Robust Lane Change Decision Making for Autonomous Vehicles: An Observation Adversarial Reinforcement Learning Approach

SUBMISSION DATE / POSTED DATE

29-09-2022 / 02-11-2022

CITATION

He, Xiangkun; Yang, Haohan; Hu, Zhongxu; Lv, Chen (2022): Robust Lane Change Decision Making for Autonomous Vehicles: An Observation Adversarial Reinforcement Learning Approach. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.21229319.v1

DOI

10.36227/techrxiv.21229319.v1

Robust Lane Change Decision Making for Autonomous Vehicles: An Observation Adversarial Reinforcement Learning Approach

Xiangkun He, *Member, IEEE*, Haohan Yang, Zhongxu Hu, *Member, IEEE*, and Chen Lv, *Senior Member, IEEE*

***Abstract*—Reinforcement learning holds the promise of allowing autonomous vehicles to learn complex decision making behaviors through interacting with other traffic participants. However, many real-world driving tasks involve unpredictable perception errors or measurement noises which may mislead an autonomous vehicle into making unsafe decisions, even cause catastrophic failures. In light of these risks, to ensure safety under perception uncertainty, autonomous vehicles are required to be able to cope with the worst case observation perturbations. Therefore, this paper proposes a novel observation adversarial reinforcement learning approach for robust lane change decision making of autonomous vehicles. A constrained observation-robust Markov decision process is presented to model lane change decision making behaviors of autonomous vehicles under policy constraints and observation uncertainties. Meanwhile, a black-box attack technique based on Bayesian optimization is implemented to approximate the optimal adversarial observation perturbations efficiently. Furthermore, a constrained observation-robust actor-critic algorithm is advanced to optimize autonomous driving lane change policies while keeping the variations of the policies attacked by the optimal adversarial observation perturbations within bounds. Finally, the robust lane change decision making approach is evaluated in three stochastic mixed traffic flows based on different densities. The results demonstrate that the proposed method can not only enhance the performance of an autonomous vehicle but also improve the robustness of lane change policies against adversarial observation perturbations.**

***Index Terms*—Autonomous vehicle, lane change decision making, robust decision making, reinforcement learning, adversarial attack.**

I. Introduction

In recent years, autonomous driving has attracted significant attention since its promise is profound to revolutionize automobile industry [1], [2]. However, safety remains a major challenge for the development of autonomous vehicles [3], [4], [5]. Undesirable decision making behaviors of autonomous vehicles may endanger life safety and cause enormous economic loss. As one of the most advanced artificial intelligence technologies, reinforcement learning (RL) has achieved a success in fulfilling a series of challenging decision making tasks (e.g., Go and StarCraft II) [6], [7], [8]. Hence, applying RL to decision making task of autonomous driving has become a hot topic for researchers [9].

While existing RL based decision making methods of autonomous vehicles have achieved many compelling results [10], [11], [12], [13], the real-world driving tasks involve unavoidable measurement errors or sensor noises which may mislead an autonomous vehicle into making suboptimal decisions, even cause catastrophic failures. In light of these risks, autonomous vehicles are required to ensure that their decision making systems can handle the natural observation uncertainties from sensing and perception system, especially adversarial perturbations. However, few researches concern and cope with the aforementioned challenge.

Therefore, in this paper, a novel observation adversarial RL (OARL) approach for robust lane change decision making is proposed to improve the performance of an autonomous vehicle while enhancing the robustness of driving policies against adversarial observation perturbations. The main contributions of this paper are summarized as follows:

- A constrained observation-robust Markov decision process (COR-MDP) is advanced to model lane change decision making behaviors of an autonomous vehicle under policy constraints and observation perturbations. Meanwhile, a black-box attack technique with Bayesian optimization is implemented to approximate the optimal adversarial observation perturbations efficiently.
- A constrained observation-robust actor-critic (COR-AC) algorithm is presented to optimize lane change policies and minimize the Jensen–Shannon (JS) divergence based average variation distance of the policies attacked by the optimal adversarial observation perturbations.

Three testing cases with different traffic flow densities are implemented to evaluate the performance of our robust lane change decision making approach through simulation of urban mobility (SUMO) [14], [15]. The results demonstrate that the proposed OARL method is effective and outperforms the competitive baselines.

The rest of this paper is arranged as follows. The related works with respect to this paper are reviewed in Section II. The proposed OARL method for robust decision making of autonomous vehicles is illustrated in Section III. Implementation details of our method are provided in Section IV. The evaluation results and analyses are discussed in Section V. The conclusions of this paper are made in Section VI.

This work was supported by Schaeffler Hub for Advanced Research (SHARE@NTU) under program Smart Mechatronic Lab for Industrial Collaborative Robotics in Manufacturing (No. I2001E0067 (Schaeffler) - PA5 and I2001E0067 (IAF-ICP) - PA5). *(Corresponding author: Chen Lv)*

X. He, H. Yang, Z. Hu and C. Lv are with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798. (e-mail: xiangkun.he@ntu.edu.sg, haohan001@e.ntu.edu.sg, zhongxu.hu@ntu.edu.sg, lyuchen@ntu.edu.sg)

II. Related Work

According to different driving behaviors (e.g., lane change, acceleration or deceleration) or tasks (e.g., overtaking or ramp merging) in existing related studies, RL based decision making of autonomous vehicles can roughly be divided into three categories: longitudinal, lateral and coordinated decision making [9]. RL based longitudinal decision-making methods generally adopt RL algorithm to determine the speed modes of autonomous vehicles, such as keeping, acceleration and deceleration [11], [16], [17], [18].

A. Reinforcement Learning based Lateral Decision Making for Autonomous Vehicles

RL based lateral decision making approaches of autonomous vehicles mostly employ RL algorithm to learn lane change behaviors or select target lanes. One popular paradigm is the lateral decision making schemes with the deep Q-network (DQN) or its variants. A lane change decision-making framework for autonomous vehicles is developed to learn risk sensitive driving policies using risk-awareness prioritized replay DQN in [12]. A lane change decision making method is presented for autonomous vehicles through DQN with safety verification in [19]. A harmonious lane-changing decision making approach based on DQN is advanced to improve overall traffic efficiency in [20]. A DQN method with rule-based constraints is developed for lane change decision making of autonomous vehicles in [21]. A lane change decision-making approach for autonomous vehicles is developed via double DQN with the structure of Deep Sets in [22]. A lane change decision making method based on partial observed Markov decision process and DQN is introduced for autonomous vehicles in [23]. The above methods are simple but effective. Moreover, combined with rule based constraints, the driving safety of autonomous vehicles can be guaranteed. However, these schemes can not find the optimal driving policies necessarily.

In addition to the DQN based paradigms, there are the autonomous driving lateral decision making approaches with other RL algorithms. A proximal policy optimization (PPO) based lane change decision-making method is presented for autonomous drving in [13]. A multi-objective approximate policy iteration algorithm is proposed to implement lane change decision making of an autonomous vehicle in [24]. A lane change decision-making scheme based on attention-based hierarchical deep RL is proposed for autonomous vehicles in [25]. Although these methods may achieve better performance than the DQN based schemes, the robust decision-making problem of autonomous vehicles is not studied among them.

B. Reinforcement Learning based Coordinated Decision Making for Autonomous Vehicles

RL based coordinated decision making schemes usually leverage RL algorithm to determine longitudinal and lateral driving behaviors of autonomous vehicles simultaneously. A longitudinal and lateral coordinated decision making approach based on AlphaGo Zero algorithm is developed for autonomous vehicles in [26]. The requested speed and target lane can be determined by the five decision making behaviors of RL agent simultaneously. A DQN based decision making method is advanced, which can simultaneously determine discrete speed modes and lane change behaviors of an autonomous vehicle in [27]. An optimization embedded RL with actor–critic framework is presented to determine longitudinal and lateral coordinated decision making behaviors for autonomous vehicles in [28]. A coordinated decision making method based on deep deterministic policy gradient algorithm is developed to determine throttle and steering maneuvers for autonomous driving in [29]. Unfortunately, the above methods mostly assume that the state observations are free of unexpected perturbations. Such assumption can hardly hold in real-world scenarios.

III. Observation Adversarial Reinforcement Learning for Robust Decision Making

A. Robust Lane Change Decision Making Framework for Autonomous Vehicles

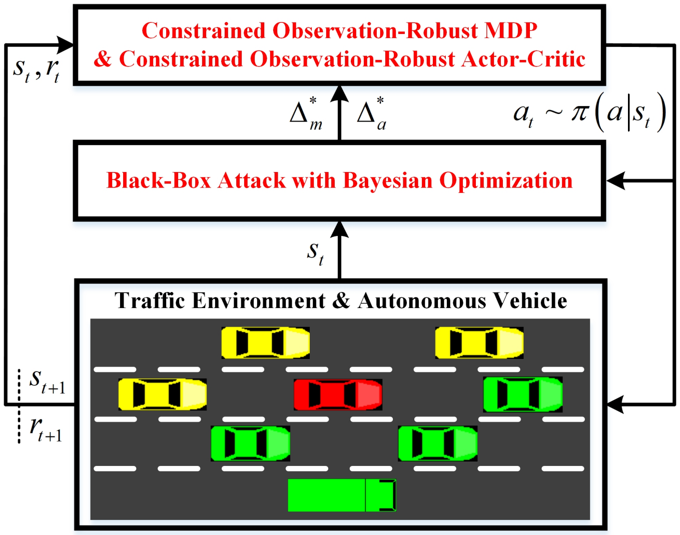

Constrained Observation-Robust MDP
& Constrained Observation-Robust Actor-Critic
s_t, r_t
Δ_m^*
Δ_a^*
$a_t \sim \pi(a|s_t)$
Black-Box Attack with Bayesian Optimization
s_t
Traffic Environment & Autonomous Vehicle
s_{t+1}
r_{t+1}

Fig. 1. Framework of the proposed robust lane change decision-making approach for autonomous driving.

Since the existing lane change decision-making framework of autonomous vehicles do not take into account perception uncertainty mostly, the robust lane change decision making framework with OARL algorithm is proposed to cope with the adversarial perturbations on state observations in autonomous driving, as shown in Fig. 1. Ego vehicle is red, and it is an autonomous vehicle. The longitudinal decision-making of the ego vehicle is implemented by SUMO based intelligent driving model (IDM). The vehicles of other colors are social vehicles, and the longitudinal and lateral driving behaviors of the social vehicles are determined by the IDM of SUMO. The social vehicles can perform lane change maneuvers via the LC2013 model [30] in SUMO. Moreover, the output of the ego vehicle is discrete, which includes lane keeping, left lane changing and right lane changing.

Our RL autonomous driving agent seeks to maximize the expected return while satisfying the policy constraints. In

Fig. 1, the block with respect to COR-MDP and COR-AC is used for optimizing robust driving policy and interacting with the environment. Its input includes state s, reward r and the optimal adversarial observation perturbations Δ^*. t denotes time step. The optimal adversarial observation perturbations Δ^* contains the optimal adversarial multiplicative-perturbation Δ_m^* and the optimal adversarial additive-perturbation Δ_a^*. The output is the action a based on the policy $\pi(a|s)$.

The block with regard to the black-box attacks is employed to approximate the optimal adversarial perturbations. The input of this block includes state s and the policy $\pi(a|s)$, and its output is the optimal adversarial perturbation. Additionally, the block associated with the environment is leveraged to generate state s and reward r. Its input is the action a based on the policy $\pi(a|s)$, and the output contains state s and reward r.

B. Constrained Observation-robust Markov Decision Process

To model the decision making behaviors of RL based autonomous driving agent under policy constraints and observation perturbations, the proposed COR-MDP is introduced in this section.

Definition 1: A COR-MDP can be characterized via a 7-tuple $(\mathcal{S}, \mathcal{A}, p, r, c, \Delta, \gamma)$. $\mathcal{S}$ is the set of states called the state space. $\mathcal{A}$ is the set of actions called the action space. p is the transition probability distribution of the next state $s' \in \mathcal{S}$ given the current state $s \in \mathcal{S}$ and action $a \in \mathcal{A}$. $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ represents the reward function, and c denotes the constraint function. Δ indicates the observation perturbation. $\gamma \in (0, 1)$ is the discount factor.

COR-MDP attempts to solve the following problem:

$$\begin{aligned} &\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right], \\ &\text{s.t.} \;\; \mathbb{E}\left[c(s, s', \Delta)\right] \le \epsilon, \end{aligned} \tag{1}$$

where T is timestep, and ϵ is an expected minimum deviation.

C. Black-Box Attack with Bayesian Optimization

In this section, the black-box attack based on Bayesian optimization is implemented to approximate the optimal adversarial observation perturbations.

Bayesian optimization is a black-box optimization algorithm with Bayes theorem [31]. This approach works by building a probabilistic model of the objective function, called the surrogate model, that is then searched efficiently through an acquisition function before candidate samples are determined for evaluation on the real objective function [32], [33].

The JS divergence is a symmetrized and smoothed version of the Kullback–Leibler (KL) divergence [34], [35]. But more importantly, JS divergence has a finite value which is bounded by 1 for two probability distributions. Hence, JS divergence is employed to measure average variation distance of the policies attacked by the observation perturbations. The optimization objective with JS divergence can be designed as:

$$\begin{aligned} c(s, s', \Delta) &= D_{JS}\left(\pi(a|s)||\pi(\tilde{a}|\tilde{s})\right) + D_{JS}\left(\pi(a|s')||\pi(\tilde{a}|\tilde{s'})\right) \\ &= \frac{1}{2} D_{KL}\left(\pi(a|s)||m\right) + \frac{1}{2} D_{KL}\left(\pi(\tilde{a}|\tilde{s})||m\right) \\ &+ \frac{1}{2} D_{KL}\left(\pi(a|s')||m'\right) + \frac{1}{2} D_{KL}\left(\pi(\tilde{a}|\tilde{s'})||m'\right), \end{aligned} \tag{2}$$

where D_{JS} represents the distance based on JS divergence, D_{KL} denotes KL divergence, and

$$\begin{cases} \tilde{s} = \Delta^m s + \Delta^a, \\ \tilde{s'} = \Delta^m s' + \Delta^a, \end{cases} \tag{3}$$

$$\begin{cases} m = \frac{1}{2}(\pi(a|s) + \pi(\tilde{a}|\tilde{s})), \\ m' = \frac{1}{2}(\pi(a|s') + \pi(\tilde{a}|\tilde{s'})), \end{cases} \tag{4}$$

where $\tilde{a}$, $\tilde{s}$ and $\tilde{s'}$ are the action, the state and the next state perturbed by observation perturbations respectively.

Therefore, our black-box attack approach is formulized as:

$$\begin{aligned} &\Delta^* \in \arg\max_{\Delta} \;\; \mathbb{E}[c(s, s', \Delta)], \\ &\text{s.t.} \;\; \left|\Delta_m - \Delta_m^0\right| \le \delta_m, \;\; \left|\Delta_a - \Delta_a^0\right| \le \delta_a, \end{aligned} \tag{5}$$

where $\Delta = [\Delta_m \;\; \Delta_a]$ represents observation perturbation, Δ_m and Δ_a are the multiplicative-perturbation and the additive-perturbation, Δ_m^0 and Δ_a^0 are the reference values of the multiplicative-perturbation and the additive-perturbation, δ_m and δ_a are the desired bounds of the multiplicative-perturbation and the additive-perturbation respectively.

Algorithm (1) outlines the black-box attack method using Bayesian optimization. The acquisition function is designed through upper confidence bound (UCB) [36]. Additionally, Gaussian process is leveraged to built surrogate model for the optimization objective in our algorithm.

Algorithm 1 Black-box attack with Bayesian optimization

```
for $i = 1, 2, ..., I$ do
    Find new adversarial observation perturbation $\Delta^i = [\Delta_m^i \;\; \Delta_a^i]$ via optimizing the acquisition function UCB$(\cdot)$ over Gaussian process model:
        $\begin{cases} \Delta^i = \arg\max_{\Delta} \;\; \text{UCB}(\Delta|\mathcal{M}_{1:i-1}), \\ \text{s.t.} \;\; \left|\Delta_m - \Delta_m^0\right| \le \delta_m, \;\; \left|\Delta_a - \Delta_a^0\right| \le \delta_a. \end{cases}$
    Compute the objective function $\mathbb{E}[c(s, s', \Delta^i)]$.
    Augment data to memory $M$:
    $\mathcal{M}_{1:i} = \mathcal{M}_{1:i-1} \cup \left\{\Delta^i, \mathbb{E}[c(s, s', \Delta^i)]\right\}$.
    Update the Gaussian process model.
end for
```

D. Constrained Observation-Robust Actor-Critic

To learn the robust optimal lane change policy, the proposed COR-AC algorithm is introduced in this section. COR-AC attempts to solve the following optimization problem:

$$\begin{aligned} &\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right], \\ &\text{s.t.} \;\; \mathbb{E}\left[c(s, s', \Delta^*)\right] \le \epsilon, \end{aligned} \tag{6}$$

where $\Delta^* = [\Delta_m^* \quad \Delta_a^*]$ represents the optimal adversarial observation perturbation.

A policy iteration (PI) scheme is employed to solve COR-MDP, which is called constrained observation-robust PI (COR-PI). COR-PI consists of policy evaluation and policy improvement, and they are iteratively updated until convergence.

According to Lagrange duality theory [37], the Lagrange function of the optimization problem (6) can be derived as:

$$L(\pi, \lambda) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t) + \lambda(\epsilon - c(s, s', \Delta^*))\right], \tag{7}$$

where λ is dual variable of RL agent.

1) Constrained Observation-Robust Policy Evaluation: The action-value function $Q^\pi(\cdot)$ with adversarial observation perturbations can be learned under a fixed policy iteratively, starting from any action-value function $Q^\pi(\cdot) : \mathcal{S} \to \mathbb{R}^{|\mathcal{A}|}$ and repeatedly leveraging a modified Bellman backup operator $\mathcal{T}^\pi$ given via:

$$\mathcal{T}^\pi Q^\pi(s_t) := r(s_t, a_t) + \gamma \mathbb{E}[V^\pi(s_{t+1})], \tag{8}$$

where

$$\mathbb{E}[V^\pi(s_{t+1})] = \pi(s_{t+1})^\intercal [Q^\pi(s_{t+1}) - \lambda c(s, s', \Delta^*)] \tag{9}$$

is the expected value function with adversarial observation perturbations. Since the policy model outputs the discrete action distribution, the expectation of value function $V^\pi(\cdot)$ can be calculcated directly.

To speed up training, COR-AC algorithm adopts two parameterized action-value functions with network parameters ϕ^z, $z \in \{1, 2\}$. The action-value function parameters can be updated via minimizing the following loss function of critic network:

$$J_c(\phi^z) = \mathop{\mathbb{E}}_{\substack{a_{t+1} \sim \pi \\ Ts \sim \mathcal{D}}} \left[\|y_t - Q^\pi(s_t; \phi^z)\|_2^2 \right], \tag{10}$$

where Ts represents transition sampled from replay buffer $\mathcal{D}$, and y_t is the target value of the action-value function in the time step t. To avoid overestimating the value function, the smaller one of two $Q^\pi(\cdot)$ values is used to train critic network. With Eq. (8) and Eq. (9), y_t can be defined as:

$$\begin{aligned} y_t = r(s_t, a_t) + \gamma \pi(s_{t+1})^\intercal \big[&\min_{z \in \{1,2\}} \hat{Q}^\pi(s_{t+1}; \bar{\phi}^z) \\ &- \lambda c(s, s', \Delta^*) \big], \end{aligned} \tag{11}$$

where $\hat{Q}^\pi(\cdot)$ is the target action-value function, and $\bar{\phi}^z$ is the network parameter of the target action-value function. The network parameters of target action-value function can be updated once per parameterized action-value function update via polyak averaging:

$$\bar{\phi}^z \leftarrow \rho \bar{\phi}^z + (1 - \rho) \phi^z, \tag{12}$$

where ρ is a hyperparameter between 0 and 1.

2) Constrained Observation-Robust Policy Improvement: In COR-PI, policy improvement designates optimizing and updating the policies of RL agent. The RL agent attempts to maximize the expected return of the policy while satisfying the nonlinear constraint $c(\cdot)$.

With Eq. (7), the Lagrange dual function can be written as:

$$\begin{aligned} \bar{L}(\lambda) &= \max_\pi L(\pi, \lambda) \\ &= \max_\pi \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t) + \lambda(\epsilon - c(s, s', \Delta^*))\right]. \end{aligned} \tag{13}$$

Furthermore, the Lagrange dual problem associated with the problem (6) can be represented as:

$$\begin{aligned} \min_{\lambda \geq 0} \bar{L}(\lambda) &= \min_{\lambda \geq 0} \max_\pi L(\pi, \lambda) \\ &= \min_{\lambda \geq 0} \max_\pi \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t) + \lambda(\epsilon - c(s, s', \Delta^*))\right]. \end{aligned} \tag{14}$$

The optimal policy π^* and the optimal dual variable λ^* can be approximated iteratively. First given a fixed λ, then solve the best policy π^* by maximizing $L(\pi, \lambda)$. Moreover, plug in π^* and find λ^* via minimizing $L(\pi^*, \lambda)$. Therefore, with Eq. (14), the following expressions can be derived:

$$\pi^* = \arg\max_\pi L(\pi, \lambda), \tag{15}$$

$$\lambda^* = \arg\min_{\lambda \geq 0} L(\pi^*, \lambda). \tag{16}$$

The value function $V^\pi(\cdot)$ is implicitly defined through the action-value function $Q^\pi(\cdot)$ and the policy $\pi(\cdot)$ and the constraint $c(\cdot)$. With the double $Q(\cdot)$ trick in Eq. (11), Eq. (9) and Eq. (15), the policy model parameters θ can be optimized via maximizing the following objective function of actor network:

$$J_a(\theta) = \mathop{\mathbb{E}}_{\substack{a_t \sim \pi \\ Ts \sim \mathcal{D}}} \Big[\pi(s_t; \theta)^\intercal \big[\min_{z \in \{1,2\}} Q^\pi(s_t; \phi^z) - \lambda c(s, s', \Delta^*) \big) \big] \Big]. \tag{17}$$

Additionally, with Eq. (16), the dual variables can be updated via minimizing the following loss function:

$$J_d(\lambda) = \mathop{\mathbb{E}}_{\substack{a_t \sim \pi \\ Ts \sim \mathcal{D}}} \big[\pi(s_t; \theta)^\intercal [\lambda(\epsilon - c(s, s', \Delta^*)] \big]. \tag{18}$$

IV. Algorithm Implementation

Algorithm 2 outlines the proposed OARL method in detail. d_t is done signal, and d_t indicates whether the ego vehicle has collided at the time step t. The proposed method can optimize autonomous driving RL agent via the following main procedure. The initial the network parameters of actor and critic are sampled from a random distribution. In each iteration, RL agent first need to collect the data of M timesteps and store them in buffer $\mathcal{D}$. Environment contains the state transition probability and the reward functions to generate the data trajectories. The optimal adversarial observation perturbations Δ^* are found by the black-box attack based on Bayesian optimization. Then the policies of RL agent is updated iteratively.

When the vehicle in front is close and driving slowly, the ego vehicle will perform lane change maneuvers to ensure

Algorithm 2 Observation Adversarial Reinforcement Learning

```
Initialize actor network parameters $\theta$, critic network parameters $\phi^1$ and $\phi^2$.
Initialize target action-value function network parameters $\bar{\phi}^1 \leftarrow \phi^1$, and $\bar{\phi}^2 \leftarrow \phi^2$.
Initialize dual variables $\lambda$ and an empty replay buffer $\mathcal{D}$.
for iteration step $n = 1, 2, \ldots N$ do
    Reset state $s_0$.
    for timestep in the environment $t = 1, 2, \ldots M$ do
        Select action based on the policy: $a_t \sim \pi_\theta(a_t|s_t)$.
        Sample transition from the environment:
            $s_{t+1}, r_t, d_t \sim p(s_{t+1}|s_t, a_t)$.
        Store the transition in the replay buffer:
            $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r_t, s_{t+1}, d_t)\}$.
    end for
    Sample a batch of transitions from replay buffer $\mathcal{D}$.
    Generate the optimal adversarial observation perturbations through Algorithm 1:
        $\Delta^* \leftarrow \begin{cases} \max \quad \mathbb{E}[c(s, s', \Delta)], \\ \text{s.t.} \quad \left|\Delta_m - \Delta_m^0\right| \le \delta_m, \quad \left|\Delta_a - \Delta_a^0\right| \le \delta_a. \end{cases}$
    Update the actor network parameters through Eq. (17):
        $\theta \leftarrow \nabla_\theta J_a(\theta)$.
    Update the critic network parameters through Eq. (10):
        $\phi^1 \leftarrow \nabla_{\phi^1} J_c(\phi^1)$, $\phi^2 \leftarrow \nabla_{\phi^2} J_c(\phi^2)$.
    Update the dual variables through Eq. (18):
        $\lambda \leftarrow \nabla \lambda J_d(\lambda)$.
    Update the target action-value function network parameters through Eq. (12):
        $\bar{\phi}^1 \leftarrow \rho\bar{\phi}^1 + (1-\rho)\phi^1$, $\bar{\phi}^2 \leftarrow \rho\bar{\phi}^2 + (1-\rho)\phi^2$.
end for
```

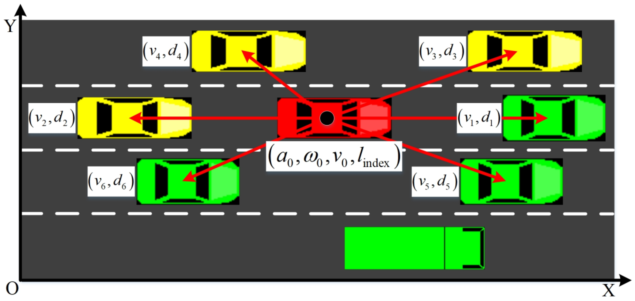


Fig. 2. Illustration of states of the OARL based autonomous driving agent.

Algorithm 3 Reward Function Design for RL Agent

```
**Input:** State and action of RL agent.
$r(\cdot) = v_0/35$.                    ▷ Encourage agent to be more efficiency
if $d_1 < 30$ then
    $r(\cdot) = r(\cdot) - 0.1$.          ▷ Encourage lane change behavior
end if
if $|3.14 \cdot \omega_0/180| > k \cdot \bar{\mu} \cdot g/v_0$ and $v_0 > 30$ then
    $r(\cdot) = r(\cdot) - 0.05$.         ▷ Penalize dynamics instability
end if
if Vehicle changes lane and $v_0 > 20$ then
    $r(\cdot) = r(\cdot) - v_0/350$.      ▷ Penalize high-speed lane change
end if
if Collision occurs then
    $r(\cdot) = r(\cdot) - 0.1$.          ▷ Penalize collision
end if
**Output:** $r(\cdot)$
```

TABLE I

STATE OBSERVED BY AUTONOMOUS DRIVING RL AGENT.

Parameters (Unit)	Definition
a_0 (m/s^2)	Longitudinal acceleration of autonomous vehicle
ω_0 (rad/s)	Yaw rate of autonomous vehicle
v_0 (m/s)	Velocity of autonomous vehicle
v_1 (m/s)	Velocity of vehicle in front in same lane
d_1 (m)	Distance from vehicle in front in same lane
v_2 (m/s)	Velocity of vehicle behind in same lane
d_2 (m)	Distance from vehicle behind in same lane
v_3 (m/s)	Velocity of vehicle in front in left lane
d_3 (m)	Distance from vehicle in front in left lane
v_4 (m/s)	Velocity of vehicle behind in left lane
d_4 (m)	Distance from vehicle behind in left lane
v_5 (m/s)	Velocity of vehicle in front in right lane
d_5 (m)	Distance from vehicle in front in right lane
v_6 (m/s)	Velocity of vehicle behind in right lane
d_6 (m)	Distance of vehicle behind in right lane
l_{index}	Index of lane in which autonomous vehicle is located

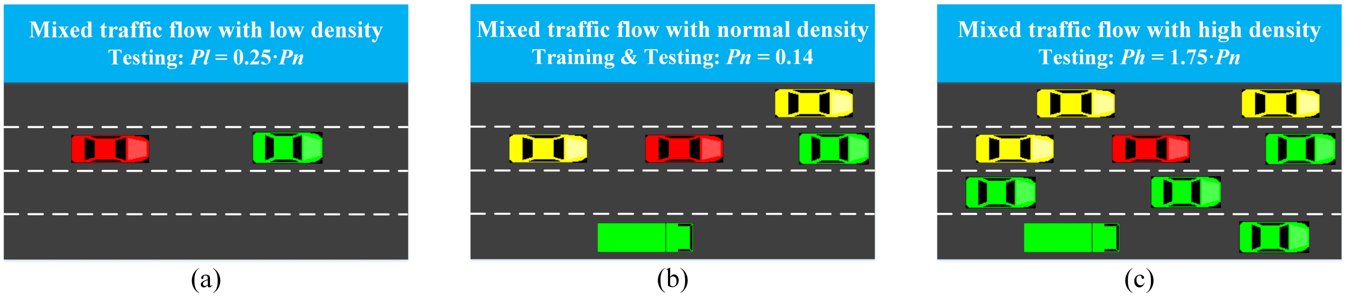
transportation efficiency. Moreover, to implement robust lane change decision-making scheme based on OARL, the state, action and reward of autonomous driving RL agent needs to be defined.

We select the relevant states of the six nearest social vehicles on the lane the ego vehicle is located and on the lanes on both sides of the ego vehicle as the observations of the ego vehicle. When the X-axis distance of the social vehicles on the left or right of the ego vehicle are greater than or equal to the one of the ego vehicle, we consider these social vehicles as left front vehicles or right front vehicles, and vice versa. The state of the autonomous driving agent includes 16 dimensions, and the detailed description is provided in Fig. 2 and Table I. The social vehicles perform lane change maneuvers by the LC2013 model [30] during the training and testing for the RL agent.

Moreover, the action of autonomous driving RL agent is discrete, which includes lane keeping, left lane changing and right lane changing.

One challenge of this work is to learn the robust lane change policies from scratch with no prior knowledge being applied. Therefore, the reward function plays a crucial role for optimizing the polices of the autonomous driving RL agent. Efficiency, comfort and safety are considered to design the reward function.

To encourage the ego vehicle to enhance transport efficiency, the reward function $r(\cdot)$ is designed as $v_0/35$. This means that the autonomous driving agent is able to increase the reward by running at high speed. To avoid the ego vehicle following the front vehicle all the time, if the distance between the ego vehicle and the front vehicle is less than 30 meters, the reward of the agent will be reduced by 0.1. In terms of autonomous driving safety, both of collision and vehicle dynamics stability are considered. According to the upper limit for the desired yaw rate given in [38], if the yaw rate of the ego vehicle exceeds the upper limit $k\bar{\mu}g/v_0$, the reward of the agent will be reduced by 0.05. k is dynamic factor proposed in [39], $\bar{\mu}$ is adhesion coefficient, and g represents gravity acceleration. Additionally, if the ego vehicle is involved in a collision, the reward of the agent will be reduced by 0.1. To avoid frequent lane changes at high speeds, when the ego vehicle performs a lane change manoeuvre at a speed of more than 20 m/s, the reward of the agent will be reduced


Mixed traffic flow with low density
Testing: Pl = 0.25·Pn
Mixed traffic flow with normal density
Training & Testing: Pn = 0.14
Mixed traffic flow with high density
Testing: Ph = 1.75·Pn

(a) (b) (c)

Fig. 3. Schematic diagram of evaluation method using SUMO-based mixed traffic flow with a random number of vehicles.

by $v_0/350$. Algorithm 3 describes the structure of the reward function in detail.

The actor and critic networks are designed via a single fully connected hidden layer, and the layer size is 128. All activation functions in hidden layers are ReLU. The inputs and outputs of the neural networks have 16 and 3 dimensions respectively. The main hyperparameters of our algorithm are provided in Table IV of Appendix.

V. Testing Results and Performance Evaluation

A. Environment

The simulation test based on SUMO platform is implemented to verify the performance of the proposed robust lane change decision-making method for autonomous vehicle in this section. We employ SUMO to create three stochastic mixed traffic flows based on different densities in highway scenarios.

Fig. 3 illustrates our evaluation scheme. P is adopted to denote the probability of emitting a vehicle each second. Pn, Pl and Ph are defined as the probabilities of emitting a vehicle each second in mixed traffic flows based on normal, low and high densities respectively. In addition, Pn, Pl and Ph are set as 0.14, 0.035, 0.245 respectively. Our method and baseline approaches is tested in both training and testing. The policy models are trained and tested based on the mixed traffic flow with normal density. Moreover, the mixed traffic flows with low and high densities are only leveraged to evaluate the policy models.

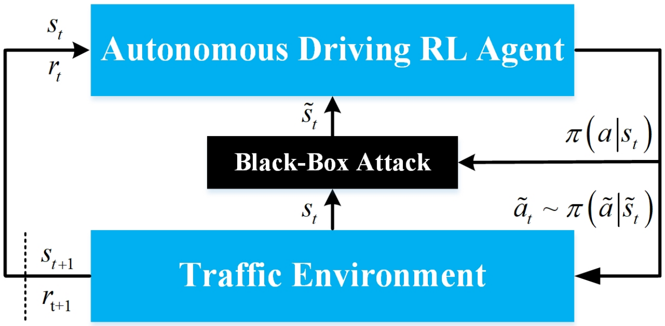


Fig. 4. Illustration of model evaluation scheme. The autonomous driving RL agent observes the perturbed state $\tilde{s}_t$ rather than the state s_t in model testing.

As shown in Fig. 4, unlike the model training stage of OARL, the autonomous driving RL agent observes the state $\tilde{s}_t$ perturbed by Bayesian optimization based Black-box attack rather than the state s_t in model testing.

TABLE II
Final Performance of Different Algorithms in Model Training.

	Return	Speed	Collision Times
DQN	92.99 ± 11.31	20.84 ± 0.78	2.20 ± 1.46
PPO	111.03 ± 16.03	**28.86 ± 1.75**	4.20 ± 0.75
SAC	120.39 ± 15.73	24.92 ± 0.65	1.40 ± 1.35
OARL	**121.36 ± 18.16**	26.98 ± 1.36	**1.20 ± 1.6**

B. Baseline

The DQN and PPO based autonomous driving lane change decision making algorithms are implemented as classical baseline methods. Additional, since soft actor-critic (SAC) with discrete action [40] is state-of-the-art discrete action RL algorithm, it is adopted as a state-of-the-art baseline scheme.

C. Evaluation

Fig. 5 demonstrates the performance of each algorithm during training in the highway scenario based on stochastic mixed traffic flow with normal density. The final performance of different schemes is given in Table II. Bold number is the best in each column of Table II. All the algorithms are evaluated for five trials via different random seeds in stochastic mixed traffic flow with normal density. The solid curve corresponds to the mean and the shaded region represents the standard deviation.

Fig. 5 and Table II shows that the robust lane change decision making method based on OARL outperforms the baseline schemes with a large margin, both in terms of the learning efficiency and the final performance. We count the average metrics over the final 2000 time steps (10 episodes × 200 time steps). Moreover, the average return of one episode is counted over the final 2000 time steps. It can be found that OARL approach performs comparably to SAC method and outperforms DQN and PPO schemes in term of the final speed in stochastic mixed traffic flow with normal density. For example, in contrast to DQN, PPO and SAC schemes, OARL gains 31.52%, 9.31% and 0.83% improvements with respect to the final return respectively. In addition, compared with DQN, PPO and SAC methods, the collision safety of

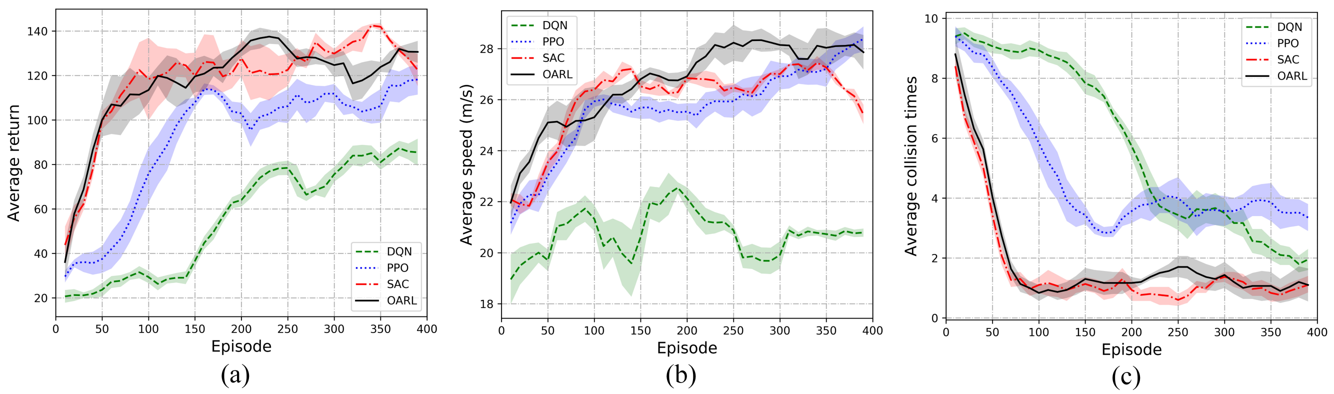


Fig. 5. Training curves obtained by DQN, PPO, SAC and OARL algorithms. (a): Average return; (b): Average speed; (c): Average collision times.

TABLE III
EVALUATION OF THE POLICIES TRAINED BY DIFFERENT ALGORITHMS IN THREE STOCHASTIC MIXED TRAFFIC FLOWS.

Environment	Metric	DQN	PPO	SAC	OARL
Low Density	Return	155.81 ± 22.97	145.50 ± 13.84	169.47 ± 30.91	$\mathbf{181.15 \pm 7.62}$
	Speed	28.99 ± 3.28	26.95 ± 2.00	31.48 ± 0.95	**32.10 ± 1.08**
	Robustness	$(0.54 \pm 0.12) \times 10^{-3}$	$(6.82 \pm 1.27) \times 10^{-3}$	$(4.64 \pm 3.36) \times 10^{-3}$	$(\mathbf{5.92 \pm 3.68}) \times \mathbf{10^{-5}}$
	Collision Times	0.25 ± 0.62	**0.00 ± 0.00**	**0.00 ± 0.00**	**0.00 ± 0.00**
Normal Density	Return	92.09 ± 27.67	113.63 ± 10.49	128.72 ± 19.76	$\mathbf{134.11 \pm 13.46}$
	Speed	24.95 ± 2.82	22.19 ± 1.49	**26.29 ± 1.77**	26.14 ± 1.10
	Robustness	$(0.53 \pm 0.11) \times 10^{-3}$	$(6.54 \pm 1.51) \times 10^{-3}$	$(1.20 \pm 1.00) \times 10^{-3}$	$(\mathbf{6.83 \pm 2.28}) \times \mathbf{10^{-5}}$
	Collision Times	2.50 ± 2.06	**0.30 ± 0.64**	0.70 ± 1.14	0.55 ± 0.86
High Density	Return	28.02 ± 16.11	86.84 ± 9.50	92.75 ± 26.64	$\mathbf{100.78 \pm 12.56}$
	Speed	18.08 ± 4.60	19.16 ± 0.99	**23.72 ± 1.62**	22.97 ± 1.23
	Robustness	$(7.86 \pm 2.68) \times 10^{-3}$	$(1.87 \pm 0.27) \times 10^{-2}$	$(4.04 \pm 2.18) \times 10^{-3}$	$(\mathbf{7.34 \pm 2.39}) \times \mathbf{10^{-5}}$
	Collision Times	6.05 ± 2.73	1.80 ± 1.21	1.80 ± 1.57	**1.40 ± 1.15**

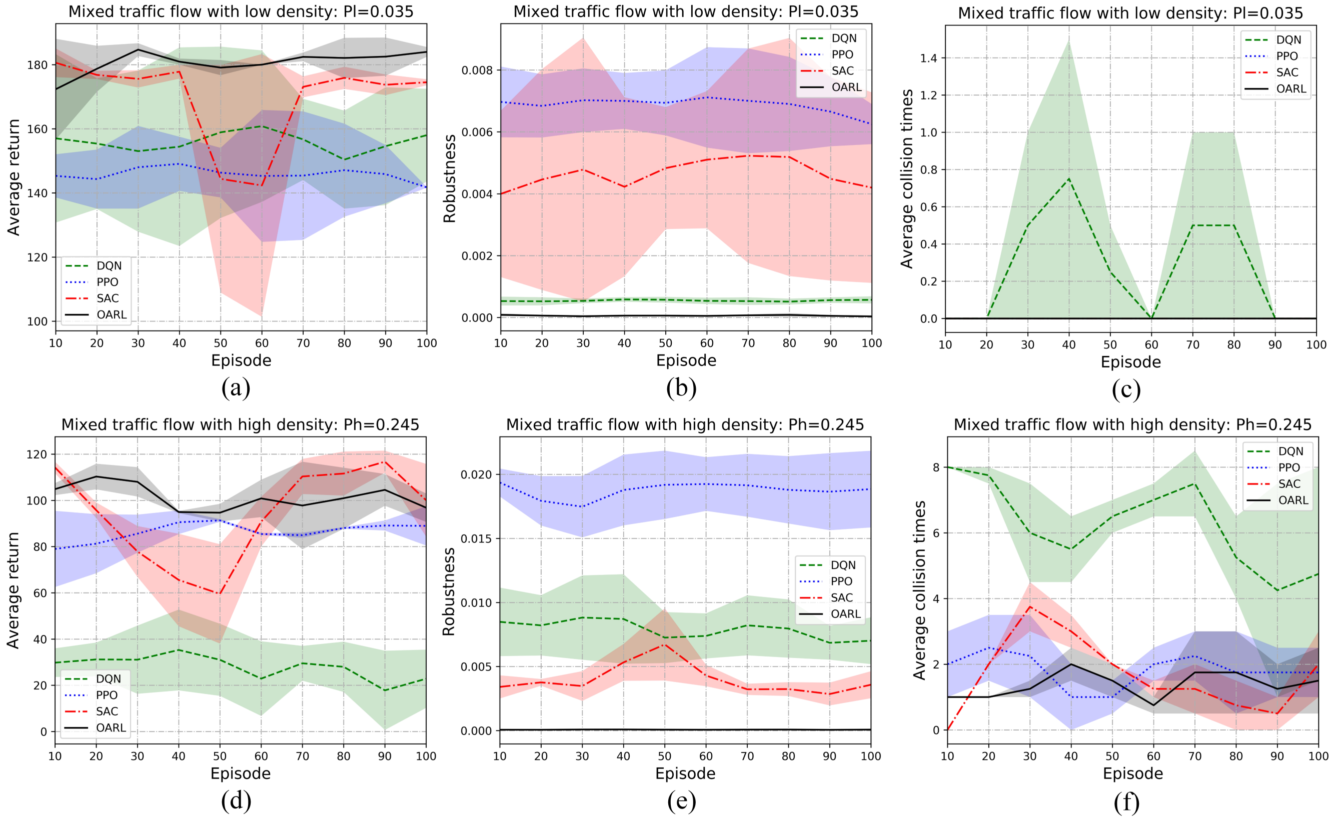
OARL is enhanced by about 83.33%, 250.00% and 16.67% respectively. It can be seen that, PPO is superior to OARL in terms of the final driving speed. However, the collision safety of PPO method is the worst.

Eq. 2 is utilized to measure the robustness of policy models against adversarial observation perturbations. We evaluate the final policy models trained by each methods with different random seeds. Additionally, the average metrics are counted over 40000 time steps (200 episodes × 200 time steps). Table III shows the test results of different policy models. The performance of OARL policies outperforms DQN, PPO and SAC in three stochastic mixed traffic flows with different densities, especially in terms of robustness metric. For instance, in contrast to DQN, PPO and SAC policies, OARL gains 16.25%, 24.83% and 7.10% improvements with respect to return in mixed traffic flow with low density respectively. Meanwhile, compared with DQN, PPO and SAC methods, the traffic efficiency of OARL policies is improved by about 10.73%, 19.11% and 1.97% respectively. It can be inferred that, to ensure the transport efficiency, the autonomous vehicle based on OARL policies performs more lane changes to overtake than one with the baseline scheme driving policies. Additionally, the robustness metric of OARL policies almost unchanged under adversarial observation perturbations.

In the stochastic mixed traffic flow scenario with normal density, the average return of OARL policies outperforms one of DQN, PPO and SAC policies. Hence, although each of PPO and SAC policies has a metric which is superior to one of OARL policies, OARL policies have better comprehensive performance than the baseline policies.

In the stochastic mixed traffic flow scenario with high density, OARL policies perform comparably to SAC policies and outperforms DQN and PPO polices in term of transport efficiency under adversarial observational perturbations. Moreover, in contrast to DQN, PPO and SAC policies, OARL gains 257.14%, 16.28% and 8.70% improvements with respect to return respectively. Compared with DQN, PPO and SAC policies, the collision safety of OARL policies is improved by about 332.14%, 28.57% and 28.57% respectively. It is obvious that the robustness of OARL policies against adversarial observation perturbations is superior to the one of DQN, PPO and SAC policies. Hence, it can be seen that the proposed method performs consistently in three different highway scenarios.

Furthermore, Fig. 6 visually shows the performance of DQN, PPO, SAC and OARL policies in the stochastic mixed traffic flows with low and high densities. it can be seen that OARL policies outperform baseline policies with a large margin, in term of return, robustness and collision safety. Moreover, the performance and robustness of OARL policies are scarcely influenced by adversarial observation perturbations. This means that the proposed robust lane change decision-making approach with OARL is able to improve



(a) (b) (c)

(d) (e) (f)

Fig. 6. Evaluation results for DQN, PPO, SAC and OARL policy models. (a): Average return; (b): Average robustness metric; (c): Average collision times.

the performance and generalization of autonomous driving RL agent while keeping the robustness of decision-making behaviors against observation uncertainties.

D. Ablation

In this section, we evaluate the impact of the nonlinear constraint on the performance of OARL agent. A scheme called actor-critic (AC) is implemented by removing the items associated with the constraint in OARL. AC and OARL methods are assessed in stochastic mixed traffic flow with normal density. Moreover, we train 5 different instances with different random seeds.

As shown in Fig. 7, the proposed OARL algorithm outperforms AC scheme with a large margin, in terms of average return. It can be found that AC algorithm fails to make any progress during policy model training. Hence, we can find two possible explanations for this phenomenon: (1) our constraint setting is able to encourage RL agent to explore and avoid falling into local optimum; (2) updating policy gradients in more directions may be beneficial to improve model performance.

Additionally, the performance of our OARL scheme with double hidden layer based network (DHLN) is evaluated in stochastic mixed traffic flow with normal density. It can be seen from Fig. 7 that OARL with a single hidden layer based neural network performs comparably to the OARL with DHLN, in terms of average return.

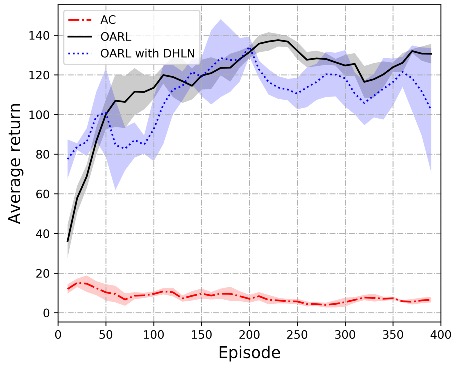


Fig. 7. Evaluation results of ablation and comparative study.

VI. Conclusion

This paper introduces a novel OARL approach for robust lane change decision making of autonomous vehicles. A COR-MDP is presented to model lane change decision making behaviors of autonomous vehicles under policy constraints and observation uncertainties. Meanwhile, the black-box attack technique with Bayesian optimization is implemented to find the optimal adversarial observation perturbations efficiently. Furthermore, a COR-AC algorithm is advanced to optimize autonomous driving lane change policies while keeping the variations of the policies attacked by the optimal adversarial observation perturbations within bounds.

The experiment results in three stochastic mixed traffic flows with different densities demonstrate that the proposed scheme can make lane change decisions robustly under observation uncertainties. In comparison with three baseline methods, the policy models trained by the proposed algorithm show superior generalization and robustness against adversarial observational perturbations.

Future work involves to evaluate the robust lane change decision making approach with OARL in more scenarios. Moreover, OARL with continuous action will be investigated to copy with longitudinal decision making problem of autonomous vehicles.

APPENDIX

TABLE IV
THE MAIN HYPERPARAMETERS OF THE PROPOSED ALGORITHM.

Parameters	**Value**	**Parameters**	**Value**
Decay factor λ	0.95	Adhesion coefficient $\bar{\mu}$	0.90
Dynamic factor k	0.85	Learning rate of actor l_a	0.0001
Learning rate of dual l_α	0.0005	Learning rate of critic l_c	0.001
Scale coefficient ρ	0.995	Constraint threshold ϵ	0.0001
Reference value Δ_m^0	1.00	Reference value Δ_a^0	0.00
Desired bound δ_m	0.2	Desired bound δ_a	0.05

REFERENCES

[1] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 187–210, 2018.

[2] S. Feng, X. Yan, H. Sun, Y. Feng, and H. X. Liu, "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment," *Nature communications*, vol. 12, no. 1, pp. 1–14, 2021.

[3] C. Pek, S. Manzinger, M. Koschi, and M. Althoff, "Using online verification to prevent autonomous vehicles from causing accidents," *Nature Machine Intelligence*, vol. 2, no. 9, pp. 518–528, 2020.

[4] C. Hubmann, J. Schulz, M. Becker, D. Althoff, and C. Stiller, "Automated driving in uncertain environments: Planning with interaction and uncertain maneuver prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 1, pp. 5–17, 2018.

[5] Z. Hu, C. Lv, P. Hang, C. Huang, and Y. Xing, "Data-driven estimation of driver attention using calibration-free eye gaze and scene features," *IEEE Transactions on Industrial Electronics*, 2021.

[6] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[7] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the game of go without human knowledge," *nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[8] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev *et al.*, "Grandmaster level in starcraft ii using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[9] 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*, 2021.

[10] P. R. Wurman, S. Barrett, K. Kawamoto, J. MacGlashan, K. Subramanian, T. J. Walsh, R. Capobianco, A. Devlic, F. Eckert, F. Fuchs *et al.*, "Outracing champion gran turismo drivers with deep reinforcement learning," *Nature*, vol. 602, no. 7896, pp. 223–228, 2022.

[11] S. Nageshrao, H. E. Tseng, and D. Filev, "Autonomous highway driving using deep reinforcement learning," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. IEEE, 2019, pp. 2326–2331.

[12] G. Li, Y. Yang, S. Li, X. Qu, N. Lyu, and S. E. Li, "Decision making of autonomous vehicles in lane change scenarios: Deep reinforcement learning approaches with risk awareness," *Transportation Research Part C: Emerging Technologies*, p. 103452, 2021.

[13] F. Ye, X. Cheng, P. Wang, C.-Y. Chan, and J. Zhang, "Automated lane change strategy using proximal policy optimization-based deep reinforcement learning," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 1746–1752.

[14] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*. ThinkMind, 2011.

[15] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2575–2582.

[16] Y. Fu, C. Li, F. R. Yu, T. H. Luan, and Y. Zhang, "A decision-making strategy for vehicle autonomous braking in emergency via deep reinforcement learning," *IEEE transactions on vehicular technology*, vol. 69, no. 6, pp. 5876–5888, 2020.

[17] H. Wang, H. Gao, S. Yuan, H. Zhao, K. Wang, X. Wang, K. Li, and D. Li, "Interpretable decision-making for autonomous vehicles at highway on-ramps with latent space reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 8707–8719, 2021.

[18] H. Shu, T. Liu, X. Mu, and D. Cao, "Driving tasks transfer using deep reinforcement learning for decision-making of autonomous vehicles in unsignalized intersection," *IEEE Transactions on Vehicular Technology*, 2021.

[19] B. Mirchevska, C. Pek, M. Werling, M. Althoff, and J. Boedecker, "High-level decision making for safe and reasonable autonomous lane changing using reinforcement learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2156–2162.

[20] G. Wang, J. Hu, Z. Li, and L. Li, "Harmonious lane changing via deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[21] J. Wang, Q. Zhang, D. Zhao, and Y. Chen, "Lane change decision-making through deep reinforcement learning with rule-based constraints," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–6.

[22] M. Huegle, G. Kalweit, B. Mirchevska, M. Werling, and J. Boedecker, "Dynamic input for deep reinforcement learning in autonomous driving," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 7566–7573.

[23] S. Jiang, J. Chen, and M. Shen, "An interactive lane change decision making model with deep reinforcement learning," in *2019 7th International Conference on Control, Mechatronics and Automation (ICCMA)*. IEEE, 2019, pp. 370–376.

[24] X. Xu, L. Zuo, X. Li, L. Qian, J. Ren, and Z. Sun, "A reinforcement learning approach to autonomous decision making of intelligent vehicles on highways," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 10, pp. 3884–3897, 2018.

[25] Y. Chen, C. Dong, P. Palanisamy, P. Mudalige, K. Muelling, and J. M. Dolan, "Attention-based hierarchical deep reinforcement learning for lane change behaviors in autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[26] C.-J. Hoel, K. Driggs-Campbell, K. Wolff, L. Laine, and M. J. Kochenderfer, "Combining planning and deep reinforcement learning in tactical decision making for autonomous driving," *IEEE transactions on intelligent vehicles*, vol. 5, no. 2, pp. 294–305, 2019.

[27] C.-J. Hoel, K. Wolff, and L. Laine, "Automated speed and lane change decision making using deep reinforcement learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2148–2155.

[28] Y. Zhang, B. Gao, L. Guo, H. Guo, and H. Chen, "Adaptive decision-making for automated vehicles under roundabout scenarios using optimization embedded reinforcement learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[29] H. An and J.-i. Jung, "Decision-making system for lane change using deep reinforcement learning in connected and automated driving," *Electronics*, vol. 8, no. 5, p. 543, 2019.

[30] J. Erdmann, "Sumo's lane-changing model," in *Modeling Mobility with Open Data*. Springer, 2015, pp. 105–123.

[31] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in neural information processing systems*, vol. 25, 2012.

[32] M. Pelikan, D. E. Goldberg, E. Cantú-Paz *et al.*, "Boa: The bayesian optimization algorithm," in *Proceedings of the genetic and evolutionary computation conference GECCO-99*, vol. 1. Citeseer, 1999, pp. 525–532.

[33] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2015.

[34] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[35] F. Huszár, "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" *arXiv preprint arXiv:1511.05101*, 2015.

[36] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: no regret and experimental design," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 2010, pp. 1015–1022.

[37] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[38] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[39] X. He, Y. Liu, C. Lv, X. Ji, and Y. Liu, "Emergency steering control of autonomous vehicle for collision avoidance and stabilisation," *Vehicle system dynamics*, vol. 57, no. 8, pp. 1163–1187, 2019.

[40] P. Christodoulou, "Soft actor-critic for discrete action settings," *arXiv preprint arXiv:1910.07207*, 2019.

Xiangkun He (Member, IEEE) received his PhD degree in 2019 from the School of Vehicle and Mobility, Tsinghua University, Beijing, China. During 2019–2021, he was a Senior Researcher at Noah's Ark Lab, Huawei Technologies, China. He is currently a Research Fellow at the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. He is the author or coauthor of more than 30 peer-reviewed publications. His research interests include reinforcement learning, multi-objective optimization, robust decision and control, autonomous driving and robotics. He was the recipient of the Tsinghua University Outstanding Doctoral Thesis Award in 2019, Best Paper Finalist Award at 2020 IEEE International Conference on Mechatronics and Automation, 1st Class Outstanding Paper Award of China Journal of Highway and Transport in 2021, Huawei Major Technological Breakthrough Award in 2021, Huawei 2012 Lab Star Award in 2021, Huawei Hisilicon Chip Star Award in 2021, and Best Paper Runner Up Award at 2022 6th CAA International Conference on Vehicular Control and Intelligence. He is also a Reviewer for more than 20 international journals or conferences, including IEEE T-VT, IEEE T-ITS, IEEE T-IV, IEEE T-II, IEEE/ASME T-MECH, IEEE RAL, VSD, JAS, MSSP, CoRL, et al.

Haohan Yang received the B.S. degree in mechanical engineering from the Dalian University of Technology, Dalian, China, in 2018, and the M.Sc. degree in vehicle engineering from the Shanghai Jiao Tong university. He is currently working toward the Ph.D. degree with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. His current research interests include human-machine cooperative driving, driver states estimation with machine learning methods, and intelligent/autonomous vehicles.

Zhongxu Hu (Member, IEEE) received a mechatronic Ph.D. degree from the Huazhong University of Science and Technology of China, in 2018. He was a senior engineer at Huawei. He is currently a research fellow within the Department of Mechanical and Aerospace Engineering of Nanyang Technological University in Singapore. His current research interests include human-machine collaboration, computer vision, and deep learning applied to driver behavior analysis and autonomous vehicles in multiple scenarios. Dr. Hu serves as a Lead Guest Editor for Computational Intelligence and Neuroscience, an Academic Editor/Editorial Board for Automotive Innovation, Journal of Electrical and Electronic Engineering, Advances in Multimedia, and is also an active reviewer for IEEE Transactions on Intelligent Transportation Systems, IEEE Transactions on Industrial Electronics, IEEE Intelligent Transportation Systems Magazine, Journal of Intelligent Manufacturing, and Journal of Advanced Transportation, etc.

Chen Lv (Senior Member, IEEE) is a Nanyang Assistant Professor at School of Mechanical and Aerospace Engineering, and the Cluster Director in Future Mobility Solutions, Nanyang Technological University, Singapore. He received his PhD degree at Department of Automotive Engineering, Tsinghua University, China in Jan 2016. He was a joint PhD researcher at UC Berkeley, USA during 2014-2015, and worked as a Research Fellow at Cranfield University, UK during 2016-2018. He joined NTU and founded the Automated Driving and Human-Machine System (AutoMan) Research Lab since June 2018. His research focuses on intelligent vehicles, automated driving, and human-machine systems, where he has contributed 2 books, over 100 papers, and obtained 12 granted patents. He serves as Associate Editor for IEEE T-ITS, IEEE TVT, and IEEE T-IV. He received many awards and honors, selectively including the Highly Commended Paper Award of IMechE UK in 2012, Japan NSK Outstanding Mechanical Engineering Paper Award in 2014, Tsinghua University Outstanding Doctoral Thesis Award in 2016, IEEE IV Best Workshop/Special Session Paper Award in 2018, Automotive Innovation Best Paper Award in 2020, the winner of Waymo Open Dataset Challenges at CVPR 2021, and Machines Young Investigator Award in 2022.