

Reward-Driven Automated Curriculum Learning for Interaction-Aware Self-Driving at Unsignalized Intersections

Zengqi Peng, Xiao Zhou, Lei Zheng, Yubin Wang, and Jun Ma

Abstract—In this work, we present a reward-driven automated curriculum reinforcement learning approach for interaction-aware self-driving at unsignalized intersections, taking into account the uncertainties associated with surrounding vehicles (SVs). These uncertainties encompass the uncertainty of SVs' driving intention and also the quantity of SVs. To deal with this problem, the curriculum set is specifically designed to accommodate a progressively increasing number of SVs. By implementing an automated curriculum selection mechanism, the importance weights are rationally allocated across various curricula, thereby facilitating improved sample efficiency and training outcomes. Furthermore, the reward function is meticulously designed to guide the agent towards effective policy exploration. Thus the proposed framework could proactively address the above uncertainties at unsignalized intersections by employing the automated curriculum learning technique that progressively increases task difficulty, and this ensures safe self-driving through effective interaction with SVs. Comparative experiments are conducted in *Highway.Env*, and the results indicate that our approach achieves the highest task success rate, attains strong robustness to initialization parameters of the curriculum selection module, and exhibits superior adaptability to diverse situational configurations at unsignalized intersections. Furthermore, the effectiveness of the proposed method is validated using the high-fidelity CARLA simulator.

I. INTRODUCTION

In recent years, autonomous driving technologies have achieved notable advancements and remarkable strides in both academic and industrial domains [1]–[3]. Nevertheless, self-driving tasks in dense and interaction-heavy scenarios remain an open challenge, primarily stemming from the lack of precise driving intention of surrounding vehicles (SVs) [4], [5]. Apparently, inaccurate assessments regarding the driving intention of SVs could significantly hinder safe motion planning and potentially lead to traffic accidents [6]. The situation becomes even more severe when it comes to unsignalized intersections, where the ego vehicle (EV)

This work was supported in part by the National Natural Science Foundation of China under Grant 62303390; in part by the Guangzhou-HKUST(GZ) Joint Funding Scheme under Grants 2023A03J0148 and 2024A03J0618; and in part by the Project of Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone under Grant HZQB-KCZYB-2020083. (*Corresponding Author: Jun Ma.*)

Zengqi Peng, Xiao Zhou, Lei Zheng, and Yubin Wang are with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China (e-mail: zpeng940@connect.ust.hk; xzhou910@connect.hkust-gz.edu.cn; lzhenl35@connect.ust.hk; ywang575@connect.hkust-gz.edu.cn).

Jun Ma is with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China, also with the Division of Emerging Interdisciplinary Areas, The Hong Kong University of Science and Technology, Hong Kong SAR, China, and also with HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Futian, Shenzhen, China (e-mail: jun.ma@ust.hk).

needs to coordinate simultaneously with multiple vehicles approaching from various directions [7]. In general, these intersections exhibit unpredictable traffic patterns, and uncertainties in the driving behavior of SVs pose serious safety concerns during the interaction process, making it challenging for autonomous vehicles to anticipate and respond appropriately to potential hazards. In this sense, unsignalized intersections typically require complex decision-making and situational awareness.

Rule-based and optimization-based methods are extensively studied towards the decision-making and planning of autonomous vehicles at intersections. Rule-based methods can prioritize driving safety and avoid potential collisions among road users by designing a set of rules in accordance with road traffic safety regulations [8]. However, rule-based methods struggle to account for all possible traffic scenarios and tend to be overly cautious, leading to an excessive level of conservatism [9]. On the other hand, optimization-based methods enable the EV to navigate at intersections by minimizing a given objective function subject to well-designed constraints [10], [11]. However, these methods are computationally expensive and time-consuming, and they may not always be able to account for unexpected or rapidly changing traffic scenarios, making them less adaptable in certain situations. Additionally, optimization-based methods are prone to getting stuck in the local optima when it comes to complex situations (e.g., scenarios involving SVs with interactive driving behaviors), which could lead to suboptimal or even unsafe decisions being made by the autonomous vehicle.

Reinforcement learning (RL) has recently demonstrated tremendous potential in autonomous driving [12]–[15]. Nevertheless, one significant limitation of RL-based approaches is the low sample efficiency when dealing with complex driving scenarios [16]. The agent requires a significant amount of time exploring the environment due to the lack of appropriate guidance. Furthermore, training directly on the data from complex target tasks can potentially lead the network to converge to a locally optimal policy, resulting in poor driving performance and adaptivity capability. One promising way to alleviate the above problem is curriculum learning [17]–[19]. In [20], a curriculum reinforcement learning approach is proposed for self-driving tasks. However, as the timing of curricular transitions is manually set, the quality of training outcomes will be highly dependent on the expert experience. Besides, the parameters governing the timing of curriculum transitions exhibit substantial variation across different scenarios.

To address the above problem, a series of automated curriculum learning methods have been proposed [21]–[23]. In [21], two learning progress signals are designed for the automated transition of curriculum for neural networks. Particularly, the impact of two signals on the training process is compared across different curricula. An action-value-based RL approach is proposed in [22] to address the autonomous driving task at single-lane crossroads with stop signs by modeling different starting positions of the EV as a curriculum set. Nonetheless, the update of curriculum importance is solely related to their respective rewards, potentially leading to curricula with high future rewards being overlooked. In [23], a state drop-out framework is designed to train agents for traversing the unsignalized intersection. The agent can obtain a fine-tuned policy in the final phase by dropping the future state information during the training process. However, SVs are set as non-interactive, and their future trajectories are assumed to be accessible to the EV in [22] and [23]. These assumptions could compromise driving safety and limit the adaptivity of the agent when it comes to realistic driving situations. It is pertinent to note that, in real-world driving situations, drivers display different levels of interactive behaviors with SVs, which is a crucial factor to consider for ensuring safe interactions. Essentially, most existing works do not adequately address the interaction behaviors of SVs, which could restrict the applicability of the trained policy in real-world settings.

To address the above challenges, we present a reward-driven automated curriculum proximal policy optimization (RD-ACPPPO) approach for self-driving tasks at unsignalized intersections. The pipeline of the proposed framework is shown in Fig. 1. Particularly, we decompose the target task into a sequence of generalized curricula with progressively increasing difficulty levels and model the self-driving tasks as a multi-armed bandit (MAB) problem. An exponential-weight-based curriculum selection module is devised to evaluate the importance weights of arms of MAB during the training, which is then integrated with the proximal policy optimization (PPO) framework to facilitate efficient training for the RL policy. The main contributions are summarized as follows:

(1) An interaction-aware RD-ACPPPO framework is proposed for self-driving tasks at unsignalized intersections. Particularly, an adaptive curriculum learning technique is presented to accommodate a progressively increasing difficulty of learning tasks in autonomous driving. In this sense, the proposed approach is able to proactively account for the uncertainty arising from the intention of SVs and the traffic density while adequately addressing the interaction behaviors of SVs.

(2) A distinctive automated curriculum selection module is devised for efficient sampling of episodes in the learning process. By modeling the selection process of curricula as an MAB problem, the importance weights of curricula are dynamically adjusted by real-time assessments of the training progress, which facilitates efficient and effective exploration within the environment for the RL agent.

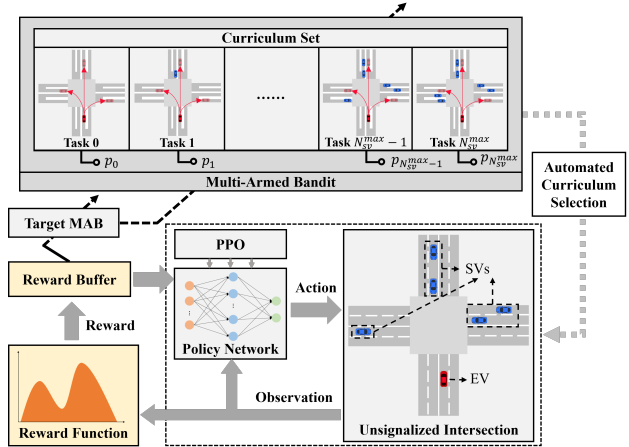


Fig. 1. Overview of the proposed framework for autonomous driving at unsignalized intersections with interactive SVs. In the four-way intersection scenario, the EV is depicted in red, and the SVs are in blue. The solid vehicle and the semi-transparent vehicle represent the start point and the goal point, respectively.

(3) The effectiveness of the proposed framework is demonstrated in simulated unsignalized intersection environments, with comparisons conducted against baseline methods. The results show that the proposed approach exhibits appropriate curriculum transition timing and strong robustness to initialization parameters of MAB, and its trained policy attains superior adaptivity to diverse situational configurations with the highest task success rate among all methods.

II. PROBLEM DEFINITION

A. Problem Statement

The task scenario is set as an unsignalized four-way intersection. The task type, including left turns, going straight, and right turns, is randomly assigned to the EV and SVs. The EV always starts from a random position within the lower lane, and its goal position is also generated randomly within the other three roads, obeying the traffic rules. The SVs are generated on the other three roads randomly, and they drive towards different target lanes.

Here, we assume that the position and velocity information of SVs can be accessed by the EV while the information about their driving intention is unknown. In contrast with [22] and [23], each SV is set to interact with other vehicles in our task scenario, and the number of SVs is not fixed in different tasks. Apparently, the settings in our task scenario introduce noticeable randomness into the task process, which makes the problem challenging but close to real-world conditions. The objective is to generate a sequence of actions to enable the EV to efficiently approach the target point, avoid collisions with other SVs, and stay within road boundaries.

B. Learning Environment

Typically, the self-driving task at unsignalized intersections is modeled as a Markov Decision Process (MDP) [20]. It can be expressed as a tuple $\mathcal{E} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, where $\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma$ represent the state space, action space, state

transition dynamics, reward function, and discount factor, respectively.

State space \mathcal{S} : In our problem, the state space \mathcal{S} consists of kinematic features of all vehicles. At time step t_s , the state matrix is defined as:

$$\mathbf{S}_{t_s} = \left[\mathbf{s}_{t_s}^0 \quad \mathbf{s}_{t_s}^1 \quad \dots \quad \mathbf{s}_{t_s}^{N_{sv}^{max}} \right]^T, \quad (1)$$

where N_{sv}^{max} denotes the maximum number of SVs; $\mathbf{s}_{t_s}^0$ is the state of the EV; $\mathbf{s}_{t_s}^i$ ($i = 1, 2, \dots, N_{sv}^{max}$) represents the state of the i -th SV, which is defined as follows:

$$\mathbf{s}_{t_s}^i = \left[x_{t_s}^i \quad y_{t_s}^i \quad v_{x,t_s}^i \quad v_{y,t_s}^i \quad \sin \psi_{t_s}^i \quad \cos \psi_{t_s}^i \right]^T, \quad (2)$$

where $x_{t_s}^i$ and $y_{t_s}^i$ represent the X-axis and Y-axis coordinates of the vehicle i in the world coordinate system, respectively; v_{x,t_s}^i and v_{y,t_s}^i are the speed of the i -th vehicle along the corresponding axes, respectively; $\psi_{t_s}^i$ is the heading angle of the i -th SV.

Action space \mathcal{A} : In our MDP, a discrete action space consisting of five actions is utilized for the RL agent, which is defined as follows:

$$\mathcal{A} = \{a_1, a_2, a_3, a_4, a_5\}, \quad (3)$$

where a_1, a_2, a_3, a_4 , and a_5 represent the left lane-changing, motion-keeping, right lane-changing, deceleration, and acceleration action, respectively. The RL agent needs to choose a high-level action from the action space \mathcal{A} . Then the selected action is converted into continuous control signals for the vehicle by low-level controllers.

State transition dynamics $\mathcal{P}(\mathbf{S}_{t_s+1} | \mathbf{S}_{t_s}, a_{t_s})$: The transition function defines the transition of state matrix \mathbf{S}_{t_s} , which follows the Markov transition distribution. It is implicitly defined by the environment and unknown to the agent.

Reward function \mathcal{R} : This work assigns a positive reward when the agent successfully completes a self-driving task or maintains survival during the task process. It penalizes collisions, out-of-the-road, and frequent lane-changing behaviors. The details about the reward function will be discussed in Section III-C.

Discount factor γ : The future reward is accumulated with a discount factor $\gamma \in (0, 1)$.

III. METHODOLOGY

A. Task Decomposition and Curriculum Modelling

In this work, we consider the training process of the RL policy as a curriculum learning problem with the target task setting, containing $N_{sv}^{max} + 1$ curriculums with an increasing number of SVs. The curriculum set is defined as

$$\Omega = \{\Omega_i | i = 0, 1, 2, \dots, N_{sv}^{max}\}, \quad (4)$$

where i represents the serial number of the task, which is set to be equivalent to the number of SVs N_{sv} in the particular curriculum. For each curriculum in this set, we can consider it as an arm in the MAB problem [24]. When we select a curriculum to collect an episode, we can obtain related states and rewards from the environment. This process is similar

to the one-step run in the MAB, where one arm is chosen and then a reward is obtained. Based on the above analysis, we consider the problem of the curriculum selection in (4) as a sampling process of the MAB \mathcal{M} with $N_{sv}^{max} + 1$ arms. Under this setting, the MAB agent will sample a sequence of arms over T rounds during the training process:

$$A_T = \{a_{s,1}, a_{s,2}, \dots, a_{s,T}\}, \quad (5)$$

where $a_{s,t}$ ($t = 1, 2, \dots, T$) is the MAB arm sampled in the t -th round. After each round, the selected arm returns a reward, and the MAB agent updates the importance weights according to the historical reward. We aim to obtain an adaptive mechanism that can maximize payoffs obtained by the sampled curriculum sequence A_T from the MAB \mathcal{M} :

$$\mathbf{w}^* = \arg \max_{\mathbf{w}, A_T} \sum_{t=1}^T \hat{r}(t), \quad (6)$$

where \mathbf{w} is the importance weight vector of the MAB; $\hat{r}(t)$ denotes the rescaled reward, which is related to the received reward of the RL agent.

B. Automated Curriculum Selection

For the MAB model constructed in Section III-A, we define its importance weight vector $\mathbf{w}(t)$ and probability distribution vector $\mathbf{p}(t)$ as follows:

$$\begin{aligned} \mathbf{w}(t) &= \{w_i(t) | i = 0, 1, \dots, N_{sv}^{max}\}, \\ \mathbf{p}(t) &= \{p_i(t) | i = 0, 1, \dots, N_{sv}^{max}\}. \end{aligned} \quad (7)$$

In our specific problem, the optimal arm fluctuates throughout different training stages due to shifts in the expected rewards, which is associated with specific tasks. Inspired by the Exp3 algorithm [25], this problem can be addressed by incorporating an external ε -greedy term into the update of the probability. This guarantees that all arms have a probability of being selected throughout the entire training process. Given the weight vector $\mathbf{w}(t)$ at t -th episode, we can calculate the sampling probability of the arm i by the following exponential-weight algorithm:

$$p_i(t) = (1 - \eta) \frac{e^{w_i(t)}}{\sum_{j=0}^{N_{sv}^{max}} e^{w_j(t)}} + \frac{\eta}{N_{sv}^{max} + 1}, \quad (8)$$

$$i = 0, 1, \dots, N_{sv}^{max},$$

where $\eta \in (0, 1)$ is a constant parameter that mediates the trade-off between the exploitation of experience data and random exploration. Then we derive a curricular sample $\Omega(t)$ at t -th episode from the calculated distribution $\mathbf{p}(t)$ as follows:

$$\Omega(t) \sim \mathcal{M}(\mathbf{p}(t)). \quad (9)$$

The MAB agent will receive a reward $r_i(t)$ from the training episode. Then the reward is rescaled to evaluate the performance of the selected arm as follows:

$$\begin{cases} \hat{r}_i(t) = \frac{r_{norm}^i(t)}{p_i(t)}, \\ r_{norm}^i(t) = \frac{2(r_i(t) - k_0 R_{min}(t))}{k_1 R_{max}(t) - k_0 R_{min}(t)} - 1, \end{cases} \quad (10)$$

where $\hat{r}_i(t)$ is the rescaled reward of i -th arm obtained in the t -th episode; $R_{max}(t)$ and $R_{min}(t)$ are the maximum reward

and minimum reward in the history of unscaled rewards up to t -th episode, respectively; k_0 and k_1 are two constant parameters to adjust the rescaling process. Here, we divide all obtained rewards by their current probability, ensuring that arms with potential optimality but currently low probability can be promptly discovered with policy updates. Then we update the weight vector of i -th arm as follows:

$$w_i(t+1) = w_i(t) + \alpha \hat{r}_i(t), \quad (11)$$

where $\alpha \in (0, 1)$ is a constant parameter for adjusting the growth rate of importance weights of each arm.

Considering the nature of the problem addressed in this work, the complexity and difficulty of task scenarios within our task set increase progressively. As the training curriculum is sampled via MAB, and the positions of SVs are randomly generated, low efficient samples are difficult to completely avoid during the training process. To mitigate the overestimation or underestimation in the reward of arms due to occasional sampling, we introduce a target MAB $\hat{\mathcal{M}}$ for stable updating of MAB. After random sampling in the MAB, the obtained rescaled rewards are used to update the parameters of the target MAB. The parameters of the MAB are synchronized with the target MAB after a certain amount of samplings. Therefore, the update of the weight of arm i can be expressed as follows:

$$w_i(t+1) = \begin{cases} w_i(t), & \text{if } t|N_{MAB} \neq 0, \\ w_i(t) + \sum_{j=0}^{N_{MAB}-1} \alpha \hat{r}_i(t-j), & \text{otherwise,} \end{cases} \quad (12)$$

where N_{MAB} is the update interval for synchronizing the parameters of target $\hat{\mathcal{M}}$ to \mathcal{M} .

C. Reward-Driven Automated Curriculum Proximal Policy Optimization

Based on the automated curriculum selection module, a specific arm is sampled from the MAB to generate an episode for the training of the RL policy. Corresponding information including observations, actions, and rewards is stored in a replay buffer for offline training. The RL policy is trained to maximize the following cumulative objective function related to the sampled curriculum sequence A_T as follows:

$$\theta^* = \arg \max_{\theta, A_T} J(\theta_t), \quad (13)$$

where $J(\theta_t)$ is the objective function for policy network with θ_t . In this study, we adopt the clipped objective function of PPO algorithm [26] for training the RL policy:

$$J(\theta_t) = \mathbb{E}_t \left[\min \left(\rho_t(\theta_t) \hat{A}_t, \text{clip}(\rho_t(\theta_t), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) \right], \quad (14)$$

where $\rho_t(\theta)$ represent the similarity between the new policy and old policy; \hat{A}_t is the estimated advantage function; ϵ is the clip parameter.

The reward function plays a pivotal role in guiding RL agents to explore the environment. In this work, the reward function is particularly designed for interactive task scenarios with varying numbers of vehicles. Considering the

characteristics of autonomous driving tasks at unsignalized intersections, the reward function is designed as follows:

$$r(t) = r_S(I_S) + r_C(I_C) + r_{TO}(I_{TO}) + r_{OfR}(I_{OfR}) + r_{LC}(I_{LC}) + r_l, \quad (15)$$

where r_S and r_l are the reward for successfully completing tasks and surviving in the task, respectively; r_C , r_{TO} , r_{OfR} , and r_{LC} are the penalty of collision with SVs, time-out, out of the road boundary, and lane-changing behavior, respectively. I_{event} is the indicator function corresponding to the occurrence of different events as follows:

$$I_{event} = \begin{cases} 1, & \text{if event occurs,} \\ 0, & \text{otherwise.} \end{cases}$$

To achieve high task efficiency, we set the success reward term r_S to be dependent on both the task completion time t_c and the number of SVs N_{sv} when passing through the intersection. This configuration can incentivize RL agents to explore complex task scenarios. Furthermore, to encourage the RL agent to obtain and enhance the interaction-aware and collision-avoidance capability, the collision penalty r_C is set to be positively correlated with the speed of the EV v_{EV} and the number of SVs when a collision occurs. Additionally, the lane-changing penalty is positively correlated with the number of lane changes n_{lc} during the task to avoid frequent lane-changing behavior of the RL agent. r_S and r_C are set as follows:

$$r_S(I_S) = I_S \cdot (\alpha_1 \cdot \frac{t_c}{t_c^{max}} \cdot N_{sv}^2 + \alpha_2), \quad (16)$$

$$r_C(I_C) = I_C \cdot (\alpha_3 \cdot v_{EV} \cdot N_{sv}^2 + \alpha_4),$$

where $\alpha_i, i = 1, 2, 3, 4$ are constant parameters. The remaining reward terms are set to constants if the corresponding events occur.

The complete procedure of the proposed RD-ACPPO framework is summarized in **Algorithm 1**.

In our problem, the difficulty of the autonomous driving task increases with the growing number of SVs in the environment. Here, we model the self-driving tasks at unsignalized intersections with varying numbers of SVs as different arms in an MAB. An exponential algorithm is leveraged to update the importance weights associated with different arms. During the initial training phase, arms corresponding to simpler task scenarios are assigned higher importance weights than more difficult ones due to the unsatisfactory performance of the policy network and the heightened penalties incurred at dense intersections. Consequently, the RL agent undergoes more episodes in simpler task scenarios during the earlier training stages. As the policy network trains over time and achieves proficiency in simple task scenarios, it begins to allocate higher probability to the more challenging arms that are less likely to be selected before, as successful completion of tasks in more complex scenarios yields greater rewards. As the performance of the policy network improves, the MAB dynamically adjusts the weights associated with arms corresponding to different tasks, thus facilitating automated curriculum progression.

Algorithm 1: RD-ACPPO

Input: Environmental state \mathbf{S}_{t_s} , curriculum set Ω ,
MAB update frequency N_{MAB}
Output: $\pi^* = f(\theta^*)$

- 1 Initialize the MAB and target MAB with arm's weights $\{w_i\}$ and $\{\hat{w}_i\}$, where $i = 0, 1, \dots, N_{sv}^{max}$, the policy network with parameter θ_0 ;
- 2 **while** $t \leq t_{max}$ **do**
- 3 Calculate the probability distribution $\mathbf{p}(t)$ according to (8);
- 4 Derive a sample $\Omega(t)$ according to (9);
- 5 Reset the environment according to the setting of sampled curriculum $\Omega(t)$;
- 6 Record the episode of the RL agent and obtain the reward $r_i(t)$;
- 7 Update the policy network by (13);
- 8 Calculate the rescaled reward $\hat{r}_i(t)$ by (10);
- 9 Update the corresponding weight of the target MAB $\hat{w}_i(t+1)$ by (11);
- 10 **if** $t \mid N_{MAB} = 0$ **then**
- 11 | $w_i(t) = \hat{w}_i(t)$, $i = 0, 1, \dots, N_{sv}^{max}$;
- 12 **end**
- 13 **end**
- 14 Output the final policy $f(\theta^*)$.

IV. EXPERIMENTS

A. Comparative Experimental Settings

The comparative experiments are conducted on the Windows 11 system with a 3.90 GHz AMD Ryzen 5 5600G CPU and an NVIDIA GeForce RTX 3060 GPU. The unsignalized intersection is constructed based on *Highway_Env* environment [27], where each road consists of two carriageways. Here, we assume that there are up to 6 SVs simultaneously at intersections. We adopt the actor-critic architecture to implement the RD-ACPPO framework. The action network and critic network are set as fully connected networks with 1 hidden layer of 128 units and 64 units by PyTorch and trained with the Adam optimizer. The number of epochs is set to 20. The learning rate of the action network and critic network are set to 5×10^{-4} and 1×10^{-3} , respectively. γ is set to 0.9.

Considering the escalating difficulty of autonomous driving tasks at unsignalized intersections with the increase in the number of SVs, we empirically set the initial weights of the arms in the MAB to an exponential form that is inversely proportional to the count of SVs as follows:

$$w_i(0) = e^{-2i}, i = 0, 1, \dots, 6 \quad (17)$$

Additionally, to investigate the impact of the initial weights of the MAB arms on the training process and the performance of the trained policy, we have also conducted training for an RL policy with identical initial weights of arms in the MAB as follows:

$$w_i(0) = 1, i = 0, 1, \dots, 6 \quad (18)$$

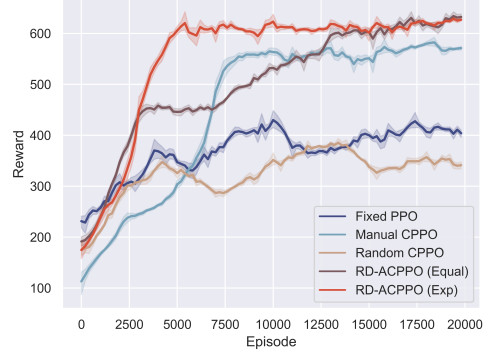


Fig. 2. Reward curve comparison among different methods. The training curves are smoothed by the Savitzky-Golay filter.

η is set to 0.2. N_{MAB} is set to 1000. Here, we compare the proposed RD-ACPPO method with the following baseline methods.

- Fixed PPO: the agent is trained by standard PPO method [26] at intersections with $N_{sv} = 6$.
- Manual CPPO: an simplified version of [20] with a fixed clip parameter.
- Random CPPO: the curriculum is sampled randomly, with an equal probability of selection for all arms.

The clip parameters of all methods were set to $\epsilon = 0.2$. For the sake of fairness, other network parameters of all methods are set to be the same. The behaviors of SVs are characterized by the intelligent driver model (IDM) [28]. Considering the physical limitations, the maximum acceleration and steering angle of vehicles are set to 8 m/s^2 and 45 degrees, respectively. Then we test all trained policies at unsignalized intersections with $N_{sv} = 0, 1, \dots, 6$ and record the success rate, collision rate, and time-out rate to evaluate the driving performance of these trained policies.

B. Training Results

Reward curve comparisons among different approaches and the probability changes of each arm in MAB during the training process are shown in Figs. 2-4. According to Fig. 2, we can find that the reward curve of RD-ACPPO (Exp) initially began with lower values compared to Fixed PPO, but it surpassed that of Manual CPPO. This discrepancy can be attributed to the sensitivity of the reward function to the complexity of the training scenario, specifically characterized by the number of SVs, while Fixed PPO was only trained in the most complex scenarios. Subsequently, according to Fig. 3, due to the evaluation of potential rewards for arms of the MAB in the automated curriculum module, arms $N_{sv} = 1, 2, 5, 6$ were identified as the most preferable choices starting from the 1000-th, 2000-th, 3000-th, and 5000-th episodes in RD-ACPPO (Exp), respectively. As a result, their probability surpassed those of other arms. Correspondingly, the reward function of the RD-ACPPO agent experienced a rapid increase from these four time points.

It is highlighted that due to the random initialization of positions for EV and SVs, along with the sampling of arms in the MAB of the RD-ACPPO method, slight fluctuations

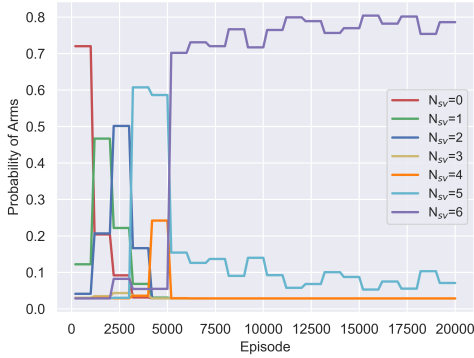


Fig. 3. Probability of arms with exponential initialization weight during the training process.

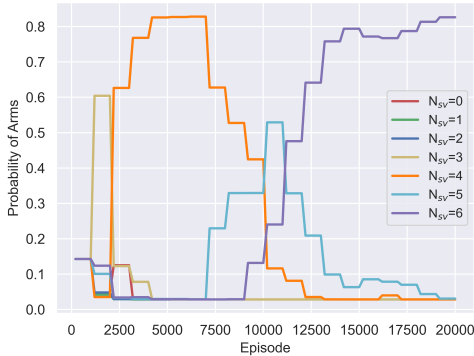


Fig. 4. Probability of arms with equal initialization weight during the training process.

in the reward curve still occur in the later stages of training. These fluctuations are attributed to the inherent randomness of the testing initialization, even though the RL policy has already converged.

According to Fig. 3 and Fig. 4, both the MAB with equal initial weights and the MAB with exponential initial weights demonstrated a trend of curriculum selection that progressed from easier to more challenging tasks during the training process. As the RL policies were updated, there was a gradual shift towards selecting arms that offered greater potential rewards. For the MAB with exponential initial weights, in the first 3000 episodes, arms corresponding to $N_{sv} = 0, 1, 2$ were primarily chosen. This phase allowed the learning of the capability to reach destinations and initial collision avoidance skills in environments with a few SVs at unsignalized intersections. Subsequently, in later episodes, the focus shifted to enhancing collision avoidance skills in scenarios with a high density of SVs. After the 5000-th episode, the MAB predominantly selected the arm with $N_{sv} = 6$, indicating continuous improvement in the collision avoidance capabilities with the RL policy. This enhancement enabled the RL agent to obtain pertinent rewards in the most complex unsignalized intersection environments associated with this arm, and it also facilitated the convergence of the RL policy.

According to Fig. 4, when initializing with identical weights, the weight adjustments of arms during the training

TABLE I
PERFORMANCE COMPARISON AT EMPTY UNSIGNALIZED INTERSECTIONS AMONG DIFFERENT METHODS.

Methods	$N_{sv}=0$		
	succ.(%)	coll.(%)	time-out(%)
Fixed PPO	77.5	0	22.5
Manual CPPO	100	0	0
Random CPPO	100	0	0
RD-ACPPPO (Equal)	100	0	0
RD-ACPPPO (Exp)	100	0	0

process exhibited certain divergences compared to that of arms initialized with exponential weights. It is worth noting that the reward curve of RD-ACPPPO (Equal) slightly exceeded that of RD-ACPPPO (Exp) between the 1000-th and 3000-th episodes. This situation occurred because RD-ACPPPO (Equal) shifted its preferable arms to $N_{sv} = 3, 4$, while RD-ACPPPO (Exp) was still engaged in the training phase focusing on $N_{sv} = 1, 2$ at this stage. Yet, as the RL policy approached convergence, the reward curves of policies under both sets of initial conditions tended to align at a similar level. However, the RL policy using MABs with equal initial weights exhibited a slower convergence compared to the policy utilizing exponential initial weights. This result indicates that while variations in initial weight settings of MAB could influence the speed of convergence, they do not significantly affect the ultimate training outcomes of the RL policy. This resilience highlights the robustness of the proposed method, affirming its effectiveness amidst variations in initial conditions, particularly within the challenging domain of autonomous driving at unsignalized intersections.

After completing the whole training process, the reward curve of RD-ACPPPO generally outperformed all baseline methods. This overall superiority suggests that the proposed approach has yielded the highest sample efficiency and the most effective training outcomes for the RL policy.

C. Performance Evaluation

To further evaluate the performance of the proposed RD-ACPPPO algorithm, we conducted tests on agents trained by all methods in target scenarios with the number of SVs ranging from 0 to 6. Each agent underwent 200 repeated tests in each testing scenario, including go-straight, left-turn, and right-turn tasks. The test results are presented in Table I and Table II, respectively. Table I shows that agents trained by Fixed PPO exhibit a 22.5% timeout rate in the scenario without SVs. This indicates that training the RL agent only in the scenario with $N_{sv} = 6$ can lead to suboptimal policy. In contrast, other agents trained through curriculum learning do not exhibit this issue in this particular scenario, indicating that curriculum learning can mitigate the problem of poor generalization to empty unsignalized intersections.

From the testing results, we can observe that the proposed RD-ACPPPO method achieved the highest task success rate across all scenarios. The test results of RD-ACPPPO (Equal) and RD-ACPPPO (Exp) across various scenario settings are similar, which is consistent with the training outcomes indicated in the reward curves. While the success rate of RD-ACPPPO decreases as the complexity of scenarios increases,

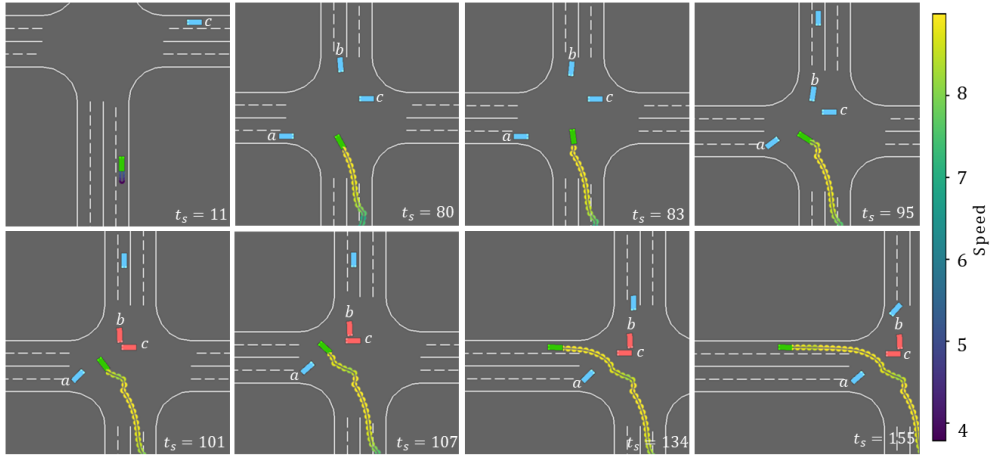


Fig. 5. Demonstration of the driving performance attained by the proposed RD-ACPPO method in an unprotected left-turn task. The green car and blue cars represent the EV and SVs under normal driving conditions, respectively. The red cars represent the vehicles that have collided.

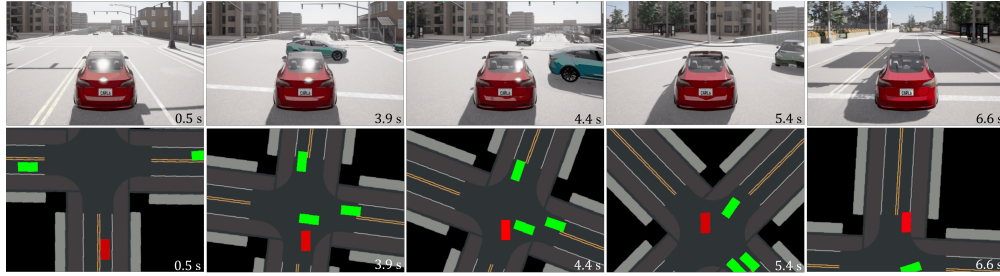


Fig. 6. Key frames of the experimental validation of our method in CARLA. The top of this figure shows the third-person views of the EV, while the bottom presents the bird-eye views. The red rectangle and the green rectangles represent the EV and SVs, respectively.

TABLE II

PERFORMANCE COMPARISON AT UNSIGNALIZED INTERSECTIONS WITH DIFFERENT NUMBERS OF SVs AMONG DIFFERENT METHODS.

Methods	$N_{sv}=1$		$N_{sv}=2$		$N_{sv}=3$		$N_{sv}=4$		$N_{sv}=5$		$N_{sv}=6$	
	succ.(%)	coll.(%)	succ.(%)	coll.(%)	succ.(%)	coll.(%)	succ.(%)	coll.(%)	succ.(%)	coll.(%)	succ.(%)	coll.(%)
Fixed PPO	92.5	4.5	83.5	16.5	75.5	24.5	73.5	26.5	67.5	32.5	64.5	35.5
Manual CPPO	99.5	0.5	87.5	12.5	80.5	19.5	76.5	23.5	73	27	70.5	29.5
Random CPPO	98	2	85.5	14.5	79	21	75	25	70	30	67.5	32.5
RD-ACPPO (Equal)	100	0	92	8	85.5	14.5	82.5	17.5	79	21	75	25
RD-ACPPO (Exp)	100	0	93.5	6.5	86.5	13.5	81.5	18.5	79	21	75.5	24.5

it still maintains the highest success rate. This indicates that the proposed approach exhibits the best driving performance and adaptivity to environmental changes, which can be attributed to the real-time assessment provided by the automated curriculum selection mechanism during the training process. Among all testing results attained by the RD-ACPPO method, we pick up the results from a left-turn task for demonstration, and the details are presented in Fig. 5. The EV initiated the left-turn driving task in the lower area at the intersection and drove towards the center area. Then the EV decelerated and steered right when it encountered an SV (labeled as *a*) approaching from its left side in the central area. Following the right turn of the EV, two SVs (labeled as *b* and *c*) approached the EV from the upper and from the right upper side, respectively. Simultaneously, the SV *a* exhibited intentions of decelerating to yield. Consequently, the EV executed a left turn to avoid collisions with SV *b* and SV *c*, subsequently realigning and continuing towards the intended lane at the top of the left area. The above actions taken by the RL agent indicated that the policy had learned

to interact safely with SVs. Finally, the EV safely arrived at its destination and successfully completed the task. This example demonstrates the interaction-aware driving behavior of the RL policy trained by RD-ACPPO. Overall, we observe that the performance of the proposed method outperformed all baselines. This highlights the capability of the automated curriculum selection module to enhance the driving strategy performance of agents in different environments.

D. Experimental Validation in CARLA

We further validate our method in the high-fidelity simulator CARLA [29]. The validation is conducted in the testbed of an unsignalized intersection scenario in Town03, where the maximum number of SVs is set to three. All vehicles on the road are set to Tesla Model 3, and SVs drive in built-in autopilot mode. Furthermore, full observability of the EV is assumed. We retrained the policy network and evaluated the driving performance in the aforementioned testbed. We test the trained policy on the unsignalized intersection and then pick up a result from a left-turn task for demonstration

in CARLA. The key frames of a representative trail are visualized in Fig. 6. The results illustrate that the cooperation ability of our interaction-aware driving strategy is retained. At 3.9 s, the EV decelerated and steered towards the left side, yielding to the SV approaching from the left side. Then the EV accelerated and drove towards the target point when it observed the yield behavior of the SV coming from the upper side from 4.4 s to 5.4 s. Finally, the EV successfully completed a collision-free left-turn task. The validation results highlight the effectiveness of our approach in the high-fidelity simulator.

V. CONCLUSION

We presented the RD-ACPPPO approach to train the RL policy for interaction-aware self-driving at unsignalized intersections. The proposed automated curriculum selection mechanism allows for the reasonable allocation of the importance weights for different curricula during various RL training periods. The RL policy trained by the proposed framework demonstrated reasonable curriculum transition timings during the training phase. The results indicated that the RD-ACPPPO agent attained the highest task success rate, strong robustness to initialization parameters of MAB, and best adaptability across various test tasks. Furthermore, the effectiveness of the proposed method was validated in CARLA. While this work primarily focuses on the application in self-driving tasks at intersections, it can also be suitably exploited and extended to address other autonomous driving scenarios.

REFERENCES

- [1] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.
- [2] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2020.
- [3] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *Journal of Field Robotics*, vol. 37, no. 3, pp. 362–386, 2020.
- [4] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 33–47, 2020.
- [5] F. Li, X. Li, J. Luo, S. Fan, and H. Zhang, "Open-set intersection intention prediction for autonomous driving," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13092–13098, IEEE, 2021.
- [6] L. Zheng, R. Yang, Z. Peng, M. Y. Wang, and J. Ma, "Spatiotemporal receding horizon control with proactive interaction towards autonomous driving in dense traffic," *arXiv preprint arXiv:2308.05929*, 2023.
- [7] L. Wei, Z. Li, J. Gong, C. Gong, and J. Li, "Autonomous driving strategies at intersections: Scenarios, state-of-the-art, and future outlooks," in *Proceedings of the IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 44–51, IEEE, 2021.
- [8] A. Aksjonov and V. Kyrki, "Rule-based decision-making system for autonomous vehicles at intersections with mixed traffic environment," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 660–666, IEEE, 2021.
- [9] R. Tian, N. Li, I. Kolmanovsky, Y. Yildiz, and A. R. Girard, "Game-theoretic modeling of traffic in unsignalized intersection network for autonomous vehicle control verification and validation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2211–2226, 2020.
- [10] L. Riegger, M. Carlander, N. Lidander, N. Murgovski, and J. Sjöberg, "Centralized MPC for autonomous intersection crossing," in *Proceedings of the IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 1372–1377, IEEE, 2016.
- [11] M. Kneissl, A. Molin, H. Esen, and S. Hirche, "A feasible MPC-based negotiation algorithm for automated intersection crossing," in *2018 European Control Conference (ECC)*, pp. 1282–1288, IEEE, 2018.
- [12] D. Isele, R. Rahimi, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating occluded intersections with autonomous vehicles using deep reinforcement learning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2034–2039, IEEE, 2018.
- [13] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2021.
- [14] C. Xia, M. Xing, and S. He, "Interactive planning for autonomous driving in intersection scenarios without traffic signs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24818–24828, 2022.
- [15] Z. Qiao, J. Schneider, and J. M. Dolan, "Behavior planning at urban intersections through hierarchical reinforcement learning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2667–2673, IEEE, 2021.
- [16] Y. Huang, S. Yang, L. Wang, K. Yuan, H. Zheng, and H. Chen, "An efficient self-evolution method of autonomous driving for any given algorithm," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [17] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th International Conference on Machine Learning (ICML)*, pp. 41–48, 2009.
- [18] Y. Song, H. Lin, E. Kaufmann, P. Dürr, and D. Scaramuzza, "Autonomous overtaking in gran turismo sport using curriculum reinforcement learning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9403–9409, IEEE, 2021.
- [19] Y. Wang, Y. Li, Z. Peng, H. Ghazzai, and J. Ma, "Chance-aware lane change with high-level model predictive control through curriculum reinforcement learning," *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.
- [20] Z. Peng, X. Zhou, Y. Wang, L. Zheng, M. Liu, and J. Ma, "Curriculum proximal policy optimization with stage-decaying clipping for self-driving at unsignalized intersections," in *Proceedings of the IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 5027–5033, IEEE, 2023.
- [21] A. Graves, M. G. Bellemare, J. Menick, R. Munos, and K. Kavukcuoglu, "Automated curriculum learning for neural networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 1311–1320, 2017.
- [22] Z. Qiao, K. Muelling, J. M. Dolan, P. Palanisamy, and P. Mudalige, "Automatically generated curriculum based reinforcement learning for autonomous vehicles in urban environment," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pp. 1233–1238, IEEE, 2018.
- [23] S. Khaitan and J. M. Dolan, "State dropout-based curriculum reinforcement learning for self-driving at unsignalized intersections," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 12219–12224, IEEE, 2022.
- [24] A. Slivkins *et al.*, "Introduction to multi-armed bandits," *Foundations and Trends® in Machine Learning*, vol. 12, no. 1-2, pp. 1–286, 2019.
- [25] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The non-stochastic multiarmed bandit problem," *SIAM Journal on Computing*, vol. 32, no. 1, pp. 48–77, 2002.
- [26] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.
- [27] E. Leurent, "An environment for autonomous driving decision-making." <https://github.com/eleurent/highway-env>, 2018.
- [28] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, p. 1805, 2000.
- [29] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Proceedings of the Conference on Robot Learning*, pp. 1–16, PMLR, 2017.