, \beta) | \gamma^{'}, \alpha, \beta) \tag{27}$$

where $\gamma^{'}$ represents the parameter of the target network.

A. Cooperative Control Process

The specific training process of the MARL-CTV model proposed in this paper is detailed in Algorithm 3. Lines 4-17 outline the training steps for the traffic signal light agent, while lines 18-20 describe the training process of the CAV agent. Both the E-DDPG-based CAV agents and the D3QN-based traffic signal light agent face the exploration-exploitation dilemma during training. Exploration enables agents to discover new actions, while exploitation prompts agents to choose the best-known actions based on current knowledge. Given that the action space of the traffic signal light is discrete and that of CAV is continuous, different strategies are required to address this issue. For the CAV agent, the Ornstein–Uhlenbeck process is employed. For the traffic signal light agent, an improved greedy algorithm is used to balance exploration and exploitation.

As shown in line7, during the training process, a random value x that follows a uniform distribution in the interval [0,1] is generated at each time step for every episode. Let assume E is the number of total episode, if x falls within the interval $[0, \frac{E-episode}{E-1}]$, the action of the traffic signal light agent at the current time step is randomly selected. If x does not fall within this interval, the action of the traffic signal light agent is the output value of the D3QN algorithm. This approach increases the likelihood that x falls within the interval $[0, \frac{M-episode}{M-1}]$

during early episodes, encouraging the traffic signal light agent to perform exploratory actions at the beginning of training. As the number of episodes increases, the probability of x falls outside the interval $[0,\frac{M-episode}{M-1}]$ grows, encouraging the traffic signal light agent to perform exploitation actions in the later stages of training. Lines 14-17 describe the process for updating the experience replay buffer and the DNN's parameters of the traffic signal light agent.

Algorithm 3 The training process of MARL-CTV model.

```
1: Input: CAV agent initial state $\boldsymbol{s_v^t}$, signal light agent initial state $\boldsymbol{s_t^l}$, simulated total run time $T$
2: Output: The position vector $\boldsymbol{P}$, the velocity vector $\boldsymbol{V}$, the acceleration vector $\boldsymbol{A_V}$, and the action vector of traffic signal light $\boldsymbol{A_L}$
3: Loading the pre-trained actor network $\theta_e^\mu$, critic networks $\theta_e^Q$, and experience replay buffer $\boldsymbol{D_e}$ of the CAV agent
4: Initialize the experience replay buffer $\boldsymbol{D_L}$ of traffic signal light agent
5: for $episode = 1$ to $M$ do
6:   for $t = 1,$ to $T$ do
7:     $x = \mu(0,1)$
8:     if $x \in [0,\frac{E-episode}{M-1}]$ then
9:       $a_t^l = random(-1,0,1)$
10:    else
11:      $a_t^l = argmax_a Q_u^l(\boldsymbol{s_t^l}, a_t^l|\gamma,\alpha,\beta)$
12:    end if
13:    take the action $a_t^l$, calculate the reward $r_t^l$, and update the status $\boldsymbol{s_{t+1}^l}$
14:    Store $(\boldsymbol{s_t^l}, a_t^l, r_t^l, \boldsymbol{s_{t+1}^l})$ to $\boldsymbol{D_L}$
15:    Randomly sample $N$ instances from dataset $\boldsymbol{D_L}$
16:    Calculate the loss and update the network parameters $\gamma,\alpha,\beta$
17:    Updates target network
18:    if CAV agent enters the control zone, then
19:      Execute the DDPG algorithm
20:    end if
21:  end for
22: end for
```

IV. Experiments

CARLA is integrated with SUMO to create a joint simulation platform to enhance the fidelity of vehicle dynamics simulation, as shown in Figure 8. CARLA's vehicle modelling relies on a detailed physics engine, making it more realistic than the simplified kinematic models used in SUMO. The first case study tests the performance of the MARL-CTV model under varying traffic flow conditions. The second case study tests the robustness of the MARL-CTV model using real-world signalized intersections from the CitySim dataset [63]. The details of each case study will be introduced in the Signalized Intersection Description section. Additionally, the PressLight [34] as one of the well-known traffic signal control methods is selected to compared with the proposed MARL-CTV in this paper.

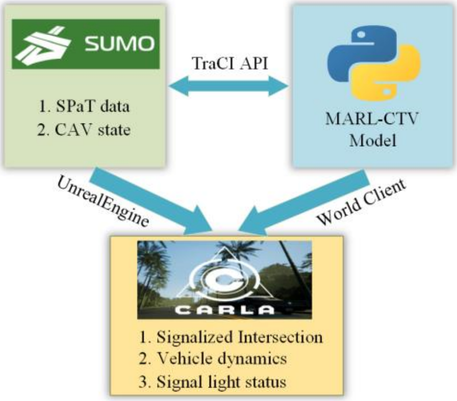


Figure 8. The architecture of joint simulation platform.

To test the performance of the proposed enhanced DDPG (E-DDPG) method for the CAV agent control, the vanilla DDPG and soft actor-critic (SAC) are used as benchmarks. Additionally, the cumulative reward of different methods at each episode is shown. Figure 9 presents the comparison results of different DRL models: E-DDPG, DDPG, and SAC. Each model was trained for 10 rounds to evaluate stability and generalization. The blue, orange, and green curves represent the average cumulative rewards of E-DDPG, DDPG, and SAC, respectively, with the shaded regions indicating the distribution of cumulative rewards across the 10 training rounds. The E-DDPG model shows a steady increase in returns, stabilizing between 50-60 after the 200th episode, and it demonstrates the superior performance and consistency of E-DDPG. SAC, while performing slightly better than DDPG, exhibits high variance, indicating less reliability. DDPG has the highest variance and lowest cumulative rewards, and it indicates that DDPG performed not well compared to E-DDPG and SAC.

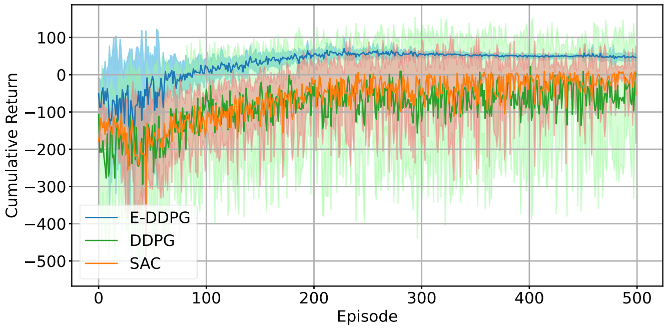


Figure 9. The average cumulative reward of three methods.

Due to the imbalance traffic flow distribution of signalized intersection at the arterial corridor [64]-[65], a signalized intersection scenario is built in SUMO, as shown in Figure 10**.** The traffic flow for straight lane one from west to east is set to 900 veh/h, and for straight lane 2, it is set to 1200 veh/h. The traffic flow for both straight lanes from east to west is set to 700 veh/h. The traffic flow for straight lane one and straight lane two from south to north and from north to south is set to 600 veh/h. The blue vehicles represent CAVs, and the yellow vehicles represent HVs. The flow model of SUMO is used to generate HVs and CAVs under the different market penetration rates (MPRs) of CAV. The parameter values of the SUMO simulation are given in Table I, and the number of vehicles arriving at signalized intersection follows the Poisson distribution. Except that, many SUMO parameters can affect the experimental results. However, determining the appropriate parameter settings is outside the scope of this paper. Consequently, we rely on the default settings for all parameters except those listed in Table I.

Table I The Parameters Values of Simulation

Parameters	Values
Simulation time step	0.1 s
Control zone length	200 m
Initial velocity	8 m/s to 12 m/s
Initial acceleration	-3 m/s^2 to 3 $\mathrm{m/s}^2$
Initial traffic signal light length	90 s
Initial green light duration	45 s
Initial red light duration	45 s
The simulation duration	660 s

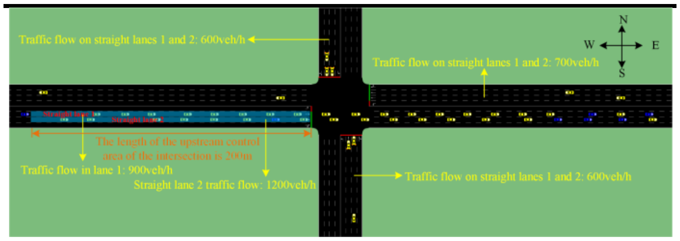

Traffic flow on straight lanes 1 and 2: 600veh/h
N
W
E
S
Traffic flow on straight lanes 1 and 2: 700veh/h
Straight lane 1
Straight lane 2
The length of the upstream control area of the intersection is 200m
Traffic flow in lane 1: 900veh/h
Straight lane 2 traffic flow: 1200veh/h
Traffic flow on straight lanes 1 and 2: 600veh/h

Figure 10. The isolated signalized intersection defined in SUMO.

A case study based on actual signalized intersection can effectively evaluate the robustness of the MARL-CTV model. Therefore, the ground-truth data provided by Intersection D of CitySim, which includes vehicle dynamic information, SPaT, and the layout of the signalized intersection, was selected to test the proposed MARL-CTV model, as shown in Figure 11. This signalized intersection is located at both ends of a large student garage at the University of Florida in Orlando, Florida, USA. The light blue area from north to south is defined as the control zone. Additionally, data analysis revealed that the real traffic flow from north to south is low, with the length of the control zone being 146.85 meters. The traffic flow is increased to 800 veh/h using the flow parameters in SUMO due to the number of vehicles recorded in the dataset is not enough.

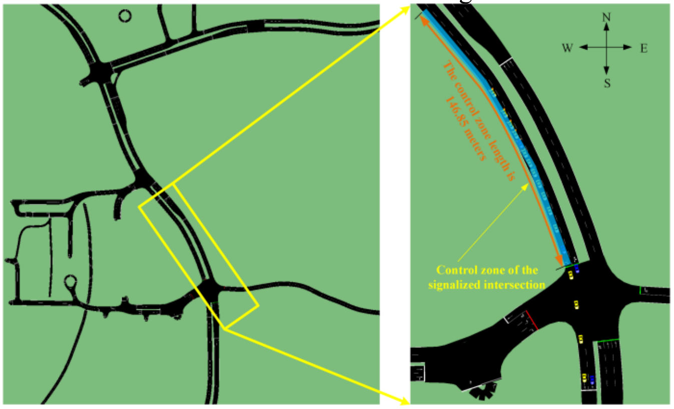

N
W
E
S
The control zone length is 146.85 meters
Control zone of the signalized intersection

Figure 11. The isolated signalized intersection defined in CitySim dataset.

A. The First Case Study

The MARL-CTV model was trained for 10 rounds, with 500 episodes per round, and the cumulative reward of the two types of agents is shown in Figure 12. The cumulative reward of the traffic signal light agent, depicted by the red curve, converged after 400 episodes. The cumulative reward of the CAV agent, depicted by the blue curve, converged after 150 episodes. The small difference between the upper and lower bounds of the cumulative reward distribution for both types of agents across different rounds of training indicates that the MARL-CTV model is stable.

The following models are selected for the comparison: PressLight, Baseline (without any optimizations), MARL-C (only active the CAV agent control of MARL-CTV model), MARL-T (only active the traffic signal light agent control of MARL-CTV model), and MARL-CTV model. The time-space diagrams of the Baseline and MARL-T are given in Figure 13, and the blue curve represents the trajectory of HV.

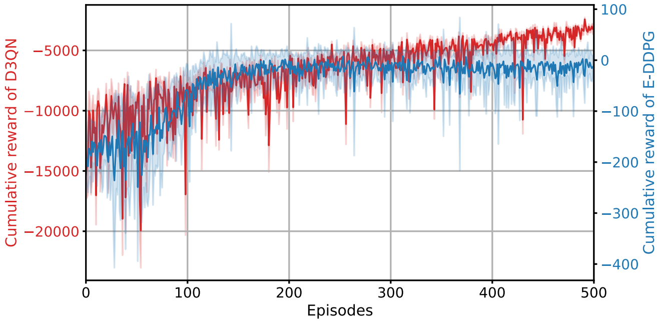
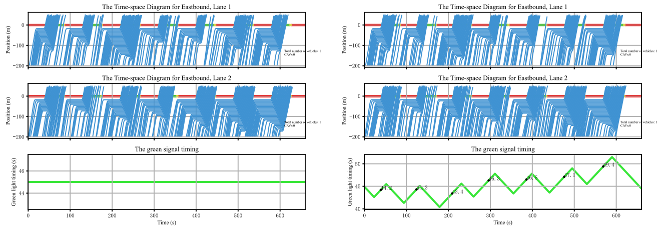


Figure 12. The cumulative reward of MARL-CTV model.

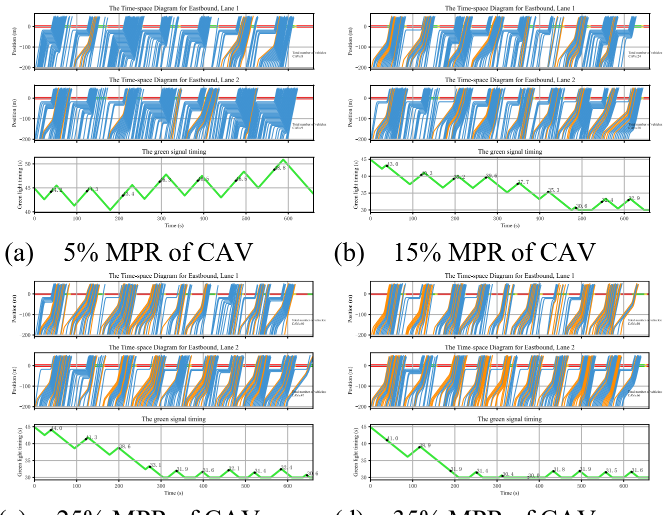


(a) Baseline (b) MARL-T

Figure 13. The time-space diagrams of the Baseline and MARL-T.

It can be seen from Figure 13(a) that the green light timing remained unchanged, only a few HVs could pass the signalized intersection without stop-and-go behavior, and most of the HVs needed to idle until the traffic light changed to green. When the MARL-T was activated, the green light timing was dynamically adjusted based on the total cumulative waiting time, with the black number representing the green light duration for each signal cycle. For example, during the first signal cycle, the simulation started with the cumulative total waiting time for HVs on lane 1 and lane 2 at 0, causing the green light timing to gradually decrease. As the vehicle queue was formed, the cumulative waiting time increased, and the remaining green light timing was extended to ensure subsequent HVs passed the signalized intersection. When the traffic signal changed from red to green, the vehicle queue dissipated, and the cumulative waiting time decreased, resulting in the green light timing reduction. Additionally, Figure 13 demonstrates that due to the high traffic flow, the green light timing of each signal cycle generally increased, mitigating the traffic congestion of signalized intersection. However, it should be noted there were still many HVs that needed to idle, indicating that the performance of MARL-T was limited.

The time-space diagrams of the MARL-CTV model under the 5%, 15%, 25%, and 35% MPR of CAV are shown in Figure 14, and the orange curve represents the trajectories of CAVs.



(a) 5% MPR of CAV (b) 15% MPR of CAV

(c) 25% MPR of CAV (d) 35% MPR of CAV

Figure 14. The time-space diagrams of the MARL-CTV under different MPR of CAV.

The performance of the CAV agent control was limited due to the MPR of CAV being only 5%, as shown in Figure 14 (a). This is particularly evident in the last signal cycle of Lane 2, in which the traffic flow was set at 1200 veh/h. In this cycle, the vehicle queue remained long because there were no CAVs present, and only the traffic light agent worked. Nevertheless, the efficiency was improved compared to the MARL-T model. From Figure 14 (b), traffic congestion in lanes 1 and 2 was significantly reduced under 15% MPR of CAV compared to 5% MPR of CAV, with a notable decrease in vehicle queue length. It can be concluded that the closer a CAV is to the head of the queue, the more HVs behind it can be indirectly controlled to pass the signalized intersection without idling. Additionally, the green signal timing gradually shortened over time, providing more green signal timing for the other phases and thereby improving the efficiency at the signalized intersection. As shown in Figures 14(c) and 14(d), vehicle queue lengths further decreased as the MPR of CAV increased. At 35% MPR of CAV, most vehicles could pass through the signalized intersection without stop-and-go behavior, except for a few HVs that needed to stop. The green light timing for each signal cycle ranged from 30 to 35 seconds. These results indicate that the MARL-CTV model can improve efficiency even without a high MPR of CAV.

To dive into the relationship between the trajectory and velocity, the time-space diagram with the trajectory colored by velocity is shown in Figure 15. It can be observed that most of the HVs approaching the signalized intersection needed to stop, indicated by the carmine color representing low velocity. When the MARL-CTV model was activated, the CAVs decelerated in advance in the control zone to pass the signalized intersection smoothly. However, under 15% MPR of CAV, there is a black ellipse in the time-space diagram where velocities fluctuated due to the minor estimation error in the queue length or queue dissipation time.

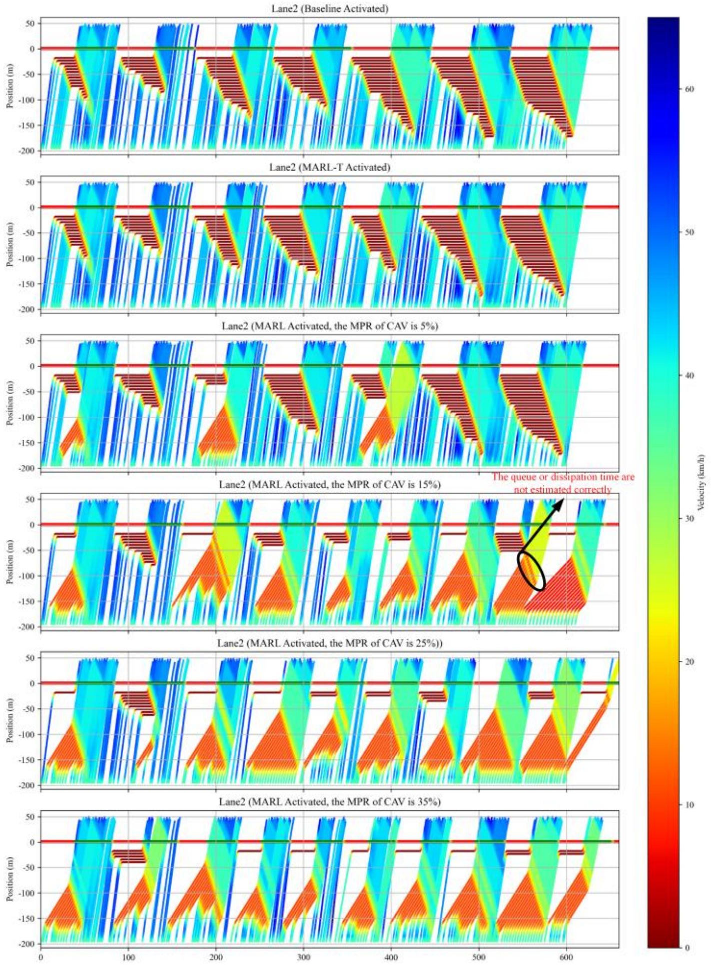


Figure 15. The time-space diagram with the trajectory colored by the velocity.

Five performance indices—average fuel consumption (AFC), number of traffic conflicts (NTC), average waiting time (AWT), average travel time (ATT), and number of vehicle stops (NVS)—are used to evaluate the performance of MARL-CTV. Here, NTC refers to instances where TTC values fall below 3 seconds. The results are shown in Figure 16.

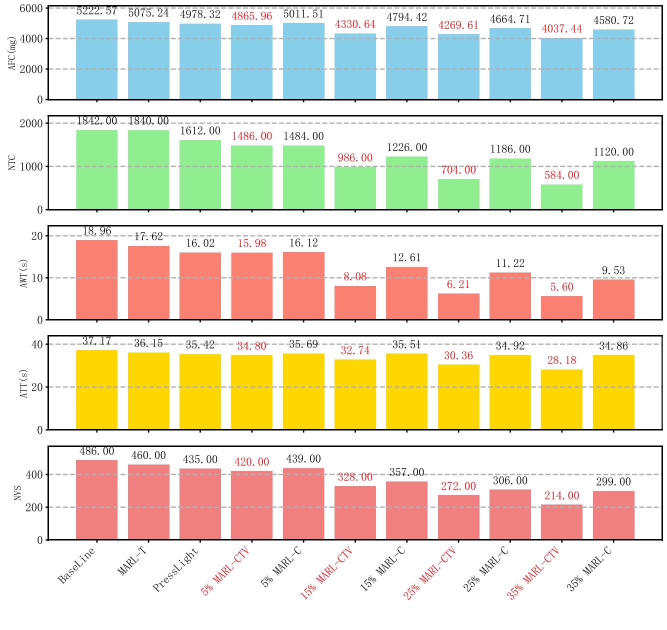
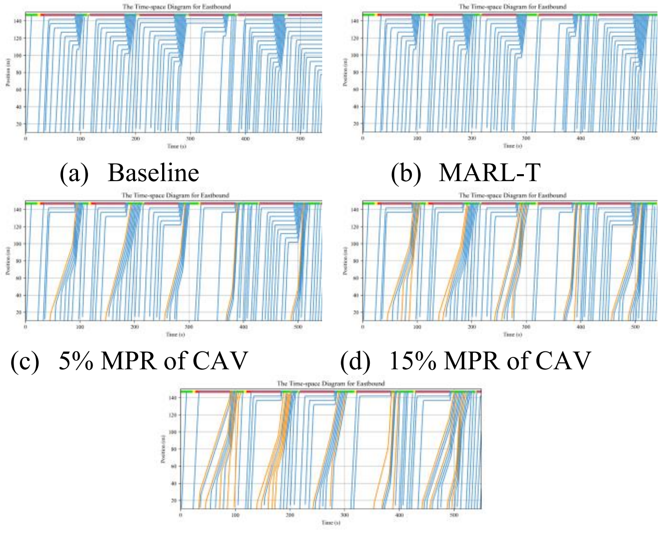


Figure 16. The comparison of four metrics of the different models.

The five metrics for the Baseline model were 5222.57mg, 1842, 18.96s, 37.17s, and 486 respectively, which were significantly higher than the results of the other three models, indicating severe traffic congestion without optimization. The MARL-T model's metrics were 5075.24 mg, 1840, 17.62 s, 36.15 s, and 460, showing reductions of 2.8%, 1%, 7.1%, 2.7%, and 5.4%, compared to the Baseline. It indicates that while the MARL-T model could mitigate traffic congestion under high traffic flow conditions, the improvement was modest. The PressLight model's metrics were 4978.32 mg, 1612, 16.02 s, 35.42 s, and 435, which were better than the MARL-T model, suggesting that the performance of the traffic signal light agent still needs improvement. When the MARL-CTV model was activated under a 5% MPR of CAV, its metrics were better than PressLight and further improved with increasing MPR of CAV. At 35% MPR of CAV, the MARL-CTV model's metrics were 4037.44 mg, 584, 5.60 s, 28.18 s, and 214, showing reductions of 22.7%, 68.3%, 70.5%, 24.2%, and 56%, compared to the Baseline, and 18.9%, 63.8%, 65%, 20.4%, 50.8% compared to the PressLight algorithm. These results indicate that the proposed MARL-CTV model can significantly improve traffic efficiency.

B. The Second Case Study

Since the total simulation time for Intersection D in the CitySim dataset is 1100 seconds, a subset of 560 seconds is selected to test the performance of the different models to reduce experimental complexity. The time-space diagram is shown in Figure 17, where the blue curves represent HVs, and the orange curves represent CAVs. From Figure 17 (a), it is evident that most HVs need to idle at the signalized intersection without any optimizations. When the MARL-T model was activated, the green light timing gradually extended, allowing subsequent arriving HVs to pass through the intersection without stop-and-go behavior. With the activation of the MARL-CTV model, efficiency increased with the rise in MPR of CAV. At 25% MPR of CAV, most vehicles could pass through the signalized intersection smoothly, as shown in Figure 17 (e).



(a) Baseline

(b) MARL-T

(c) 5% MPR of CAV

(d) 15% MPR of CAV

(e) 25% MPR of CAV

Figure 17. The time-space diagrams of the MARL-CTV under different MPR of CAV.

The results of five metrics in a real-world signalized intersection scenario are illustrated in Fig. 18. The Baseline model exhibited the highest values for all metrics, indicating severe traffic congestion. When the MARL-T model was activated, the metrics were 780.84 mg, 954, 14.00 s, 45.95 s, and 48 respectively, representing reductions of 1%, 9.8%, 7.4%, 11.6%, 23.8% compared to the Baseline model. The PressLight model improved the metrics compared to the Baseline and MARL-T models, but the values were higher when MARL-CTV was activated under the 5% MPR of CAV. At a 25% MPR of CAV, the metrics were 684.96 mg, 96, 7.89 s, 38.06 s, and 14 respectively, showing reductions of 13.1%, 90.9%, 47.8%, 26.8%, and 77.8% compared to the Baseline, and 7.6%, 78.2%, 41.2%, 10.7%, and 72.5% compared to the PressLight.

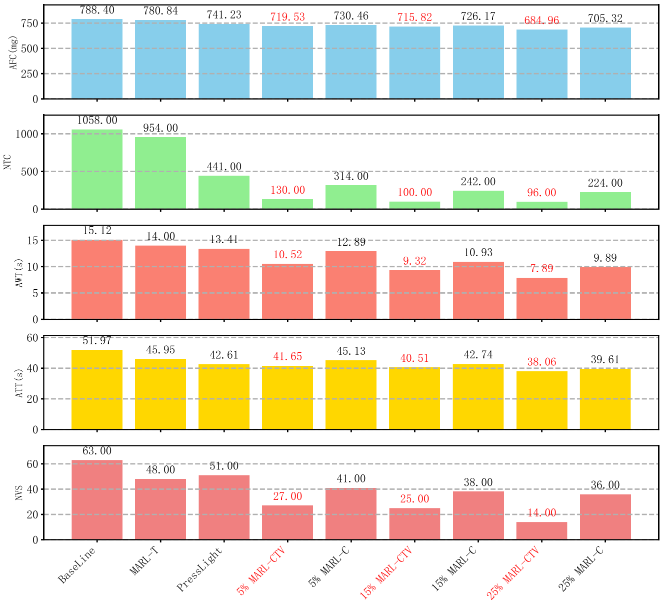


Figure 18. Comparison of four evaluation indexes of different algorithms

V. Conclusion

In this paper, we propose the MRAL-CTV model, which cooperatively controls the CAV agents and traffic signal light agent to improve the efficiency, fuel consumption and safety at the signalized intersection. The traffic signal light agent is controlled by the D3QN algorithm, while the CAV agents are controlled by the DDPG algorithm. To reduce the control burden and ensure the cumulative reward converges, the actor and critic networks of DDPG are pre-trained using the expert dataset. This dataset is constructed from multiple acceleration profiles generated by the genetic algorithm, based on various random initial states of CAV agents. Numerical experiments, based on real-world signalized intersection data, indicate that when the traffic flow of a single lane is 800 veh/h and the MPR of CAV is 25%, the average travel time is reduced by up to 26.8% compared to the Baseline and 10.7% compared to PressLight. Significant improvements in fuel consumption, number of traffic conflicts, and average waiting time are also observed.

Several research directions need to be further investigated. This paper focuses on the longitudinal control of CAV, excluding lateral maneuvers such as lane changing. The main reason is that incorporating lane-change behavior requires each CAV to track the real-time states of surrounding vehicles, thereby creating a high-dimensional state space. Therefore, the complexity of controlling multiple agents at signalized intersections increases significantly, making it challenging or even infeasible for the agents to discover an optimal policy. Additionally, future research will focus on how to promote the robustness of our proposed model under high traffic flow conditions across all four directions of the signalized intersection.

REFERENCES

[1] Shang, W. L., Chen, Y., Yu, Q., Song, X., Chen, Y., Ma, X., ... & Ochieng, W. (2023). Spatio-temporal analysis of carbon footprints for urban public transport systems based on smart card data. Applied Energy, 352, 121859.

[2] Lasley, P. (2023). *2023 Urban Mobility Report.*

[3] Shang, W. L., Zhang, M., Wu, G., Yang, L., Fang, S., & Ochieng, W. (2023). Estimation of traffic energy consumption based on macro-micro modelling with sparse data from Connected and Automated Vehicles. Applied Energy, 351, 121916.

[4] Sun, C., Guanetti, J., Borrelli, F., & Moura, S. J. (2020). Optimal eco-driving control of connected and autonomous vehicles through signalized intersections. *IEEE Internet of Things Journal*, 7(5), 3759-3773.

[5] Li, J., Yu, C., Shen, Z., Su, Z., & Ma, W. (2023). A survey on urban traffic control under mixed traffic environment with connected automated vehicles. *Transportation research part C: emerging technologies,* 154, 104258.

[6] Katsaros, K., Kernchen, R., Dianati, M., & Rieck, D. (2011, July). Performance study of a Green Light Optimized Speed Advisory (GLOSA) application using an integrated cooperative ITS simulation platform. *In 2011 7th International Wireless Communications and Mobile Computing Conference* (pp. 918-923). IEEE.

[7] Barth, M., Mandava, S., Boriboonsomsin, K., & Xia, H. (2011, June). Dynamic ECO-driving for arterial corridors. *In 2011 IEEE forum on integrated and sustainable transportation systems* (pp. 182-188). IEEE.

[8] Khan, S. M., & Chowdhury, M. (2021). Situation-aware left-turning connected and automated vehicle operation at signalized intersections. *IEEE Internet of Things Journal*, 8(16), 13077-13094.

[9] Ghiasi, A., Li, X., & Ma, J. (2019). A mixed traffic speed harmonization model with connected autonomous vehicles. *Transportation Research Part C: Emerging Technologies*, 104, 210-233.

[10] Zhou, Q., Zhou, B., Hu, S., Roncoli, C., Wang, Y., Hu, J., & Lu, G. (2023). A safety-enhanced eco-driving strategy for connected and autonomous vehicles: A hierarchical and distributed framework. *Transportation research part C: emerging technologies*, 156, 104320.

[11] Zhang, H., Fu, R., Wang, C., Guo, Y., & Yuan, W. (2022). Turning maneuver prediction of connected vehicles at signalized intersections: A dictionary learning-based approach. *IEEE Internet of Things Journal*, 9(22), 23142-23159.

[12] Hu, J., Li, S., Wang, H., Wang, Z., & Barth, M. J. (2024). Eco-approach at an isolated actuated signalized intersection: Aware of the passing time window. *Journal of Cleaner Production*, 435, 140493.

[13] Shi, J., Qiao, F., Li, Q., Yu, L., & Hu, Y. (2018). Application and evaluation of the reinforcement learning approach to eco-driving at intersections under infrastructure-to-vehicle communications. *Transportation Research Record,* 2672(25), 89-98.

[14] Mousa, S. R., Ishak, S., Mousa, R. M., Codjoe, J., & Elhenawy, M. (2020). Deep reinforcement learning agent with varying actions strategy for solving the eco-approach and departure problem at signalized intersections. *Transportation Research Record*, 2674(8), 119-131.

[15] Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., ... & Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971.*

[16] Zhou, M., Yu, Y., & Qu, X. (2019). Development of an efficient driving strategy for connected and automated vehicles at signalized intersections: A reinforcement learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 21(1), 433-443.

[17] Liu, B., Sun, C., Wang, B., & Sun, F. (2021). Adaptive speed planning of connected and automated vehicles using multi-light trained deep reinforcement learning. *IEEE Transactions on Vehicular Technology*, 71(4), 3533-3546.

[18] Fujimoto, S., Hoof, H., & Meger, D. (2018, July). Addressing function approximation error in actor-critic methods. *In International conference on machine learning* (pp. 1587-1596). PMLR.

[19] Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018, July). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *In International conference on machine learning* (pp. 1861-1870). PMLR.

[20] Wegener, M., Koch, L., Eisenbarth, M., & Andert, J. (2021). Automated eco-driving in urban scenarios using deep reinforcement learning. *Transportation research part C: emerging technologies*, 126, 102967.

[21] ang, S., Wang, Z., Jiang, R., Yan, R., & Du, L. (2022). Trajectory jerking suppression for mixed traffic flow at a signalized intersection: a trajectory prediction based deep reinforcement learning method. *IEEE Transactions on Intelligent Transportation Systems*, 23(10), 18989-19000.

[22] Guo, Q., Angah, O., Liu, Z., & Ban, X. J. (2021). Hybrid deep reinforcement learning based eco-driving for low-level connected and automated vehicles along signalized corridors. *Transportation Research Part C: Emerging Technologies*, 124, 102980.

[23] Gu, Z., Yin, Y., Li, S. E., Duan, J., Zhang, F., Zheng, S., & Yang, R. (2022). Integrated eco-driving automation of intelligent vehicles in multi-lane scenario via model-accelerated reinforcement learning. *Transportation Research Part C: Emerging Technologies*, 144, 103863.

[24] Yang, Z., Zheng, Z., Kim, J., & Rakha, H. (2024). Eco-driving strategies using reinforcement learning for mixed traffic in the vicinity of signalized intersections. Transportation Research Part C: Emerging Technologies, 165, 104683.

[25] Shang, Y., Zhu, F., Jiang, R., Li, X., & Wang, S. (2024). Trajectory planning at a signalized road section in a mixed traffic environment considering lane-changing of CAVs and stochasticity of HDVs. Transportation Research Part C: Emerging Technologies, 158, 104441.

[26] Yang, L., Zhan, J., Shang, W. L., Fang, S., Wu, G., Zhao, X., & Deveci, M. (2023). Multi-lane coordinated control strategy of connected and automated vehicles for on-ramp merging area based on cooperative game. IEEE Transactions on Intelligent Transportation Systems, 24(11), 13448-13461.

[27] Yang, Z., Zheng, Z., Kim, J., & Rakha, H. (2024). Eco-driving strategies using reinforcement learning for mixed traffic in the vicinity of signalized intersections. Transportation Research Part C: Emerging Technologies, 165, 104683.

[28] Ma, W., Wan, L., Yu, C., Zou, L., & Zheng, J. (2020). Multi-objective optimization of traffic signals based on vehicle trajectory data at isolated intersections. *Transportation research part C: emerging technologies*, 120, 102821.

[29] Dasgupta, S., Rahman, M., & Jones, S. (2024). Harnessing Digital Twin Technology for Adaptive Traffic Signal Control: Improving Signalized Intersection Performance and User Satisfaction. *IEEE Internet of Things Journal*.

[30] Wei, H., Zheng, G., Yao, H., & Li, Z. (2018, July). Intellilight: A reinforcement learning approach for intelligent traffic light control. *In Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 2496-2505).

[31] Chu, T., Wang, J., Codecà, L., & Li, Z. (2019). Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE transactions on intelligent transportation systems,* 21(3), 1086-1095.

[32] Van der Pol, E., & Oliehoek, F. A. (2016). Coordinated deep reinforcement learners for traffic light control. Proceedings of learning, inference and control of multi-agent systems (at NIPS 2016), 8, 21-38.

[33] Vlachogiannis, D. M., Wei, H., Moura, S., & Macfarlane, J. (2024). HumanLight: Incentivizing ridesharing via human-centric deep reinforcement learning in traffic signal control. T*ransportation Research Part C: Emerging Technologies*, 162, 104593.

[34] Wei, H., Chen, C., Zheng, G., Wu, K., Gayah, V., Xu, K., & Li, Z. (2019, July). Presslight: Learning max pressure control to coordinate traffic signals in arterial network. *In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 1290-1298).

[35] Shang, W. L., Chen, Y., Li, X., & Ochieng, W. Y. (2020). Resilience Analysis of Urban Road Networks Based on Adaptive Signal Controls:

Day-to-Day Traffic Dynamics with Deep Reinforcement Learning. Complexity, 2020(1), 8841317.

[36] Chen, C., Wei, H., Xu, N., Zheng, G., Yang, M., Xiong, Y., ... & Li, Z. (2020, April). Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. *In Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 04, pp. 3414-3421).

[37] Shabestary, S. M. A., & Abdulhai, B. (2022). Adaptive traffic signal control with deep reinforcement learning and high dimensional sensory inputs: Case study and comprehensive sensitivity analyses. *IEEE Transactions on Intelligent Transportation Systems*, 23(11), 20021-20035.

[38] Aslani, M., Mesgari, M. S., & Wiering, M. (2017). Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. *Transportation Research Part C: Emerging Technologies*, 85, 732-752.

[39] Li, L., Zhu, R., Wu, S., Ding, W., Xu, M., & Lu, J. (2023). Adaptive multi-agent deep mixed reinforcement learning for traffic light control. IEEE Transactions on Vehicular Technology, 73(2), 1803-1816.

[40] Zhu, R., Ding, W., Wu, S., Li, L., Lv, P., & Xu, M. (2023). Auto-learning communication reinforcement learning for multi-intersection traffic light control. Knowledge-Based Systems, 275, 110696.

[41] Yang, S., Yang, B., Zeng, Z., & Kang, Z. (2023). Causal inference multi-agent reinforcement learning for traffic signal control. Information Fusion, 94, 243-256.

[42] Rahmani, S., Baghbani, A., Bouguila, N., & Patterson, Z. (2023). Graph neural networks for intelligent transportation systems: A survey. IEEE Transactions on Intelligent Transportation Systems, 24(8), 8846-8885.

[43] Zhang, Y., & Su, R. (2021). An optimization model and traffic light control scheme for heterogeneous traffic systems. Transportation research part C: emerging technologies, 124, 102911.

[44] Li, Z., Elefteriadou, L., & Ranka, S. (2014). Signal control optimization for automated vehicles at isolated signalized intersections. *Transportation Research Part C: Emerging Technologies*, 49, 1-18.

[45] Guo, Y., Ma, J., Xiong, C., Li, X., Zhou, F., & Hao, W. (2019). Joint optimization of vehicle trajectories and intersection controllers with connected automated vehicles: Combined dynamic programming and shooting heuristic approach. *Transportation research part C: emerging technologies*, 98, 54-72.

[46] Al Islam, S. B., & Hajbabaie, A. (2017). Distributed coordinated signal timing optimization in connected transportation networks. *Transportation Research Part C: Emerging Technologies*, 80, 272-285.

[47] Feng, Y., Yu, C., & Liu, H. X. (2018). Spatiotemporal intersection control in a connected and automated vehicle environment. *Transportation Research Part C: Emerging Technologies*, 89, 364-383.

[48] Xu, B., Ban, X. J., Bian, Y., Li, W., Wang, J., Li, S. E., & Li, K. (2018). Cooperative method of traffic signal optimization and speed control of connected vehicles at isolated intersections. *IEEE Transactions on Intelligent Transportation Systems*, 20(4), 1390-1403.

[49] Liu, M., Zhao, J., Hoogendoorn, S. P., & Wang, M. (2022). An optimal control approach of integrating traffic signals and cooperative vehicle trajectories at intersections. *Transportmetrica B: transport dynamics*, 10(1), 971-987.

[50] Yu, C., Feng, Y., Liu, H. X., Ma, W., & Yang, X. (2018). Integrated optimization of traffic signals and vehicle trajectories at isolated urban intersections. *Transportation research part B: methodological*, 112, 89-112.

[51] Yu, C., Feng, Y., Liu, H. X., Ma, W., & Yang, X. (2019). Corridor level cooperative trajectory optimization with connected and automated vehicles. *Transportation Research Part C: Emerging Technologies*, 105, 405-421.

[52] Yang, Z., Feng, Y., & Liu, H. X. (2021). A cooperative driving framework for urban arterials in mixed traffic conditions. *Transportation research part C: emerging technologies*, 124, 102918.

[53] Ying, J., & Feng, Y. (2024). Infrastructure-Assisted cooperative driving and intersection management in mixed traffic conditions. *Transportation Research Part C: Emerging Technologies*, 158, 104443.

[54] Jiang, X., & Shang, Q. (2022). A dynamic CAV-dedicated lane allocation method with the joint optimization of signal timing parameters and smooth trajectory in a mixed traffic environment. *IEEE Transactions on Intelligent Transportation Systems*, 24(6), 6436-6449.

[55] Liu, H., Kurzhanskiy, A. A., Hong, W., & Lu, X. Y. (2024). Integrating vehicle trajectory planning and arterial traffic management to facilitate eco-approach and departure deployment. *Journal of Intelligent Transportation Systems*, 1-14.

[56] Guo, J., Cheng, L., & Wang, S. (2023). CoTV: Cooperative control for traffic light signals and connected autonomous vehicles using deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 24(10), 10501-10512.

[57] Han, Y., Wang, M., & Leclercq, L. (2023). Leveraging reinforcement learning for dynamic traffic control: A survey and challenges for field implementation. *Communications in Transportation Research*, 3, 100104.

[58] Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y. P., Hilbrich, R., ... & Wießner, E. (2018, November). Microscopic traffic simulation using sumo. *In 2018 21st international conference on intelligent transportation systems (ITSC)* (pp. 2575-2582). IEEE.

[59] Fang, S., Yang, L., Zhao, X., Wang, W., Xu, Z., Wu, G., ... & Qu, X. (2023). A Dynamic Transformation Car-Following Model for the Prediction of the Traffic Flow Oscillation. IEEE Intelligent Transportation Systems Magazine.

[60] Wen, M., Kuba, J., Lin, R., Zhang, W., Wen, Y., Wang, J., & Yang, Y. (2022). Multi-agent reinforcement learning is a sequence modeling problem. Advances in Neural Information Processing Systems, 35, 16509-16521.

[61] Li, S., Puig, X., Paxton, C., Du, Y., Wang, C., Fan, L., ... & Zhu, Y. (2022). Pre-trained language models for interactive decision-making. Advances in Neural Information Processing Systems, 35, 31199-31212.

[62] Liu, H. X., Wu, X., Ma, W., & Hu, H. (2009). Real-time queue length estimation for congested signalized intersections. *Transportation research part C: emerging technologies*, 17(4), 412-427.

[63] Zheng, O., Abdel-Aty, M., Yue, L., Abdelraouf, A., Wang, Z., & Mahmoud, N. (2024). CitySim: a drone-based vehicle trajectory dataset for safety-oriented research and digital twins. *Transportation research record*, 2678(4), 606-621.

[64] Zheng, X., Huang, N., Bai, Y. N., & Zhang, X. (2023). A traffic-fractal-element-based congestion model considering the uneven distribution of road traffic. Physica A: Statistical Mechanics and its Applications, 632, 129354.

[65] Shiomi, Y., Taniguchi, T., Uno, N., Shimamoto, H., & Nakamura, T. (2015). Simulating lane-changing dynamics towards lane-flow equilibrium based on multi-lane first order traffic flow model. Transportation Research Procedia, 6, 128-143.