Learning the Policy for Mixed Electric Platoon Control of Automated and Human-Driven Vehicles at Signalized Intersection: A Random Search Approach

Xia Jiang, Jian Zhang, Member, IEEE, Xiaoyu Shi, and Jian Cheng

Abstract—The upgrading and updating of vehicles have accelerated in the past decades. Out of the need for environmental friendliness and intelligence, electric vehicles (EVs) and connected and automated vehicles (CAVs) have become new components of transportation systems. This paper develops a reinforcement learning framework to implement adaptive control for an electric platoon composed of CAVs and human-driven vehicles (HDVs) at a signalized intersection. Firstly, a Markov Decision Process (MDP) model is proposed to describe the decision process of the mixed platoon. Novel state representation and reward function are designed for the model to consider the behavior of the whole platoon. Secondly, in order to deal with the delayed reward, an Augmented Random Search (ARS) algorithm is proposed. The control policy learned by the agent can guide the longitudinal motion of the CAV, which serves as the leader of the platoon. Finally, a series of simulations are carried out in simulation suite SUMO. Compared with several traditional reinforcement learning approaches, the proposed method can obtain a higher reward. Meanwhile, the simulation results demonstrate the effectiveness of the delay reward, which is designed to outperform distributed reward mechanism. Compared with some state-of-the-art optimization-based frameworks, the simulation analysis reveals that more energy can be saved for different sizes of platoon. Sensitivity analysis is also conducted by adjusting the relative importance of the optimization goal to show the flexibility of ARS and delay reward setting. On the premise that travel delay is not sacrificed, the proposed control method can save up to 53.64% electric energy.

Index Terms—Connected and automated vehicles, reinforcement learning, platoon control, signalized intersection, random search.

Manuscript received 28 December 2021; revised 6 May 2022 and 17 November 2022; accepted 2 February 2023. Date of publication 16 February 2023; date of current version 8 May 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2021YFB1600500, in part by the Key Research and Development Program of Jiangsu Province under Grant BE2020013, and in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant SJKX21_0062. The Associate Editor for this article was S. E. Li. (Corresponding author: Jian Zhang.)

Xia Jiang and Xiaoyu Shi are with the School of Transportation, Southeast University, Nanjing, Jiangsu 210096, China (e-mail: summer142857.jiang@gmail.com; 2373497459@qq.com).
Jian Zhang is with the School of Transportation, Southeast University, Nanjing, Jiangsu 210096, China, and also with the School of Engineering, Tibet University, Lhasa, Tibet 850000, China (e-mail: jianzhang@seu.edu.cn).
Jian Cheng is with Nanjing Les Information Technology Company Ltd., Nanjing, Jiangsu 210096, China (e-mail: cheng.j@les.cn).

Digital Object Identifier 10.1109/TITS.2023.3242678

I. INTRODUCTION

The advancements in artificial intelligence (AI), communication technologies, and vehicular technology have promoted the automation and electrification of vehicles in recent years. Automation nurtures the creation of connected and automated vehicles (CAVs), which is widely accepted as an effective way to improve traffic conditions [1], [2], [3], [4], [5]. One important concern associated with the application of CAVs to real world is that the design of the control strategy should be real-time and effective, as the efficient functioning of the CAVs is based on their decision and control modules. The control task is especially challenged in signalized intersections, the places where traffic flows in different directions converge, which are viewed as the bottlenecks of urban traffic. Since the operation of vehicles can be interrupted by traffic signals, the control law of CAVs at signalized intersections is crucial for boosting traffic performance in such urban scenarios. Previous studies reveal the significance of CAV-related research at intersections by showing that proper control strategies are capable of improving travel efficiency, energy consumption, and safety of road traffic [6], [7], [8].

Based on the belief that CAVs are beneficial to intersection performance, relative research has gone through different stages. In the first stage, the most frequently discussed topic is a traffic environment with 100% CAV penetration rate, in which traffic signals can be eliminated, because the information of vehicular traffic can be completely obtained in real time [9] and CAVs can be controlled in a centralized manner [10], [11], [12]. Despite the fact that a pure CAV environment can create an unprecedented intelligent transportation system (ITS), there is a general consensus among researchers about the inevitability of the coexistence of CAVs and human-driven vehicles (HDVs) [13], [14], [15], [16], [17], [18], which is known as the so-called mixed traffic. Given this, controlling individual CAV under mixed traffic conditions becomes the main focus in the second stage of research, aiming to exploit the potential advantages that CAVs bring to urban transportation system. In this stage, having CAVs under control by embedded controller, relative indicators, such as travel time, energy consumption, and traffic safety, can be optimized for individual vehicle [19], [20], [21].
The formulation of control laws for CAV-related control problems in these studies is usually obtained by Model Predictive Control (MPC) [22], [23], [24] or Dynamic Programming (DP) [25], [26], which can be challenged with computation complexity. It is also pointed out that these model-based methods need to simplify the dynamics of the environment or decompose the control problem into several sub-processes [27]. Accordingly, the lack of accuracy and generalization ability of the methods can impose an adverse impact on their practical application. To achieve cost-efficient in terms of computation time, some rule-based approaches are studied [28], [29], [30], but the optimality can not be ensured. With the intent to implement adaptive control with real-time ability, more competent approaches are supposed to be developed.

However, regardless of the defects in the control strategies, there are also research gap in control objects. The application of automated driving system and vehicle-to-infrastructure (V2I) communication should not only enable the intelligent vehicles to make better decisions and enhance its individual functionality [31], but also improve the overall traffic performance, instead of sacrificing the mobility or energy consumption of other HDVs. Whereas the operation of CAVs may have a direct impact on other HDVs, and sometimes this influence would interfere with the normal running of those controlled by human drivers [32], leveraging CAVs in mixed traffic condition to avert negative impact and explicitly promote the performance of HDVs is crucial, and this topic with mixed traffic is rarely discussed in urban intersection scenario. Two notable exceptions are in [24] and [33], which tried to extend the beneficial effect of CAVs to platoon level, when platoon control is widely regarded as an effective way to generate smooth trajectories and implement cooperative control [14], [34], [35], [36], [37], [38]. The platoon developed in their studies is referred to “mixed platoon”, which is composed of a leading CAV and several following HDVs. There is no doubt that mixed platoon control creates a new stage in this field, but the aforementioned mixed platoon control frameworks are also based on a perspective of optimal control theory by expressly embodying cost functions, constraints, and solving algorithms. They still have limitations in solving high-order nonlinear optimal control problems, so the scenario must be simplified (e.g., using very simple car-following model for HDVs [33]) to obtain numerical results.

The Deep Reinforcement Learning (DRL) algorithms recently bring about new solutions for vehicular control and have the potential to tackle the above problems related to computation complexity and local optimum [39]. Benefiting from the strong fitting ability of deep neural networks (DNNs), the DRL technique has the potential to approximate the optimal control law. In reinforcement learning theory, an agent chooses actions according to the observed states so as to maximize its expected accumulated reward. For the general traffic control problems, the reward can be energy consumption, traffic delay, or the combination of relevant indicators. Compared to the aforementioned typical approaches, DRL can usually achieve low computation complexity after the training phase, and they are also promising to endow vehicles with “intelligence” and have them react to the environment adaptively.

Based on the DRL algorithms, a few frameworks have been proposed in recent years to control individual CAV in the proximity of signalized intersections. To name a few, Shi et al. [40] applied Q-learning to improve the fuel consumption efficiency of a connected vehicle at a signalized intersection. An improved version of Q-learning based control framework, integrating with a deep Q network (DQN), was developed by Mousa et al. [20]. However, as one of the value-based DQL algorithms, the DQN approach cannot deal with the problems with continuous action space. Therefore, they directly took discrete velocity change rate as the action space, which can result in a local optimum solution. With the application of policy-based algorithms, the aforementioned problems can be tackled. Guo et al. [27] and Zhou et al. [6] utilized a deep deterministic policy gradient (DDPG) algorithm to implement continuous longitudinal control of a CAV. Furthermore, based on DDPG algorithm, Qu et al. [41] demonstrated that the method could improve travel efficiency by reducing the negative impact of traffic oscillations. Wegner et al. [42] and Zhang et al. [43] had explored the energy-saving potential of electric CAV at urban signalized intersection by employing a twin-delayed deep deterministic policy gradient (TD3) agent, which is trained to control itself adaptively.

Although DRL algorithms can be powerful tools to control CAVs in urban scenarios, there are some drawbacks that do exist among those approaches. Firstly, they tended to use stepwise reward signals to facilitate the learning process, and the policy learned by the agent in this situation cannot be equivalent to global optimum. For example, the framework put forward by Guo et al. used stepwise travel distance to surrogate the total travel time of a CAV in an episode [27], while the value of travel time can only be acquired after the CAV crosses the intersection. Although the agent obtains distributed reward signal in each simulation step, the combination of the travel distance of all the steps is not tantamount to the total travel time. Intuitively, the agent may encounter a red light if it chooses the action in such a greedy way (i.e., aiming to maximize its stepwise travel time). Secondly, the previous DRL-based studies focus on the performance of a single CAV and ignore the interaction between CAV and HDVs. The CAVs may produce selfish policies in an “ego-efficient” way, which cannot guarantee satisfying performance of mixed platoons. Finally, it is known that algorithms like DDPG are highly sensitive to hyperparameter choices [44]. The traditional DRL approaches also suffer from the sample efficiency problem, especially for the delayed reward situation. Therefore, a more effective method should be built to promote the application of reinforcement learning.

To address the issues in existing DRL-based frameworks and extend the adaptive and real-time control approach to mixed platoon domain, this article develops a novel reinforcement learning framework for CAV control in a continuous action space. A delayed reward Markov Decision Process (MDP) is formulated to serve as the mathematical model of the longitudinal motion control process of the platoon.
state of the MDP considers the leading CAV and its following HDVs in a mixed platoon. With regard to the reward signal, this paper defines that it can only be obtained when the platoon crosses the junction, and simulation studies would manifest the benefits of this setting. An augmented random search (ARS) algorithm is proposed for the agent learning the control policy with the intent to deal with the delayed reward. The learning and evaluation of the framework are carried out in SUMO platform [45], which can demonstrate the effectiveness of the proposed method through microscopic traffic simulations.

The remainder of this paper is structured as follows. Section II introduces the preliminaries of DRL and the car-following model of HDVs. Section III provides the MDP formulation of the platoon-based control strategy. Section IV proposes the ARS algorithm to implement the self-learning mechanism. Section V reports a series of simulations carried out in the SUMO software and makes a comparison study with several state-of-the-art (SOTA) methods. Finally, some concluding remarks are presented in Section VI.

II. PRELIMINARY

A. Background of DRL

Reinforcement learning is an important branch of machine learning. The object to be controlled in reinforcement learning is seen as an agent, and the learning process can be promoted by a series of agent-environment interactions. One complete play of the agent interacting with the environment is called an episode. Generally, in step $t$ of an episode, an agent can observe a state $s_t$, which is usually a feedback by the environment. Then, the agent can conduct an action $a_t$ according to its policy $\pi(a_t|s_t)$. As a result, the agent can obtain a reward signal $r_t$, which is usually the representation of its optimization goal. Note that $r_t$ can be sparse when the reward can only be acquired in the terminal stage (i.e., with delayed rewards).

The process can be basically given by the MDP, which is defined as a five-tuple $(S, A, R, P, \gamma)$. $S, A,$ and $R$ denote the state space, the action space, and the reward space of the agent, respectively. For each timestep $t$, we have $s_t \in S$, $a_t \in A$, and $r_t \in R$. Meanwhile, $P$ specifies the state transition probability function: $S \times S \times A \rightarrow [0, \infty)$, which can emit the probability density of the next state $s_{t+1} \in S$ given the current state $s_t \in S$ and action $a_t \in A$. Moreover, $\gamma$ is a discount factor that measures the relative importance of the current reward and future reward. By interacting with the environment continuously, the agent aims to find an optimum policy that can maximize the expected sum of discounted future rewards $r'_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$. For any policy $\pi$, the state-action value function is $q^\pi(s, a) = E[r^\gamma_t|s_t = s, a_t = a, \pi]$, where $a_{t+k} \sim \pi(\cdot|s_{t+k})$ for all $a_{t+k}$ and $s_{t+k}$ for $k \in [1, \infty)$. Meanwhile, the state value function is $V^\pi(s) = E[r^\gamma_t|s_t = s]$. Taking an action space with finite number of actions as an example, we have $V^\pi(s) = \sum_{a \in A} \pi(a|s)q^\pi(s, a)$ according to the Bellman equation. Finally, let $\Pi$ represent the set of all possible policies, and the optimal policy $\pi^*$ can be defined as:

$$\pi^* \in \arg\max_{\pi \in \Pi} E[r^\gamma_t|\pi] \tag{1}$$

As a result, the agent can always select the optimal action following the optimal policy $\pi^*$.

The DRL technique makes use of deep learning to promote the traditional reinforcement learning approaches. Suppose the set of parameters is $w$ for a value network and $\theta$ for a policy network, we can parameterize the state-action value function by $Q(s, a|w) \approx q^\pi(s, a)$ for the value-based DRL algorithms, in order to approximate the optimal state-action value function $q^*(s, a)$. As for the policy-based DRL algorithms, the policy is directly parameterized as $\pi(s, a|\theta)$. Hereafter, the learning process will adjust the set of parameters $w$ or $\theta$ according to the “trial-and-error” mechanism to search for a suitable policy.

B. Car Following Model of HDVs

In this paper, we adopt the Intelligent Driver Model (IDM) to simulate the driving behavior of human drivers [46], whereas the model is widely used in microscopic traffic simulations [47], [48], [49]. The acceleration of the $n_{th}$ vehicle at time $t$ is related to its current velocity, time headway, and the velocity of the front vehicle. The mathematical form of IDM is defined by Equation 2 and Equation 3.

$$a_n(t) = \frac{dv_n(t)}{dt} = a_0(1 - \frac{v_n(t)}{v_0} - \frac{s_n^*(t)}{s_n(t)})^2 \tag{2}$$

$$s_n^*(t) = s_0 + T v_n(t) + \frac{v_n(t)\Delta v(t)}{\sqrt{2a_0 b}} \tag{3}$$

where, $a_0$ and $v_0$ are the maximal acceleration and the expected velocity of the vehicle in free flow; $v_n(t)$ denotes the velocity of vehicle $n$ at time $t$; $s_n^*(t)$ and $s_n(t)$ are the expected headway and the real headway between the vehicle and its front vehicle, respectively; $s_0$ represents the minimal headway; $T$ denotes the safe time headway; $\Delta v(t)$ denotes the velocity difference between the vehicle and its leading vehicle. Finally, $b$ denotes an acceptable comfort-related deceleration.

III. MARKOV DECISION PROCESS FOR THE PROBLEM

A. Problem Description

As shown in Figure 1, this study mainly focuses on a “1+n” form of the mixed platoon, consisting of one leading CAV and $n$ following HDVs. We call the electric CAV of the platoon
“ego CAV”, while the platoon led by the ego CAV is called “ego platoon”. Besides the ego platoon, there are some other HDVs travel on the road, and this can make the simulation get close to the real traffic situation. In order to simplify the problem without losing any generality, we make some assumptions as below:

1) With the support of V2I communication, the ego CAV can obtain the Signal Phase and Timing (SPaT) information of the fixed-timing traffic signal.

2) The ego CAV can get the position, velocity, and acceleration of itself by vehicular operation system.

3) The positions and velocities of the HDVs belonging to the ego platoon can be obtained by the ego CAV. At the same time, the ego CAV can also get these data of its leading vehicle if the position of the leading vehicle is in a predefined range. The assumption can be usually achieved by the vehicle-to-vehicle (V2V) communication, roadside units, or the perception ability of the CAV [24], [33].

Moreover, this paper takes the electric mixed platoons as research objects and make effort to optimize its electricity consumption. The starting point of electric vehicles is based on following reasons: (1) The electrification of vehicles shows great promise to achieve sustainable traffic development [50], as the carbon emissions and air pollution caused by the transportation system is still rising [51]. (2) Due to the regenerative braking energy of electric vehicles (EVs), the control of electric CAVs is more challenging than traditional gasoline cars. At the same time, the EVs show a higher potential of energy conversion efficiency at low load range [52]. In this case, the usage of electric mixed platoon would have realistic meaning for an electric and intelligent road transportation system in the near future.

Since the operation of the mixed platoon can be interrupted by other HDVs or traffic signals, the goal of the platoon is to reduce the overall delay and electric energy consumption. We basically study the longitudinal motion of the vehicles, because the unexpected lane changing may interfere with normal operations of other HDVs, especially in the vicinity of signalized intersections. Although the scenario presented in Figure 1 is a single-lane environment, the proposed framework can be conducted for CAVs in a decentralized fashion for multi-lane scenarios. Accordingly, an effective control law will generate a speed profile for the leading CAV and consider the motion of the subsequent HDVs. In this case, unnecessary stops and oscillations can be avoided to achieve the energy-saving goal.

B. Specification of the MDP

The elements in the MDP model, including \( S, A, \) and \( \mathbb{R} \) should be specifically defined to apply the DRL framework. For a “1+n” mixed platoon, the formation of the three components can be defined as follows.

1) State: State is the description of the agent in current situation. All of the vehicles within the mixed platoon should be taken into account as part of the state. Meanwhile, the potential leading vehicle is also considered, as the ego CAV should keep a safe gap and estimate the traffic ahead. With the intent to reduce unnecessary stop-and-go operation, the agent also needs the SPaT information of the first downstream traffic signal. Therefore, let \( s^C_t, s^H_t, s^L_t, \) and \( s^S_t \) be the CAV-related part, the HDVs-related part, the leading-vehicle-related part, and the signal-related part of the state, the state can be expressed as:

\[
s_t = (s^C_t, s^H_t, s^L_t, s^S_t)^T
\]

In this case, the details of each part of the state are shown as follows:

\[
s^C_t = (x(t), v(t))
\]

\[
s^H_t = (x_1(t), v_1(t), x_2(t), v_2(t), \ldots, x_n(t), v_n(t))
\]

\[
s^L_t = (\Delta x(t), \Delta v(t), \Delta a(t))
\]

\[
s^S_t = (RT(t), E_i(t))
\]

where, \( x(t) \) is the distance between the ego CAV and the stop line of the first downstream intersection at time \( t; v(t) \) is the velocity of the ego CAV at time \( t; x_i(t) \) is the distance between \( i_{th} \) HDV in the platoon to the stop line for \( i \) from 1 to \( n \), and \( v_i \) is the speed of the corresponding HDV. For the third item in Equation 4, we set a predefined threshold \( \chi_s \) to judge if there is a leading vehicle in front of the platoon. Let \( L \) be the index of the potential leading vehicle, we set a boolean variable \( \delta \) to identify the existence of the potential preceding vehicle:

\[
\delta = \begin{cases} 
\text{True}, & \text{if } x(t) - x_L(t) \leq \chi_s; \\
\text{False}, & \text{otherwise}.
\end{cases}
\]

The calculation of \( \Delta d, \Delta v, \) and \( \Delta a \) according to the value \( \delta \) are expressed as:

\[
\Delta d = \begin{cases} 
x(t) - x_L(t), & \text{if } \delta; \\
\chi_s, & \text{otherwise.}
\end{cases}
\]

\[
\Delta v = \begin{cases} 
v_L(t) - v(t), & \text{if } \delta; \\
\chi_v, & \text{otherwise.}
\end{cases}
\]

\[
\Delta a = \begin{cases} 
a_L(t) - a(t), & \text{if } \delta; \\
\chi_a, & \text{otherwise.}
\end{cases}
\]

where, \( \chi_v \) and \( \chi_a \) are the predefined default value of the two variables. In this paper, \( \chi_s \) is set to 500m, which means that the vehicle 500 meters away from the ego CAV will not affect its driving. Moreover, \( \chi_v \) and \( \chi_a \) are set to 13.88m/s and 7.5m/s².

As for the signal-related state, \( RT(t) \) in Equation 8 denotes the remaining time of the current phase for the first downstream traffic signal, and this value can be retrieved in a communication environment. Furthermore, \( E_i(t) \) denotes the one-hot encoding of the current phase of the traffic signal. The encoding process is illustrated in Figure 2. The phase diagram shows the signal phase used in this study, and yellow light is added between two adjacent phases. If one phase is activated by the traffic signal (i.e., the phase with red box in Figure 2), the corresponding element in the encoding vector will be set to 1, while other elements are all set to 0.

2) Action: Considering the maneuverability of the system, the action should be set to acceleration changing of the ego
CAV. Hence, the action space is constrained by the dynamics of the vehicle: $a_t \in [\omega_{\min}, \omega_{\max}]$, where $\omega_{\min}$ and $\omega_{\max}$ are the maximal deceleration and acceleration of the vehicle. However, it is problematic to take the acceleration as the action directly. On the one hand, irrational accelerations will lead to unsafe operations like rear-end accidents, and this kind of phenomenon can occur very often during the training process; on the other hand, the speed of the vehicle may exceed the road speed limit with the effect of the action. Consequently, the modified action is stipulated as:

\[ a_t = \min(\tilde{a}_t, a^{IDM}(t)) \]  

(13)

where, $\tilde{a}_t$ denotes the original acceleration value output by the DRL algorithm; $a^{IDM}(t)$ is the acceleration calculated by IDM. Equation 13 makes the acceleration of the ego CAV be kept in a safe range.

The velocity of the ego CAV in timestep $t$ is defined as below to meet the speed limit $V_{\text{max}}$:

\[ v(t) = \max(\min(V_{\text{max}}, v(t-1) + a_t), 0) \]  

(14)

where, $v(t-1)$ is the speed of the ego CAV in last timestep.

3) Reward: The optimization goal, including total energy consumption and travel delay, is supposed to be reflected in reward function. However, the travel delay can only be calculated when the vehicles have crossed the signalized intersection. Distributing the delayed reward to each step in an episode is known as the temporal Credit Assignment Problem (CAP) [53], which is hard to deal with. The previous studies took stepwise energy consumption and travel distance to serve as a distributed proxy of the two parts of the delayed reward [27], [43]. Nevertheless, the cumulative travel distance cannot indicate the delay of the vehicles accurately. A more intuitive way is using the delayed reward, directly reflecting the optimization goal. In this case, the reward is non-Markovian. In this study, we will show that our algorithm can commendably solve the CAP and train the agent. The reward function is defined as:

\[ r_t = \begin{cases} 
\sum_{i=0}^{n} -\omega_1 e_i - \omega_2 d_i, & \text{if } t = t_{\text{final}} \\
0, & \text{otherwise.} 
\end{cases} \]  

(15)

where, $e_i$ denotes the total energy consumption of vehicle $i$; $d_i$ denotes the delay of vehicle $i$. Note that the vehicle with $i = 0$ here represents the ego CAV. Meanwhile, $\omega_1$ and $\omega_2$ are weighting parameters that measure the relative importance of mobility indicator and energy indicator. Finally, $t_{\text{final}}$ specifies the finale of an episode. It is the time when the last HDV in the ego platoon crosses the intersection.

In Equation 15, $e_i$ is calculated by a series of records in the whole episode. This study utilizes an energy model with energy brake-recovery mechanism embedded in SUMO to calculate the instantaneous electric consumption [54]. Note that any other energy model can be used owing to the generality of the proposed framework, even if a simple indicator that derived from the difference of the battery. The instantaneous energy consumption is calculated for each vehicle within the platoon in each step. Finally, the total energy consumption $e_i$ is calculated in a cumulative way when vehicle $i$ enters the intersection. Similarly, $d_i$ is expressed as:

\[ d_i = t_f^i - t_0^i = \frac{L}{V_{\text{max}}} \]  

(16)

where, $t_f^i$ is the time when vehicle $i$ crosses the stop line in front of the traffic signal; $t_0^i$ is the initial time that vehicle $i$ enters the road network; $L$ denotes the length of the entrance lane where the platoon locates.

IV. AUGMENTED RANDOM SEARCH

The purpose of the algorithm is to directly search a policy in continuous action space, while the obtained policy can approximate the optimal policy $\pi^*$ in Equation 1. As the transition dynamics is unknown in most cases, model-free reinforcement learning algorithms are usually deployed. It is pointed out that many model-free DRL methods need too much data to search a proper optimization direction, and they can be very complicated without robustness [44]. Considering the practicability, we develop an ARS algorithm in this paper to search the policy in a black-box way. Being compared with the gradient-based DRL methods, the black-box optimization approach can achieve sample efficiency and have an advantage in cases with long action sequences and delayed reward [55].

In the context of DRL, the policy is usually parameterized by a set of parameters $\theta$, which is supposed to be trained in training process. The ARS utilized a linear policy with parameter set $\theta$ instead of DNNs like most DRL algorithms. Note that throughout the rest of the paper we use $\pi_{\theta}$ to denote the ARS-based policy with parameter set $\theta$. Let the dimension of the state in Equation 4 be $p$. The parameter set $\theta$ is a $p \times u$ matrix, while the dimension of action is represented as $u$.

The basic idea of ARS is to randomly adds some tiny variables to the parameter $\theta$ along with the negative value of the corresponding value, which aims to sample some directions in parameter space. Let $\nu$ be a positive real number that denotes the standard deviation of the exploration noise; $\mu$ denotes a matrix with the same dimension as $\theta$, in which the elements obey the zero mean Gaussian distribution; $\xi_1$ and $\xi_2$ denote the variables that encode the randomness of the environment (i.e., random initial states). After the perturbation, the agent applies actions based on policy $\pi_{\theta+\nu\mu}$ and $\pi_{\theta-\nu\mu}$ and collect...
Algorithm 1 ARS for Mixed Platoon Control

Hyperparameters: step-size $\alpha$, number of directions sampled per iteration $K$, noise $\nu$, number of top-performing directions to use ($b < K$)

Initialize: $\theta_0 = 0 \in \mathbb{R}^{p \times n}$, $\sigma_0 = 0 \in \mathbb{R}^n$, $\Sigma_0 = I_n \in \mathbb{R}^{n \times n}$, $j = 0$

1: while end condition not satisfied do
2: Sample $\mu_1, \mu_2, \ldots, \mu_K$ in $\mathbb{R}^{p \times n}$ with i.i.d standard normal entries.
3: Collect $2K$ episodes from $\pi_{j,k,+}$ and $\pi_{j,k,-}$ of horizon $H$ and their corresponding rewards using the $2K$ policies in SUMO:
   
   
   
   
4: Sort the directions $\mu_k$ according to $\max r(\pi_{j,k,+}, \pi_{j,k,-})$. Let $\mu_{(k)}$ be the $k$-th largest direction, and by $\pi_{j,(k),+}$, $\pi_{j,(k),-}$ the corresponding policies.
5: Update $\theta$ by step ($\epsilon_K$ denotes the standard deviation of the $2b$ rewards): $\theta_{j+1} = \theta_j + \frac{\alpha}{\sqrt{b}} \sum_{k=1}^{b}[r(\pi_{j,(k),+}) - r(\pi_{j,(k),-})] \mu_{(k)}$
6: Set $\sigma_{j+1}, \Sigma_{j+1}$ to be the mean and covariance value of the $2KH(j + 1)$ states encountered from the start of training.
7: $j \leftarrow j + 1$
8: end while

The purpose of normalization is to eliminate the influence of dimensional inconsistency of different elements in state vectors. For the parametric linear policy, it can promote non-isotropic explorations in the parameter space. For a perturbation direction $\mu$, there is:

$$ (\theta + \nu \mu) \text{diag}(\Sigma)^{-\frac{1}{2}} (x - \sigma) = (\hat{\theta} + \nu \mu \text{diag}(\Sigma)^{-\frac{1}{2}})(x - \sigma) $$

where, $\hat{\theta} = \theta \text{diag}(\Sigma)^{-\frac{1}{2}}$

3) Using top-performing directions: The perturbation direction $\mu$ is weighted by the difference of two opposed rewards $r(\pi_{j,k,+})$ and $r(\pi_{j,k,-})$ (see line 3 in Algorithm 1). Without this trick, the update steps push $\theta$ in the direction of $\mu_k$. However, using top-performing directions can order decreasingly the directions $\mu_k$ by $\max[r(\pi_{j,k,+}), r(\pi_{j,k,-})]$. Finally, only the top $b$ directions are utilized to update the policy parameters (see line 5 in Algorithm 1).

During the iterations of training, only the total reward of an episode is used to evaluate the performance of a series of actions, so ARS can deal with maximally sparse and delayed rewards and avoid the difficulties produced by CAP. The feature makes it suitable to solve the platoon control problems with delayed reward configurations. Without the training of DNN, ARS can save much inference time with a higher sample efficiency, and it is promising to deploy such a computation efficient framework in real world.

V. SIMULATION ANALYSIS

As one of the most popular open-source traffic simulator, SUMO allows the modeling of the microscopic behavior of vehicles and pedestrians. The value of simulation in SUMO can be retrieved and changed through the “TraCI” interface by other program languages. In this study, a signalized intersection is built in SUMO environment, and the scenario is similar to that shown in Figure 1.

A. Simulation Settings

The signal phases are shown in Figure 2. As a fixed-timing traffic signal, the last time for each phase is 30s, while a 3s yellow phase is inserted for every phase changing. Under the premise of comprehensive consideration of reality and generality, the other related parameters for simulation configuration are presented in Table 1. Before the learning process of each episode, a pre-loading procedure is carried out. More precisely, the traffic volume is loaded for time $t_p$ (sampled from a uniform distribution) before the ego platoon enters the road, aiming at generating more dynamic
### TABLE I

**THE PARAMETER SETTING OF SIMULATIONS**

| Item                                           | Value   | Unit |
|------------------------------------------------|---------|------|
| Maximum acceleration of vehicles $a_{\text{max}}$ | 3.0     | $m/s^2$ |
| Minimum acceleration of vehicles $a_{\text{min}}$  | -4.5    | $m/s^2$ |
| Road speed limit $V_{\text{max}}$               | 13.88   | $m/s$  |
| The length of the lane $L$                      | 500     | $m$    |
| Safe time headway $T$ in IDM                   | 1       | $s$    |
| Acceptable comfort-related deceleration $b$ in IDM | -2.8   | $m/s^2$ |
| Hourly traffic volume                           | 400     | $veh$  |
| Pre-loading time $t_p$                         | $U(180, 220)$  | $s$    |

traffic scenarios. Meanwhile, after a series of simulations, the hyperparameters of ARS are tuned manually. The standard deviation of parameter noise $\nu$ is set to 0.2; the number of directions sampled per iteration is set to 32. Note that the weighting parameters are set as: $\omega_1 = 6$, $\omega_2 = 1$ if no special explanation is provided. The sensitivity analysis of the two parameters are presented in the subsequent subsection.

### B. Training Performance

Firstly, with the intent to show the robustness of the ARS approach, we conduct 7 rounds of independent training with different random seeds, whereas the random search approach may be sensitive to its random seed. Figure 4 illustrates the training results. The reward of each episode is collected in the final step, when the platoon cross the stop line of the intersection. Owing to the noise rewards for different episode, a moving average function is applied to smooth the tendency of the mean values: $R_k \leftarrow 0.8R_{k-1} + 0.2R_k$, where $k$ denotes the $k$-th episode. According to the figure, although the fluctuation range of each round can be different, they can all converge to the same result with about $-1250$ reward. Therefore, the stability and robustness of the ARS-based framework is proved.

Secondly, we make a comparison study with other DRL-based SOTA methods, including Proximal Policy Optimization (PPO), DDPG, and DQN. Note that the action space of the DQN is set to a 16-length vector, which varies from $a_{\text{min}}$ to $a_{\text{max}}$ with the step of $0.5m/s^2$. The DQN algorithm utilizes a neural network with two hidden layers to approximate value function, and the number of neuron units of the two hidden layers are 256 and 64, respectively. DDPG and PPO are all based on an actor-critic structure, with a two-layer actor network and a two-layer critic network. The neuron number of the two algorithms are the same with 300 and 400 for both actor and critic network. For each algorithm, other hyperparameters are tuned manually through a series of simulations (refer to appendix for main parameter settings), and the results from seven independent training are aggregated to obtain the final result. Taking the scenario with “1+3” mixed platoon as an example, the training processes are shown in Figure 5. It can be seen that it is hard to train the agent for the other three SOTA algorithms with the delayed reward cases. However, the reward of the ARS agent can converge to a higher value compared with the other approaches.

### C. Exploring the Impact of Reward Configuration

To investigate the influence of reward settings, we compare the cases with episodic reward (ER) and distributed reward (DR) settings. In DR setting, the reward is calculated step by step according to the stepwise sum of energy consumption and travel distance of the platoon. Accordingly, we train the ARS agent five times, and collect five episodes of reward for each trained agent. More precisely, 25 groups of simulations are carried out to record the data for each reward setting. The results are presented in Figure 6. Whether in terms of travel delay or electric consumption, the ER setting can outperform the DR setting. The agents with DR show high variance with respect of energy consumption indicator, and this illuminates the instability of this kind of configurations.

Similar studies can be conducted for PPO algorithm. Table II shows the mean value of the indicators, deriving from 25 episodes of simulations. IDM is introduced to serve as a baseline. In this case, the ego CAV is controlled by the IDM, which can represent the general car following scenario.
Table II demonstrates that the ER-based ARS can reduce energy consumption to the maximum extent. The DR-based PPO has a similar performance with the ER-based ARS in terms of total delay. However, ARS can reduce the electric energy consumption by 52.95% compared with the DR-based PPO for a “1+3” platoon on average. Inevitably, the optimization on energy will lead to the sacrifice of mobility [27]. With the setting of \( \omega_1 = 6 \) and \( \omega_2 = 1 \), 82.89% energy is saved by the adaptive control implemented by ARS algorithm compared with IDM. The agent in this case behaves toward an extreme energy efficiency direction. Nevertheless, we will conduct a sensitivity analysis in the following subsection. The analysis can reveal that the agent can reduce energy consumption with almost no sacrifice of mobility.

### D. Performance for Different Platoon Size

Figure 7 shows the smoothed training curve for the scenarios with different platoon sizes. It can be concluded that the cases with different platoon sizes can be optimized, and the results can converge to different values. The more HDVs are considered by the agent, the higher optimization rate can be observed. As a result, the framework has the potential to normally extend to multi-vehicle systems.

More specifically, we use the trained ARS agents with different platoon size configurations to run the simulations for evaluation. We make comparison study with several other approaches and offer the single-vehicle-based statistical results for different platoon size configurations.

In comparative study, since the traditional learning-based control (i.e., DDPG, PPO, and DQN) has limitations in this task, some SOTA rule-based and optimization-based strategies are implemented for a more exhaustive comparison between ARS and traditional methods, including: (1) Green Light Optimal Speed Advisory (GLOSA) [56]: GLOSA belongs to the rule-based model, which provides CAVs with speed guidance in an “ego-efficient” way. It can usually control the CAV to cross the intersection within green phase in a free flow condition. (2) Trajectory optimization based on Particle Swarm Optimization (TO-PSO) [57]: TO-PSO is built for mixed platoon control through a two-layer framework. The first layer estimates the space of CAV arrival time, while the second layer formulates a trajectory optimization model. The whole process is solved by a Particle Swarm Optimization algorithm to reduce time complexity. (3) MPC [24]: The optimal control model considers both CAV and HDVs in its state variables and system dynamics. The control problem is transformed into a nonlinear programming model through Gauss pseudospectral method and is solved numerically. The results are also collected from 10 independent simulations, which is illustrated in Figure 8. It can be seen that the energy
consumption and traffic delay can maintain a stable level under ARS control. The consumed energy and time decline slightly when rule-based GLOSA system is employed, but this change is limited by its “ego-efficient” feature, which cannot takes the following HDVs into account. Although the traffic delay increases compared to GLOSA approach, it is just the result of extremely energy-saving setting due to the large ratio of weighting parameters $\omega_1$ and $\omega_2$, and we will show that the sacrificed travel delay can be reduced to approximately zero by regulating the parameters. Similar patterns to GLOSA can be found for TO-PSO method with a relatively low energy efficiency and high travel delay. Although the SOTA MPC approach presents similar performance with ARS, there is still a certain gap between MPC and the random search method. In addition, ARS is more stable for different platoon size settings, which means that it can achieve a stronger generalization. The optimization-based approaches still need iterations in each time step to converge and obtain a satisfying control input, which requires better computing hardware and more computing time and brings difficulties to practical applications. Therefore, they are not practical enough even without looking at control performance.

For each platoon size configuration, the trajectories of the vehicles in the ego platoon are collected. We randomly sample several trajectories and draw the figures. The results are shown in Figure 9. The color depth reflects the speed of the vehicles, while the horizontal line represents the phase of the traffic signal. Meanwhile, we implement an IDM-based study to make a comparison, and the sampled trajectories are also provided in Figure 9. According to the figure, the ego platoon can cross the signalized intersection without any stops when the ego CAV is controlled by ARS. Thus, the unnecessary stop and rapid acceleration/deceleration can be avoided to promote energy conservation. In addition, the ego CAV can consider the crossing of more HDVs as the size of the platoon increases. When the number of HDVs exceeds 4, the platoon controlled purely by IDM can be divided so that some of the vehicles in the platoon cannot cross the intersection with the leading vehicles during the same phase. The ARS agents can adjust its velocity to relatively low value to fit the phase change and guarantee the effective operation of subsequent HDVs, while the CAVs controlled by IDM can only speed up if there is no interruption. This also illuminated that only based on the appropriate control methods can the comprehensive benefits of the CAVs be brought into traffic.

E. The Impact of Weighting Parameters

The weighting parameters determine the optimization direction of the algorithm. Exploring the impact of the weighting parameters is valuable for understanding the effect of the ER-based reward signal. In particular, the impact of $\omega_1$ and
Fig. 10. The trajectories of the mixed platoon with different weighting parameter settings.

| Ratio ($\omega_1/\omega_2$) | Delay per vehicle (s) | Energy consumption per vehicle (Wh) | Improvement |
|-----------------------------|-----------------------|-----------------------------------|-------------|
|                             | PPO with DR | DDPG | ARS | Imp. ↑ | PPO with DR | DDPG | ARS | Imp. ↑ |
| 1/1                         | 116.24 | 61.78 | 133.84 | -130.84% | 99.47 | 89.78 | 59.30 | 61.27% |
| 1/2                         | 107.44 | 162.51 | 103.24 | -78.07% | 126.26 | 78.27 | 92.99 | 39.27% |
| 1/3                         | 78.71 | 81.91 | 96.38 | -66.22% | 106.04 | 106.77 | 79.54 | 48.05% |
| 1/4                         | 96.77 | 81.98 | 77.04 | -32.88% | 124.78 | 111.74 | 83.48 | 45.47% |
| 1/5                         | 124.84 | 162.24 | 61.04 | -5.28% | 106.10 | 77.70 | 81.58 | 46.72% |
| 1/6                         | 98.64 | 161.71 | 55.11 | 4.95% | 88.01 | 77.94 | 70.98 | 53.64% |
| 2/1                         | 72.78 | 151.91 | 164.64 | -183.97% | 113.87 | 77.81 | 29.03 | 81.04% |
| 3/1                         | 67.78 | 79.64 | 167.71 | -189.26% | 95.69 | 111.56 | 31.55 | 79.40% |
| 4/1                         | 90.91 | 165.38 | 163.78 | -182.47% | 112.64 | 77.79 | 29.84 | 80.51% |
| 5/1                         | 98.91 | 159.24 | 163.41 | -181.83% | 102.88 | 78.39 | 29.73 | 80.58% |
| 6/1                         | 154.18 | 162.04 | 165.78 | -185.92% | 48.40 | 69.68 | 26.79 | 82.51% |

\(\omega_2\) mainly originates from the ratio (i.e., \(\omega_1/\omega_2\)) of the two values. Therefore, we fix \(\omega_2\) to 1 and change \(\omega_1\) from 1 to 6, and then fix \(\omega_1\) to 1 with \(\omega_2\) changing from 1 to 6. A “1+3” mixed platoon scenario is still taken as an example to observe the impact of the ratio, and the results are shown in Figure 11. According to the figure, we can see that the delay of vehicles reduces rapidly with the increase of \(\omega_2\) when \(\omega_1 < \omega_2\). The policies learned by the agent can reduce both delay and energy consumption in these cases. When we set \(\omega_1 > \omega_2\), the energy consumption can be reduced significantly. The policies in these cases can serve as economic driving strategies to maximize energy efficiency.

Simulations for other DRL algorithms with different weighting parameter settings are also carried out to make more comprehensive comparison studies, and the results are collected in Table III. It can be found that the performance of the proposed ARS-based control varies regularly with the change of weighting parameters, while the same outcome cannot be achieved by the other two DRL algorithms. In other words, we may not get the expected results when we try to build a specific strategy (i.e., set different importance for energy efficiency and mobility) if we use these gradient-based algorithms. This finding further enhances the flexibility and applicability of the framework with delay reward when considering regulating the relative importance between mobility and energy efficiency. Moreover, the ARS algorithm can achieve the optimal performance in terms of both travel delay and energy consumption. The significant decline of consumed electricity demonstrates that our method possess tremendous potential for the mixed platoon control task.

More precisely, compared with the basic IDM car-following behavior, the electricity consumption is reduced by 39.27% to 82.51% with different weighting parameter settings. If we set \(\omega_1 = 1\) with \(\omega_2 = 6\), the energy can be saved by 53.64% with approximately the same performance in terms of delay. This result achieves SOTA performance when it is difficult to have both energy consumption and travel delay decline [27], [42].

We sample three scenarios with different parameter settings and plot the trajectories of the platoon. As shown in Figure 10, the leading CAV shows different driving behaviors in the three scenarios. The CAV tends to formulate a platoon and let the platoon decelerate smoothly and recover energy when \(\omega_1/\omega_2\) is not that small. When \(\omega_2\) is set to 6, the CAV chooses to cross the intersection with maximum velocity after forming a platoon. There is one major difference between Figure 10 and Figure 9, that is, if vehicular mobility is emphasized (i.e., \(\omega_2\) is
relatively big), the platoon will cross the intersection within the next green phase, otherwise, it will run more smoothly with relatively slow speed to cross the intersection within the second green phase to save energy.

VI. CONCLUSION

In this paper, we propose a reinforcement learning framework to control the mixed platoon composed of CAVs and HDVs at a signalized intersection. By designing a novel state representation and reward function, the approach can be extended to the platoons with different platoon size. ARS is implemented to overcome the challenge caused by episodic reward, which is proved can outperform the distributed reward configuration for the utilized algorithm. Analysis and simulation results validate that ARS is capable of controlling the ego CAV and make the platoon cross the intersection without any stops. Meanwhile, great energy-efficient performance can be achieved, so we recommend the method as an economic driving strategy in practice. Being compared with several DRL algorithms, the proposed method gives a much higher reward with a simple architecture. Moreover, the simple random search approach outperform several SOTA frameworks based on predefined rules or optimization models. It is demonstrated that the proposed algorithm shows feasible control outcomes with different parameter settings, and the flexibility further enhance its applicability, not to mention its real-time feature.

It should be noted that the strategy put forward in this paper is still feasible with multi-intersection scenario by taking the SPaT information of the first downstream traffic signal as part of the state in succession. However, we only study the control of a single agent, while multi-agent cooperation may bring about more return with the improvement of the whole regional traffic flow. A collaboration way can be introduced with the support of vehicle-to-vehicle communication in this context.

As for the future research, firstly, the longitudinal motion of vehicles can be controlled by setting the acceleration in support of vehicle-to-vehicle communication in this context. More comprehensive studies can start from the combination of longitudinal and lateral control in order to further tap the advantages of CAVs. By designing proper strategy to incorporate car-following and lane-changing motion, the cooperative operation of CAVs from multi-lane traffic environment may have a profound influence on the overall mixed traffic performance. Secondly, the influence of the traffic signal timing scheme is not explored in this paper, and it can be discussed specifically. Thirdly, the difference between traditional gasoline vehicles and electric vehicles can be discussed for the DRL-based adaptive control. Finally, it is valuable to study the impact range of the ego CAV, which is determined by its sensing ability or communication ability, so as to make the model more practical. With the development of ITS, more reliable control methods will be implemented to create a sustainable and efficient urban traffic environment.

APPENDIX

See Table IV.

REFERENCES

[1] D. J. Fagnant and K. Kockelman, “Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations,” Transp. Res. A, Policy Pract., vol. 77, pp. 167–181, Jul. 2015.
[2] J. Rios-Torres and A. A. Malikopoulos, “A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps,” IEEE Trans. Intell. Transp. Syst., vol. 18, no. 5, pp. 1066–1077, May 2017.
[3] D. Bevly et al., “Lane change and merge maneuvers for connected and automated vehicles: A survey,” IEEE Trans. Intell. Vehicles, vol. 1, no. 1, pp. 105–120, Mar. 2016.
[4] S. I. Guler, M. Menendez, and L. Meier, “Using connected vehicle technology to improve the efficiency of intersections,” Transp. Res. C, Emerg. Technol., vol. 46, pp. 121–131, Sep. 2014.
[5] C. Dong, H. Wang, Y. Li, W. Wang, and Z. Zhang, “Route control strategies for autonomous vehicles exiting to off-ramps,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 7, pp. 3104–3116, Jul. 2020.
[6] M. Zhou, Y. Yu, and X. Qu, “Development of an efficient driving strategy for connected and automated vehicles at signalized intersections: A reinforcement learning approach,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 1, pp. 433–443, Jan. 2020.
[7] J. Guanetti, Y. Kim, and F. Borrelli, “Control of connected and automated vehicles: State of the art and future challenges,” Annu. Rev. Control, vol. 45, pp. 18–40, Jan. 2018.
[8] B. Peng, M. F. Keskin, B. Kulesak, and H. Wymeersch, “Connected autonomous vehicles for improving mixed traffic efficiency in unsignalized intersections with deep reinforcement learning,” Commun. Transp. Res., vol. 1, Dec. 2021, Art. no. 100017.
[52] H. Kato, R. Ando, Y. Kondo, T. Suzuki, K. Matsushashi, and S. Kobayashi, “The eco-driving effect of electric vehicles compared to conventional gasoline vehicles,” AIMS Energy, vol. 4, no. 6, pp. 804–816, 2016.
[53] M. Minsky, “Steps toward artificial intelligence,” in Proc. IRE, vol. 49, no. 1, pp. 8–30, Jan. 1961.
[54] T. Kurczveil, P. A. López, and E. Schnieder, “Implementation of an energy model and a charging infrastructure in SUMO,” in Simulation of Urban Mobility, M. Behrisch, D. Krajzewicz, and M. Weber, Eds. Berlin, Germany: Springer, 2014, pp. 33–43.
[55] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, “Evolution strategies as a scalable alternative to reinforcement learning,” 2017, arXiv:1703.03864.
[56] A. Stevanovic, J. Stevanovic, and C. Kergaye, “Green light optimized speed advisory systems: Impact of signal phasing information accuracy,” Transp. Res. Rec., vol. 2390, no. 1, pp. 53–59, 2013.
[57] M. A. Guney, R. Guo, and P. Tiwari, “Cooperative maneuver planning for mixed traffic at signalized intersections,” in Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC), Sep. 2021, pp. 1171–1178.

Xia Jiang received the B.E. degree in traffic and transportation engineering from Central South University, Changsha, China. He is currently pursuing the master’s degree with the School of Transportation, Southeast University, Nanjing, China. His research interests include the cooperative control of connected and automated vehicles, reinforcement learning, and cooperative vehicle-infrastructure systems.

Jian Zhang (Member, IEEE) received the Ph.D. degree from Southeast University, Nanjing, China, in 2011. From 2009 to 2010, he was a Visiting Student at the University of Wisconsin–Madison, Madison, WI, USA. He is currently a Professor at the School of Transportation, Southeast University. He has authored or coauthored over 70 papers in international journals and conferences. His research interests include intelligent transportation systems, connected vehicles, and the public transportation systems. He is a member of ASCE.

Xiaoyu Shi received the B.E. degree in traffic engineering from Southeast University, Nanjing, China. She is currently pursuing the master’s degree with the School of Transportation, Southeast University, and the Department of Civil Engineering, Monash University, Melbourne, Australia. Her research interests include eco-driving control of automated vehicles.

Jian Cheng received the B.E. degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2002. He has long been engaged in the research of road intelligent transportation system. He is currently the Vice President of Nanjing Les Information Technology Company Ltd.