Robust Multi-Agent Reinforcement Learning for Autonomous Vehicle in Noisy Highway Environments

Lilan Lin LINLILANLLL@163.COM
Xiaotong Nie* NIEXIAOTONG@ZSTU.EDU.CN
Jian Hou CHANGELEAP@163.COM
School of Computer Science and Technology, Zhejiang Sci-Tech University.Zhejiang, China

Editors: Vu Nguyen and Hsuan-Tien Lin

Abstract

The field of research on multi-agent reinforcement learning (MARL) algorithms in self-driving vehicles is rapidly expanding in mixed-traffic scenarios where autonomous vehicles (AVs) and human-driven vehicles (HDVs) coexist. Most studies assume that all AVs can obtain accurate state information. However, in real-world scenarios, noisy sensor measurements have a significant impact. To address this issue, we propose an effective and robust MARL algorithm Multi-Agent Proximal Policy Optimization with Curriculum-based Adversarial Learning (CA-MAPPO) for situations where the observation perturbations are considered. The proposed approach incorporates adversarial samples during training and adopts a curriculum learning approach by gradually increasing the noise intensity. By evaluating the proposed approach in the ideal environment and scenarios under noise attacks with varying intensities, experiment results demonstrate that the proposed algorithm enables AVs to achieve a success rate of over 70% for the multi-lane highway on-ramp merging task, achieving a maximum average speed of up to over 19 m/s and performing significantly better than the state-of-the-art MARL algorithms such as MAPPO and MAACKTR.

Keywords: Autonomous driving, multi-agent deep reinforcement learning, adversarial learning, curriculum learning

1. Introduction

In recent years, the increasing introduction of autonomous vehicle (AV) products by companies such as Waymo, Tesla, and Baidu Apollo has highlighted the importance of autonomous driving as the future of transportation. Most existing research on autonomous driving assumes that all AVs operate normally. However, in real-world environments, factors such as hardware or software failures, adverse weather conditions, etc., may lead to abnormal behavior of AVs, significantly affecting the performance on the road and even causing traffic accidents. Therefore, it is vital to develop strategies that can handle disruptions and failures in complex driving scenarios.

To tackle these challenges, multi-agent reinforcement learning (MARL) has emerged as a promising approach. In recent years, there has been a substantial amount of research on MARL, which has achieved notable success in a variety of complex tasks. StarCraft II (Vinyals et al., 2019) enables agents to compete at the level of top human players in this real-time strategy game and the Google Football environment (Kurach et al., 2020) has also been instrumental in training multiple agents for both cooperation and competition,

* Corresponding author

highlighting complex strategic behavior. These accomplishments illustrate the potential of MARL in coordinating multiple agents and solving complex problems.

In the context of MARL, each AV can be considered an agent. Our research focuses on developing a robust MARL algorithm to address situations where the AV's observations are perturbed in highway on-ramp merging scenarios. The objective is to develop strategies that enhance the algorithm's resilience and improve the system's stability in the presence of abnormal conditions. Firstly, we introduce noise attacks targeting the sensors of AVs to simulate perturbed observations. Then we incorporate adversarial samples into the training process. However, the existence of excessive noise attacks during adversarial training can lead to unstable and suboptimal policies. To address this issue, we adopt a curriculum learning approach, where the noise intensity is gradually increased throughout the training process. The primary contributions of our research can be summarized as follows:

- By incorporating adversarial samples into the training process, AVs are enabled to learn adversarial strategies, thereby enhancing their ability to counteract noise attacks and improving the robustness of the system significantly.
- We propose a novel robust MARL algorithm Multi-Agent Proximal Policy Optimization with Curriculum-based Adversarial Learning (CA-MAPPO) by gradually increasing the noise intensity in the training process. This allows AVs to progressively adapt to more challenging scenarios, leading to a more stable and effective learning process.
- The proposed algorithm significantly improves the success rate of the multi-lane highway on-ramp merging task and the average speed of AVs in the ideal environment and noisy environments with different noise intensities, achieving higher overall returns compared to the state-of-the-art (SOTA) MARL benchmarks MAPPO and MAACKTR.

The remainder of the paper is organized as follows. Section 2 briefly introduces the related work of our research. The problem formulation is described in Section 3 and the proposed method is detailed in Section 4. Experiments, results, and evaluations are presented in Section 5. We conclude the paper and discuss future works in Section 6. Hyperparameter setting and more explorations about the proposed method are supplemented in Appendix.

2. Related Work

2.1. On-ramp Merging Strategies for AVs in Mixed Traffic

The task of on-ramp merging for AVs is a complex endeavor due to the dynamic nature of traffic and the necessity for precise timing and coordination with human-driven vehicles(HDVs). A multitude of strategies have been proposed to address this challenge:

Rule-based and optimization-based approaches offer contrasting methodologies for guiding AVs (Ding et al., 2019; Hu and Sun, 2019). Rule-based methods rely on predefined rules and heuristics, effective in simpler scenarios but inadequate in complex situations due to their lack of adaptability. Optimization-based methods treat vehicle interactions as dynamic systems, using actions from controlled vehicles as inputs. Despite their capacity to handle complexity, optimization-based methods require precise dynamic models and significant computational resources. Furthermore, it is important to note that the Model Predictive Control (MPC) approach (Dixit et al., 2019), a notable optimization-based method, may not be suitable for mixed-traffic scenarios.

Recent research findings suggest that MARL presents a promising strategy for AVs in navigating the intricacies of highway on-ramp merging scenarios (Zhang et al., 2023; Chen et al., 2023). Chen et al. (2023) have proposed a scalable MARL framework for AVs to adapt to HDVs in mixed-traffic on-ramp merging, enhancing traffic throughput and safety with action masking and a priority-based safety supervisor. (Zhang et al., 2023) have presented the Independent Proximal Policy Optimization (IPPO) algorithm, which significantly improves the success rate and efficiency of AVs in the scenario. However, both studies are based on relatively ideal environments, and in real-world setting, unexpected situations may occur, leading to incorrect behaviors by AVs. Consequently, designing robust MARL algorithms becomes one of the most significant challenges in this context.

2.2. Robust MARL

In addition to integrating safety constraints into the MARL formulation, adversarial learning (Goodfellow et al., 2015) is also a common method for improving the robustness of the MAS.

Early research in adversarial reinforcement learning has introduced the concept of Robust Adversarial Reinforcement Learning (RARL) (Pinto et al., 2017). RARL treats environmental biases and errors in the modeling process as disturbances and addresses them through adversarial training. Subsequent studies have introduced the Antagonist-Ratio Training Scheme (ARTS) (Phan et al., 2020) as an extension of RARL into multi-agent systems. ARTS adopts a mixed cooperative-competitive MAS setting, where two groups of agents collaborate internally while competing with each other. Thus, its adversarial learning relies on continuously adjusting and changing the team members of the agents. Subsequently, further enhancement was proposed with the Randomized Adversarial Training (RAT) method (Phan et al., 2021). RAT incorporates a uniformly distributed adversary ratio, using a phased approach and the Value-Decomposition Networks (VDN) (Sunehag et al., 2017) method to update the learning functions for protagonists and antagonists separately, enabling the MAS to better adapt to various unforeseen circumstances and thereby improving system robustness.

The research mentioned earlier concentrates on employing specific adversarial training patterns to improve the robustness of systems. In subsequent studies, there is a notable transition in focus toward adversarial training that emphasizes the diversification of adversarial samples. This shift aims to provide further improvements in the overall robustness of the systems under consideration.

The first study on robustness in MARL introduces an attack method from the perspective of influencing the observations of agents (Lin et al., 2020). In the past two years, research has emerged that approaches adversarial attacks from the Markov Decision Process (MDP) perspective, introducing perturbations to the states, actions, and rewards of agents in three dimensions (Guo et al., 2022). The study (Zhao et al., 2022) addressing the phenomenon of "crashed agents" in the MAS presents a coach-assisted training framework and a method for adaptively adjusting crashed rates to enhance system robustness.

Recently,in the field of autonomous driving, He et al. (2022) have proposed a novel constrained adversarial reinforcement learning approach for ensuring robust decision-making of AVs at highway on-ramps, effectively addressing environmental uncertainties and distur-

bances for safer merging. However, there has been scarce research in the domain of the MAS within this field.

3. Problem Formulation

3.1. Problem Description

This study investigates the intricate challenge of developing robust cooperative strategies for AVs operating in a mixed traffic environment, where the vulnerability of AVs' states to external attacks is considered. Modeling the on-ramp merging environment in a mixed traffic is achieved through a model-free multiple-agent network $\mathcal{G} = (\nu, \varepsilon)$. Each agent i (AV i) $\in \nu$ communicates with neighbor AV $j \in \mathcal{N}_i$ where $\mathcal{N}_i = \{j | \varepsilon_{ij} \in \mathcal{E})\}$ through edge connections (ε_{ij}), establishing a decentralized MARL framework (Chen et al., 2023). In this framework, each AV i selectively observes a subset of the environment, mirroring the realistic constraints of AVs in perceiving or communicating only with nearby vehicles. The holistic modeling approach characterizes the entire dynamic system as a Partially Observable Markov Decision Process (POMDP) $\{A_i, S_i, R_i\}_{i \subseteq \nu}, T$, encapsulating the essential components of AV actions, states, and rewards within the multi-agent network context, emphasizing the stochastic nature of system evolution through the transition function T.

3.2. State-perturbed MARL Problem Formulation for AVs

State Space (S): Assuming an observation range $\leq 150\ m$ for AVs. The state space of AV i is defined as a matrix of size $N_{\mathcal{N}i} \times C$, where $N_{\mathcal{N}_i}$ denotes the number of observed vehicles ($N_{\mathcal{N}_i} = 5$ in Fig. 1) and C is the number of features per vehicle. The features are given by $(ispresent, x, y, v_x, v_y)$, where $ispresent$ indicates whether the vehicle is observed by the ego vehicle, x and y represent longitudinal and lateral positions respectively, and v_x and v_y represent longitudinal and lateral velocities. $N_{\mathcal{N}i}$ is predefined to ensure the dimensionality of the state space remains constant.

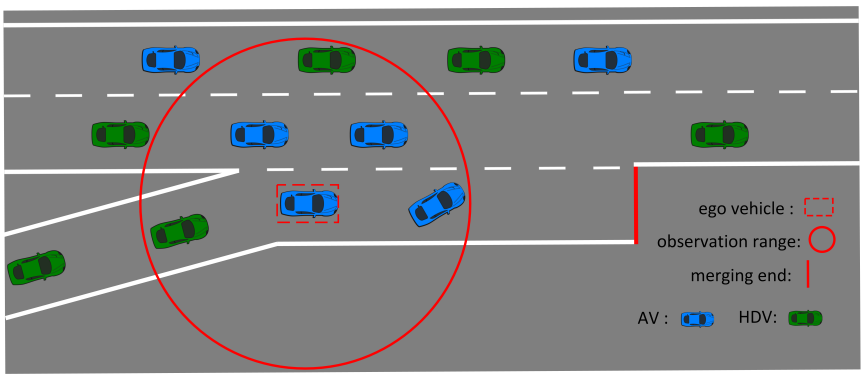


Figure 1: A multi-lane highway on-ramp merging scenario

Action Space (A): Advanced decisions include turning left, turning right, cruising, speeding up, and slowing down. With selected high-level decisions, the low-level controllers generate corresponding steering and throttle control signals to maneuver the AV.

Reward Function(R): The formulation of the reward function holds paramount significance in the training of AVs. The objective is to enable the trained agent to navigate the merging section both safely and efficiently. Consequently, the reward for the i-th AV at time step t is defined as follows:

$$r_{i,t} = w_c r_c + w_s r_s + w_h r_h + w_m r_m + w_l r_l \tag{1}$$

The positive weighting scalars w_c, w_s, w_h, w_m, w_l correspond to collision evaluation r_c, stable-speed evaluation r_s, headway time evaluation r_h, merging cost evaluation r_m, and lane-change cost evaluation r_l respectively. Given the paramount importance of safety, w_c is assigned a significantly greater magnitude compared to the other weights, emphasizing the prioritization of safety. The details of various reward metrics are as follows:

To ensure safety during the driving process, a collision penalty r_c is chosen as a punishment, and it is given sufficient weight

$$r_c = \begin{cases} -1, & \text{if collision happens} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The speed reward r_s is a positive incentive within the environment that encourages AVs to achieve higher speed values within a restricted speed range.

$$r_s = \frac{v_t - v_{\min}}{v_{\max} - v_{\min}} \tag{3}$$

where v_t is the vehicle speed, v_{max} is the maximum speed, and v_{min} is the minimum speed.

For safety during the driving process of AVs, it is imperative to maintain a certain distance between the vehicle and the preceding one by r_h (Chen et al., 2023)

$$r_h = \min\left(\log \frac{d_{\text{headway}}}{t_h v_t}, 0\right) \tag{4}$$

where d_{headway} denotes the distance between a vehicle and the one preceding it, t_h is set as a predefined threshold. A penalty is incurred if the time gap between the following vehicle and the preceding one falls below this headway threshold. The recommended setting for t_h is $1.2s$ (Ayres et al., 2001).

Waiting too long on a ramp can easily lead to deadlock, greatly impacting efficiency. Therefore, r_m is set to penalize the waiting time of AVs merging into lanes (Chen et al., 2023)

$$r_m = -\exp\left(-\frac{(x-L)^2}{10L}\right) \tag{5}$$

in the equation, x represents the distance traveled by the vehicle on the ramp, and L denotes the length of the merging area of the ramp. The closer the vehicle's position to the merging endpoint of the ramp, the higher the penalty value.

Finally, to deter excessive lane changes on straight roads, the penalty r_l is applied to lane-changing maneuvers

$$r_l = \begin{cases} -1, & \text{if change lane} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

At the reward structure level, global rewards are replaced with region-based rewards within the observation range of AVs, which proved to be more effective. During the process of training, the reward for an individual AV can be formulated as follows:

$$r_{i,t} = \frac{1}{|\nu_i|} \sum_{j \in \nu_i} r_{j,t} \tag{7}$$

where ν_i is a set including the i-th AV and its neighbor AV $j \in \mathcal{N}_i$, and $|\nu_i|$ indicates the cardinality of the set.

The Multi-Agent Markov Decision Process (MMDP) outlined above operates under the assumption that all AVs are free from attacks. Given adversary p that perturbs the state s, B is the available perturbation set, the purpose of our attack can be formulated as:

$$\begin{aligned} &\min_{p(s)} G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}, \\ &\text{s.t. } p(s) \in B(s), s' \sim P(s'|s, a'), a' \sim \pi(.|p(s)), \end{aligned} \tag{8}$$

where $\pi(.|p(s))$ means the policy π is learned from perturbed state $p(s)$.

In the process of testing robustness, we aim for the MAS to maintain acceptable performance levels even when under attack towards state. So we hope to obtain the optimal strategy π^*. G_t below is the total return mentioned in (8). This process can be formulated as:

$$\pi^* = \arg\max_{\pi} G_t \tag{9}$$

4. Methodology

In this section, we propose a decision-making framework for AVs. A method has also been proposed to simulate the noise attack on AV's observations. Additionally, we describe exactly how adversarial training is enhanced through the curriculum learning method.

4.1. The Decision-making Framework

The proposed algorithm is an improvement built upon the baseline MAPPO algorithm. We adopt a decentralized training mode due to its superior effectiveness compared to centralized training. Fig. 2 illustrates the decision framework of the entire autonomous driving system.

4.1.1. The Basic Decision-making Algorithm MAPPO

In this article, MAPPO is derived by modifying the PPO algorithm using an independent learning approach (Zhang et al., 2023). The PPO algorithm structure primarily comprises two actor networks and two critic networks. Throughout the training phase, AVs sample actions from the probability distribution output by the actor network, execute these actions to interact with the environment and collect interaction data.

Initially, the ratio between the new and old policies is computed to regulate the magnitude of policy updates during training. Subsequently, the actor network, representing the new policy, undergoes updating via gradient ascent. Periodically, the parameters are copied to the actor network, defining the old policy based on a specified time step length.

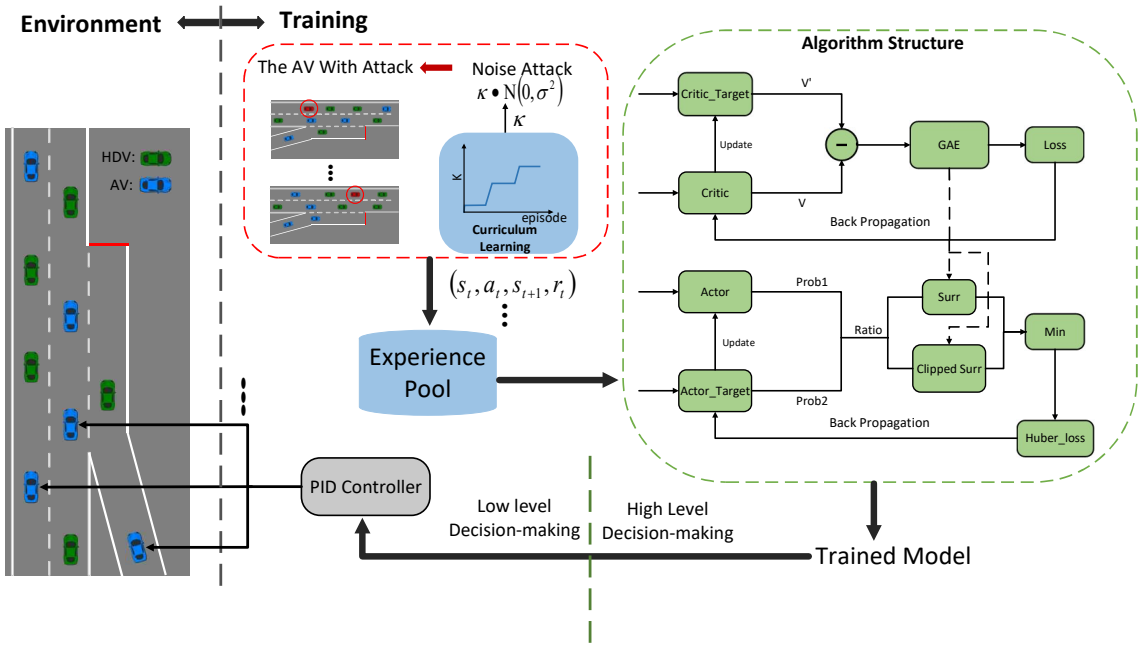


Figure 2: The decision-making framework of the proposed algorithm

To prevent significant distribution differences between the new and old policies, the ratio is clipped. The loss function of the actor network is as follows:

$$L^{\mathrm{CLIP}}(\theta) = \mathbb{E}_t \left\{ \min \left[r_t(\theta) A_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon) A_t \right] \right\} \tag{10}$$

where $r_t(\theta)$ is the ratio of the old to the new strategy, and ϵ is the hyperparameter.

The loss function of the critic network is defined as follows:

$$\mathrm{Loss}(\theta) = \begin{cases} \frac{1}{2}(V_\theta(s_t) - V_t')^2 & \text{if } |V_\theta(s_t) - V_t'| < 1 \\ |V_\theta(s_t) - V_t'| - \frac{1}{2} & \text{otherwise} \end{cases} \tag{11}$$

where the current environmental state s_t is input into the critic network of the algorithm to generate value $V_\theta(s_t)$. And the target value V_t' is computed using generalized advantage estimation (GAE) (Schulman et al., 2017)

$$V_t' = V_\theta(s_t) + A_t = V_\theta(s_t) + \sum_{l=t}^{T} (\gamma\lambda)^{l-t-1} \delta_{l-1} \tag{12}$$

where δ_{t-1} is the temporal difference error, and $\delta_{t-1} = r_{t-1} + \gamma V(s_t) - V(s_{t-1})$.

Both the actor and critic networks randomly sample data from the experience pool, calculate the loss function, and update network parameters through backpropagation.

4.1.2. The Behavioral Decision Model

In Fig. 2, the high-level decisions of AVs are made by the proposed algorithm and will be tracked by the low-level controller (PID controller). Firstly, The AV collects information

about its state and the traffic around it. The behavioral network then uses this information to determine one of five driving behaviors: changing lanes to the left, lane keeping, changing lanes to the right, accelerating, and decelerating. Finally, the kinematic model translates the driving behavior into acceleration and steering angle commands to control the AV on the road by the PID controller.

4.2. The Perturbed Observations on the AV

Defining the noise attack targets the latter four dimensions, x, y, v_x, v_y of the observation space (ispresent, x, y, v_x, v_y) of the AV, the attack follows a normal distribution, thus the process of being attacked can be formalized as:

$$s_{\text{noise}} = s + K \cdot N(0, \sigma^2) \tag{13}$$

The above can be regarded as a multi-dimensional Gaussian noise attack N with mean 0 and variance σ^2 on each dimension of the observation space, where K represents the noise intensity. Inspired by Yang et al. (2023), the initial assumption is that the σ on different dimensions are $\sigma_x = 10m$, $\sigma_y = 1m$, $\sigma_{v_x} = 2m/s$, and $\sigma_{v_y} = 0.2m/s$.

4.3. Curriculum-based Adversarial Learning

A curriculum learning approach is adopted to enhance the stability of adversarial learning by gradually increasing noise intensity. After determining the target value of K, we experimentally determine the initial value K_0 and the step size ΔK.

Algorithm 1 Curriculum-based Adversarial Learning Scheme

Initialize Policy π_{rl}, Experience Pool M, $K = K_0$, ΔK, K_{target}, R_{best}, Testing interval $T_{testing}$, c
for *episode* $i = 0, 1, 2, 3, \ldots, N$ **do**
  for *step* $t = 0, 1, 2, 3, \ldots, T$ **do**
    for $AV = 0, 1, 2, \ldots, n \in \mathcal{V}$ **do**
      if *AV is attacked* **then**
        $s_{\text{noise}} = s_t + K \cdot N(0, \sigma^2)$, and select action a_t via $\pi_{rl}(a_t|s_{\text{noise}}; \theta)$
      end
      else
        Select action a_t via $\pi_{rl}(a_t|s_t; \theta)$
      end
      Interact with the traffic environment and update M
    end
    Train with the adversarial samples from M to update the parameter θ of π_{rl}
  end
  if $(episode\, i + 1)$ % $T_{testing} == 0$ **then**
    Conduct c rounds of random testing with different numbers of AV respectively
    Calculate the average reward R_i
    if $R_i > R_{best}$ **then**
      $R_{best} = R_i$ and $K = K + \Delta K$
      $K = max(K_0, min(K, K_{target}))$
    end
  end
end

Algorithm 1 outlines the curriculum-based adversarial learning scheme in detail. Following every certain interval $T_{testing}$, we conduct rounds of random testing. Then calculate the average reward R_i of all rounds. If the current reward R_i exceeds the best reward of all previous tests R_{best}, it demonstrates favorable training conditions. Consequently, the noise intensity K is increased by ΔK, indicating that the current training environment is more adversarial. And when K reaches the target value, it no longer increases.

5. Experiment and Evaluation

In this section, to evaluate the effectiveness of the proposed decision-making method, we utilized the highway-env (Leurent et al., 2018) simulator to establish a highway driving scenario, which focuses on a multi-lane highway on-ramp merging, with modifications derived from highway-env.

5.1. Simulation Settings

5.1.1. Experimental Environment Setup

In this scenario, vehicles are categorized into AVs and HDVs. HDVs are controlled by the intelligent driver model (IDM) (Treiber et al., 2000) for longitudinal acceleration and MOBIL (Kesting et al., 2007) model for lateral lane changing respectively, thus serving as part of the complex environment.

The first step is to determine the size of the observation space for the AV, which means determining $N_{\mathcal{N}_i}$ mentioned in 3.2. In most research about single-lane highway on-ramp merging (Chen et al., 2023), $N_{\mathcal{N}_i}$ is typically set to 5. Considering the larger space and higher number of vehicles in the multi-lane highway on-ramp merging, we investigated the best value of $N_{\mathcal{N}_i}$. The experimental results are shown in Fig. 3.

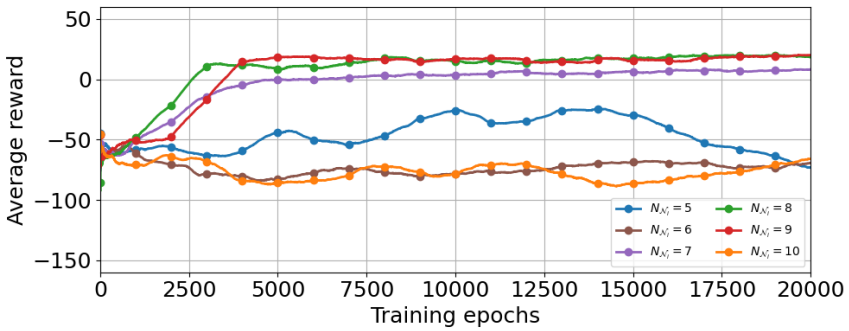


Figure 3: The performance of MAPPO(GAE) at different values of $N_{\mathcal{N}_i}$

As can be seen from Fig. 3, considering the cost of computational resources at the same time, the benchmark algorithm MAPPO performs relatively best when $N_{\mathcal{N}_i} = 8$, so $N_{\mathcal{N}_i} = 8$ is chosen for further experiments. Fig. 4 shows that AVs and HDVs are randomly generated at designated points in the first 220 meters before the merge lane. There is a lateral distance of 40 meters between each point, based on actual traffic conditions. The generated positions

are subject to some random noise. Table 1 shows additional configurations specific to this environment.

Table 1: Environment parameters setting

Traffic simulator terms	**Value**
Total lane length	520m
Merge lane length	100m
Initial speed	25m/s with random noise
Speed range of AV	[10,30]m/s
Number of AV	$4 \sim 6$
Number of HDV	$4 \sim 6$
Simulation frequency	15Hz
Policy frequency	5Hz

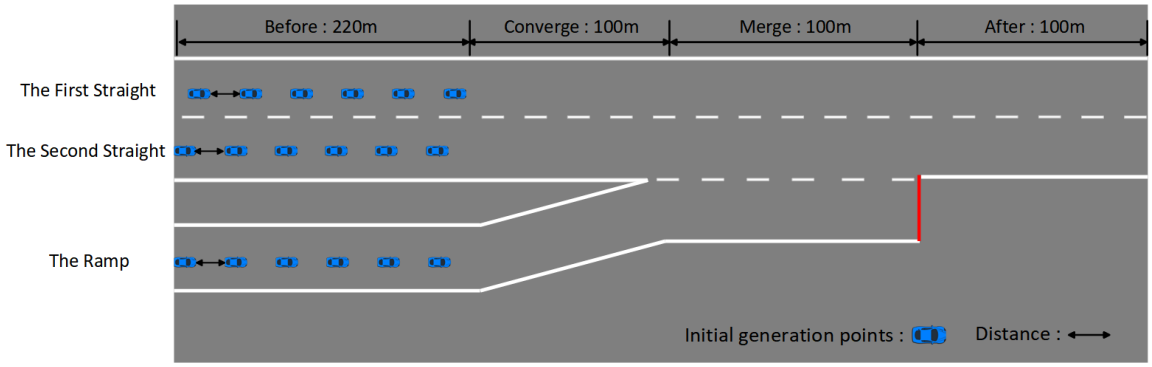


Figure 4: Simulation settings of the environment

5.1.2. Training Parameters Setting

Due to the strict safety requirements of autonomous driving, in this paper, we only consider scenarios where the observation of a single AV is under attack. The parameters utilized in the proposed algorithm during training are shown in Appendix A. In addition, we trained the MAPPO with GAE and the MAACKTR (Wu et al., 2017) as benchmark algorithms.

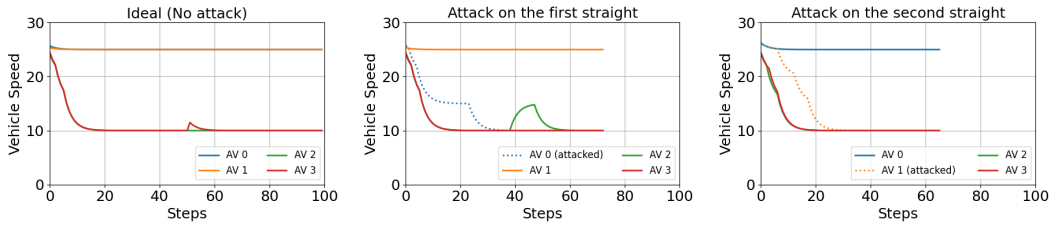
5.2. Analysis and Comparison of Experimental Results

5.2.1. Analysis of Test Experiments to Determine the Best Adversarial Sample

Adversarial learning enhances system robustness through iterative interactions with adversarial environments, thus determining an appropriate adversarial environment is essential. It is hypothesized that attacks occurring on different lanes may result in varying impacts on the MAS. Therefore, we test the AV against noise attacks on two main straight highways respectively and compared it with the test in the ideal environment. To better analyze the impact when the AV on the main straight road is attacked, we examine the trajectory of speeds of different AVs.

As shown in Fig. 5, first according to the generation rule of the environment, AV 0 is generated in the first straight, AV 1 is generated in the second straight, and the rest of the AVs are generated on the ramp. Moreover, step = 100 represents that the AVs coordinate with each other to complete the merging task safely. In Fig. 5 (a), AV 0 and AV 1 travel at a constant speed, and other AVs on the ramp slow down to avoid collisions, finally all AVs securely pass through merged sections with no noise attack. In Fig. 5 (b) and Fig. 5 (c), the speeds of AV 0 and AV 1 decrease when subjected to the noise attack respectively, which affects the normal traveling of other AVs, and finally collisions occur around 70 steps. These results show the attack effectively affects the performance of the MAS by influencing the decisions of the AV.

Next, we test the benchmark algorithm trained in an ideal environment over three different random seeds to further determine the best adversarial sample. From the test results in Table 2, the noise attack on the first straight results in the task success rate decrease of more than 20% and an average speed decrease of over 1 m/s compared to a scenario without the noise attack, which is more detrimental than the attack occurring on the second straight. To train AVs capable of developing stronger adversarial strategies, we choose to conduct training in scenarios where the attack occurs on the first straight road, treating it as the adversarial environment.



(a) (b) (c)

Figure 5: The trajectory of the speed of AVs for different tests in the same episode

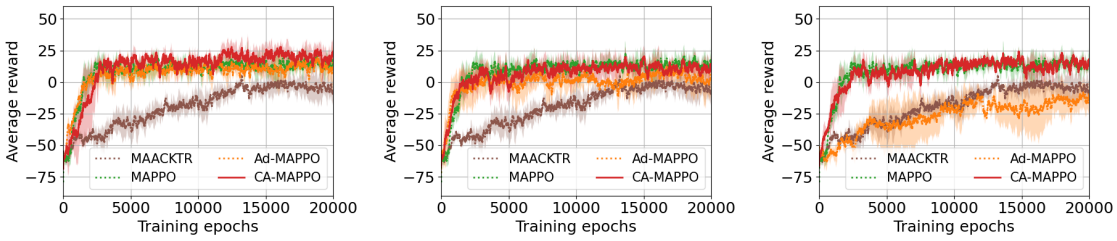
Table 2: The testing experiment to determine the optimal adversarial samples under the noisy environment when $K = 1$

Test Scenarios	Metrics	MAPPO
Ideal (No attack)	success rate	0.72
	avg. speed [m/s]	17.10
	avg. reward	**12.94**
Attack on the first straight	success rate	0.50
	avg. speed [m/s]	15.97
	avg. reward	**-16.30**
Attack on the second straight	success rate	0.66
	avg. speed [m/s]	15.70
	avg. reward	**-2.49**

5.2.2. Comparative experiments and analysis of training and testing situations under different noise intensities

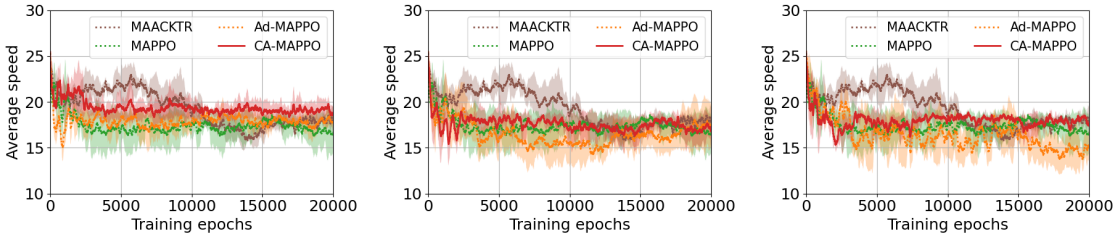
To further validate the effectiveness of the proposed method, we compare CA-MAPPO with benchmark algorithms. It is worth noting that benchmark algorithms are all trained in an ideal environment. Additionally, we conduct an ablation study with Ad-MAPPO which involves adversarial learning only, without curriculum learning. The experiments are conducted under three different noise intensities and the training results are shown in Fig. 6 and Fig. 7.

As illustrated in Fig. 6, when the noise intensity $K = 1$, the average reward achieved by CA-MAPPO is comparable to that of MAPPO, much higher than that of MAACKTR and slightly higher than that of Ad-MAPPO. As the noise intensity increases, the advantage of CA-MAPPO over Ad-MAPPO becomes more prominent. When $K = 5$, the average reward of Ad-MAPPO falls below that of MAACKTR, while the average reward of CA-MAPPO still remains at a high level. Fig. 7 reflects the convergence of the average speed during the training process. To achieve convergence, the average speed of AVs will first decrease thus pursuing a higher success rate of the task. The average speed of AVs under the CA-MAPPO algorithm with $K = 1, 3, 5$ all stably converge to approximately 18 m/s, whereas Ad-MAPPO exhibits instability and converges to around only 15 m/s only when $K = 5$. The above results demonstrate incorporating curriculum learning methods significantly improved the system performance during the training process.



(a)$K = 1$ (b)$K = 3$ (c)$K = 5$

Figure 6: The average reward curves of different algorithms during training over three random seeds



(a)$K = 1$ (b)$K = 3$ (c)$K = 5$

Figure 7: The average speed curves of different algorithms during training over three random seeds

Next, we test all trained algorithms in various testing environments. The tests are conducted under three scenarios: one involves normal conditions without any noise attack, the second with the attack on the AV's observations on the first straight, and the third with the attack on the AV's observations on the second straight. As the testing results are shown from Table 3 to Table 5, in the ideal environment, both under CA-MAPPO and MAPPO, AVs achieve a task success rate of more than 70%, and CA-MAPPO has an advantage in terms of average speed, which can reach 19 m/s under $K = 1$. With the attack on the AV's observations, CA-MAPPO can maintain comparable or even higher performance levels than ideal environments. In contrast, other algorithms experience significant degradation in system performance when faced with this scenario. And it is worth noting that Ad-MAPPO shows a significant performance drop under $K = 5$, compared to its previous performance, highlighting the importance of introducing the curriculum learning approach. Across 3 levels of K, CA-MAPPO consistently demonstrates a significant performance advantage over other algorithms.

Table 3: The average testing performance over three random seeds for 100 episodes under $K = 1$

Test Scenarios	Metrics	MAACKTR	MAPPO	Ad-MAPPO	CA-MAPPO(ours)
Ideal (No attack)	success rate	0.71	0.72	0.67	0.72
	avg. speed [m/s]	17.94	17.1	17.95	19.18
	avg. reward	**-3.04**	**12.94**	**5.60**	**16.67**
Attack on the first straight	success rate	0.58	0.50	0.63	0.74
	avg. speed [m/s]	17.63	15.97	18.17	19.09
	avg. reward	**-18.82**	**-16.30**	**3.46**	**17.41**
Attack on the second straight	success rate	0.35	0.66	0.63	0.71
	avg. speed [m/s]	19.78	15.70	18.42	19.16
	avg. reward	**-38.12**	**-2.49**	**0.86**	**7.00**

Table 4: The average testing performance over three random seeds for 100 episodes under $K = 3$

Test Scenarios	Metrics	MAACKTR	MAPPO	Ad-MAPPO	CA-MAPPO(ours)
Ideal (No attack)	success rate	0.71	0.72	0.69	0.71
	avg. speed [m/s]	17.94	17.10	16.74	17.28
	avg. reward	**-3.04**	**12.94**	**5.51**	**10.98**
Attack on the first straight	success rate	0.60	0.54	0.72	0.71
	avg. speed [m/s]	17.51	15.88	15.59	17.18
	avg. reward	**-18.33**	**-13.58**	**5.23**	**10.99**
Attack on the second straight	success rate	0.39	0.66	0.73	0.74
	avg. speed [m/s]	19.64	15.69	15.53	17.16
	avg. reward	**-37.27**	**-2.07**	**4.89**	**12.84**

Table 5: The average testing performance over three random seeds for 100 episodes under $K = 5$

Test Scenarios	Metrics	MAACKTR	MAPPO	Ad-MAPPO	CA-MAPPO(ours)
Ideal (No attack)	success rate	0.71	0.72	0.59	0.70
	avg. speed [m/s]	17.94	17.10	16.05	18.24
	avg. reward	**-3.04**	**12.94**	**-14.68**	**12.42**
Attack on the first straight	success rate	0.60	0.35	0.62	0.73
	avg. speed [m/s]	17.51	16.56	14.99	18.14
	avg. reward	**-19.03**	**-27.45**	**-14.59**	**16.00**
Attack on the second straight	success rate	0.37	0.66	0.53	0.70
	avg. speed [m/s]	19.55	15.67	16.10	18.27
	avg. reward	**-39.32**	**-2.22**	**-26.56**	**14.62**

6. Conclusion

This paper presents a novel MARL algorithm CA-MAPPO, for the multi-lane on-ramp merging task of AVs on highways. The algorithm accounts for the observation perturbation of AV and is compared with SOTA MARL benchmarks MAPPO and MAACKTR through comprehensive experiments. Experiment results show that the proposed method enables AVs to achieve a success rate of over 70% and an average speed no lower than 17 m/s, significantly better than the performance of the benchmarks in the scenarios under noise attacks of varying intensity significantly and in ideal environments with no noise attack. Furthermore, we have explored the factors affecting the performance of the CA-MAPPO algorithm in Appendix B and find that the initial noise intensity K_0 plays a critical role.

However, more comprehensive considerations are essential for the successful transition from simulation to real-world applications. In the future, we will explore creating reliable fault-tolerant mechanisms for handling more AVs in complex environments, including diverse severe weather simulations.

References

TJ Ayres, L Li, David Schleuning, and D Young. Preferred time-headway of highway drivers. In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585)*, pages 826–829. IEEE, 2001.

Dong Chen, Mohammad R Hajidavalloo, Zhaojian Li, Kaian Chen, Yongqiang Wang, Longsheng Jiang, and Yue Wang. Deep multi-agent reinforcement learning for highway on-ramp merging in mixed traffic. *IEEE Transactions on Intelligent Transportation Systems*, 2023.

Jishiyu Ding, Li Li, Huei Peng, and Yi Zhang. A rule-based cooperative merging strategy for connected and automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 21(8):3436–3446, 2019.

Shilp Dixit, Umberto Montanaro, Mehrdad Dianati, David Oxtoby, Tom Mizutani, Alexandros Mouzakitis, and Saber Fallah. Trajectory planning for autonomous high-speed over-

taking in structured environments using robust mpc. *IEEE Transactions on Intelligent Transportation Systems*, 21(6):2310–2323, 2019.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

Jun Guo, Yonghong Chen, Yihang Hao, Zixin Yin, Yin Yu, and Simin Li. Towards comprehensive testing on the robustness of cooperative multi-agent reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 115–122, 2022.

Xiangkun He, Baichuan Lou, Haohan Yang, and Chen Lv. Robust decision making for autonomous vehicles at highway on-ramps: A constrained adversarial reinforcement learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 24(4):4103–4113, 2022.

Xiangwang Hu and Jian Sun. Trajectory optimization of connected and autonomous vehicles at a multilane freeway merging area. *Transportation Research Part C: Emerging Technologies*, 101:111–125, 2019.

Arne Kesting, Martin Treiber, and Dirk Helbing. General lane-changing model mobil for car-following models. *Transportation Research Record*, 1999(1):86–94, 2007.

Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zając, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, et al. Google research football: A novel reinforcement learning environment. 34(04):4501–4510, 2020.

Edouard Leurent et al. An environment for autonomous driving decision-making. 2018.

Jieyu Lin, Kristina Dzeparoska, Sai Qian Zhang, Alberto Leon-Garcia, and Nicolas Papernot. On the robustness of cooperative multi-agent reinforcement learning. In *2020 IEEE Security and Privacy Workshops (SPW)*, pages 62–68. IEEE, 2020.

Thomy Phan, Thomas Gabor, Andreas Sedlmeier, Fabian Ritz, Bernhard Kempter, Cornel Klein, Horst Sauer, Reiner Schmid, Jan Wieghardt, Marc Zeller, et al. Learning and testing resilience in cooperative multi-agent systems. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1055–1063, 2020.

Thomy Phan, Lenz Belzner, Thomas Gabor, Andreas Sedlmeier, Fabian Ritz, and Claudia Linnhoff-Popien. Resilient multi-agent reinforcement learning with adversarial value decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11308–11316, 2021.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *International Conference on Machine Learning*, pages 2817–2826. PMLR, 2017.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*, 2017.

Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical review E*, 62(2):1805, 2000.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575 (7782):350–354, 2019.

Yuhuai Wu, Elman Mansimov, Roger B Grosse, Shun Liao, and Jimmy Ba. Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation. *Advances in neural information processing systems*, 30, 2017.

Kai Yang, Xiaolin Tang, Sen Qiu, Shufeng Jin, Zichun Wei, and Hong Wang. Towards robust decision-making for autonomous driving on highway. *IEEE Transactions on Vehicular Technology*, 72(9):11251–11263, 2023.

Xinfeng Zhang, Lin Wu, Huan Liu, Yajun Wang, Hao Li, and Bin Xu. High-speed ramp merging behavior decision for autonomous vehicles based on multi-agent reinforcement learning. *IEEE Internet of Things Journal*, 2023.

Jian Zhao, Youpeng Zhao, Weixun Wang, Mingyu Yang, Xunhan Hu, Wengang Zhou, Jianye Hao, and Houqiang Li. Coach-assisted multi-agent reinforcement learning framework for unexpected crashed agents. *Frontiers of Information Technology & Electronic Engineering*, 23(7):1032–1042, 2022.