Safe Efficient Policy Optimization Algorithm for Unsignalized Intersection Navigation

Xiaolong Chen, Biao Xu, *Member, IEEE*, Manjiang Hu, Yougang Bian, *Member, IEEE*, Yang Li, and Xin Xu, *Senior Member, IEEE*

***Abstract*—Unsignalized intersections pose a challenge for autonomous vehicles that must decide how to navigate them safely and efficiently. This paper proposes a reinforcement learning (RL) method for autonomous vehicles to navigate unsignalized intersections safely and efficiently. The method uses a semantic scene representation to handle variable numbers of vehicles and a universal reward function to facilitate stable learning. A collision risk function is designed to penalize unsafe actions and guide the agent to avoid them. A scalable policy optimization algorithm is introduced to improve data efficiency and safety for vehicle learning at intersections. The algorithm employs experience replay to overcome the on-policy limitation of proximal policy optimization and incorporates the collision risk constraint into the policy optimization problem. The proposed safe RL algorithm can balance the trade-off between vehicle traffic safety and policy learning efficiency. Simulated intersection scenarios with different traffic situations are used to test the algorithm and demonstrate its high success rates and low collision rates under different traffic conditions. The algorithm shows the potential of RL for enhancing the safety and reliability of autonomous driving systems at unsignalized intersections.**

***Index Terms*—Autonomous driving, decision-making, reinforcement learning (RL), unsignalized intersection.**

I. Introduction

UNSIGNALIZED intersections represent a challenging road scenario where autonomous vehicles (AVs) face complex decision-making and coordination tasks due to the absence of regulatory signals or signs. AVs must navigate and communicate with other vehicles that exhibit varying levels of autonomy and communication capabilities, which complicates the control and decision-making processes [1], [2]. To tackle these challenges, researchers have proposed several methods, including rule-based approaches that follow predefined behavior patterns [3], [4], game-theoretic approaches that model strategic interactions among agents [5], [6], and reinforcement learning (RL) approaches that learn optimal policies from data or simulation [7], [8].

Vehicle behavior at unsignalized intersections can be guided by predefined rules or constraints, such as priority rules based on arrival order or lane position or collision avoidance rules based on distance or speed thresholds [4], [9]. These rule-based methods are simple and easy to implement, but may lack optimality or adaptability to dynamic traffic situations. Alternatively, the game-theoretic framework can be used to model the interaction among vehicles at unsignalized intersections, where each vehicle seeks to optimize its own payoff or utility function that reflects safety, efficiency, comfort, and social preferences [10], [11]. However, this framework may face challenges in finding a solution for complex or uncertain games at unsignalized intersections.

In particular, RL methods offer a promising approach to addressing the decision and control problems associated with unsignalized intersections, as they can learn optimal policies for vehicle actions (e.g., acceleration or braking) based on environmental states (e.g., traffic conditions) without the need for explicit rules or models of other vehicles' behavior [12]. RL aims to maximize a long-term reward function [13] that captures safety and efficiency objectives, providing advantages such as flexibility, scalability, generalization, and self-improvement compared to rule-based and game-theoretic methods [14], [15].

However, applying RL to urban autonomous driving poses several challenges: a suitable representation of the state space that captures the relevant information is required; and high-dimensional action spaces and partial observability have to be handled. Bird's eye view or image as input representation to capture the low-dimensional latent states at unsignalized intersections has been used by some researchers [16]−[18]. Such representation can facilitate complex driving behaviors learning but entail high computation resource. Moreover, these studies rely on deep Q-learning method and are limited to discrete actions [19].

Policy gradient or actor-critic methods have been utilized to tackle the challenge of high-dimensional continuous actions. In the context of intersection control, a partially observable Markov decision process (POMDP) based on deep determinis-

Manuscript received July 18, 2023; revised December 25, 2023; accepted January 30, 2024. This work was supported by the National Natural Science Foundation of China (52102394, 52172384), Hunan Provincial Natural Science Foundation of China (2023JJ10008), and Young Elite Scientists Sponsorship Program by CAST (2022QNRC001). Recommended by Associate Editor Tao Yang. *(Corresponding author: Biao Xu.)*

Citation: X. Chen, B. Xu, M. Hu, Y. Bian, Y. Li, and X. Xu, "Safe efficient policy optimization algorithm for unsignalized intersection navigation," *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 9, pp. 2011–2026, Sept. 2024.

X. Chen and Y. Li are with the State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, the College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China (e-mail: xlchan18@hnu.edu.cn; yangli05@hnu.edu.cn).

B. Xu, M. Hu, and Y. Bian are with the State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, the College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, and also with the Wuxi Intelligent Control Research Institute (WICRI) of Hunan University, Wuxi 214115, China (e-mail: xubiao@hnu.edu.cn; manjiang_h@hnu.edu.cn; byg19@hnu.edu.cn).

X. Xu is with the College of Intelligence Science and Technology, Institute of Unmanned Systems, National University of Defense Technology, Changsha 410073, China (e-mail: xinxu@nudt.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2024.124287

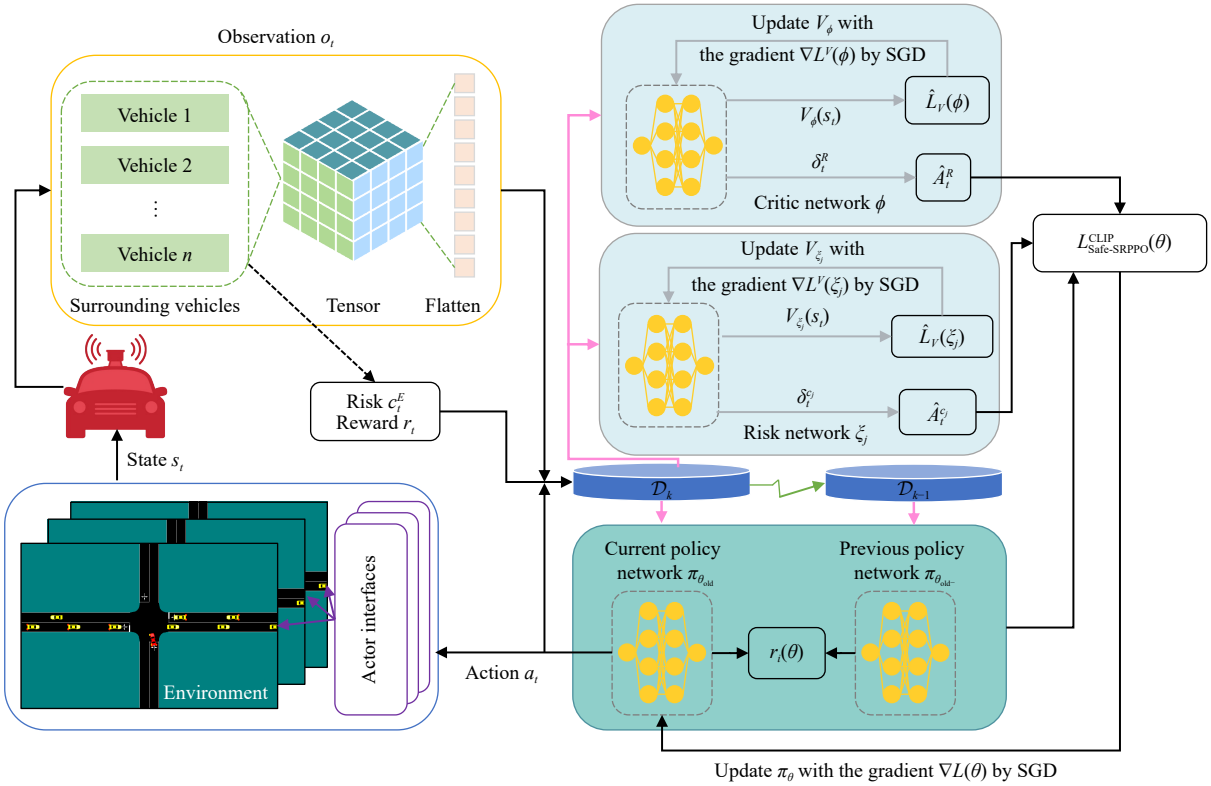


Fig. 1. The proposed DRL framework for unsignalized intersection scene.

tic policy gradient (DDPG) was developed [20]. However, this approach required significant computational and memory resources. Reference [21] proposed a minimax distributional soft actor-critic algorithm to improve the generalization ability of RL algorithms in uncertain intersection environments. Unfortunately, this approach struggled with a dynamic number of vehicles, which did not match the real road condition. Reference [22] designed a soft actor-critic framework with double Q-learning to handle multiple intersection scenarios, but the results indicated that the final converged strategies were not always safe for completing the tasks. Additionally, [23] evaluated the performance of twin delayed DDPG using visual multi-frame image as input. The results showed that the algorithm had difficulty in policy convergence and data efficiency.

The safety of vehicles at unsignalized intersections is a crucial issue that has received some attention in the existing literature. RL methods have been applied to this, but they face several challenges such as uncertainty and scalability. Reference [24] proposed a two-level decision algorithm that combines RL with model predictive control (MPC) to balance safety and efficiency at intersections. The algorithm used RL to learn a high-level policy that decided when to yield or proceed, and MPC to generate low-level trajectories that satisfied kinematic and comfort constraints. Reference [25] extended this approach by incorporating risk-awareness into the RL agent, using uncertainty estimation to guide action selection in novel situations. However, applying RL algorithms on safety-critical systems still requires careful justification due to the exploration nature of many RL algorithms, especially when the model of the robot and the environment are unknown. Reference [26] addressed this challenge by proposing a data-driven safety layer that filters out unsafe actions based on historical data. The security of the exploration strategy and its generalizability are not directly considered by these methods.

Unsignalized intersections pose several challenges for RL applications in traffic. First, most RL methods use discrete actions, while continuous actions are underexplored. Second, using images as state inputs makes it hard to capture vehicle correlations, leading to slow convergence and low data efficiency. Third, using state vector representations cannot handle variable vehicle counts, large-scale scenarios, or realistic environments. Fourth, most RL methods focus on learning efficiency and optimality, but neglect safety issues. Therefore, RL needs to address data efficiency, safety assurance, scalability, and coordination at unsignalized intersections.

This paper proposes a decision-making framework with policy optimization for safe and efficient passing of AV at unsignalized intersections, where this control problem is formulated as an RL problem. The framework is shown in the Fig. 1. A semantic scene graph is proposed to represent the intersection environment and handle scenarios with varying numbers of vehicles. The policy optimization algorithm is inspired by proximal policy optimization (PPO) [27], an on-policy RL algorithm that can handle both discrete and continuous actions and scale up to large-scale training. To improve the sampling efficiency of PPO's on-policy feature, we propose an algorithm that utilizes experience replay. Moreover, to address the problem of vehicle collisions at intersections and to draw inspiration from stability guarantees suggested in [28], we propose a risk assessment function to be applied within the deep RL (DRL) framework. The framework pro-

poses scalable RL algorithms with a focus on enhancing efficiency and ensuring safety. The main contributions of this paper are summarized as follows:

1) A better state representation method that can be applied to intersection scenarios is proposed, which can cope with dynamic vehicle numbers and learn potential states more easily for DRL.

2) A reward function for the intersection scene is proposed, comprising a main line reward and an auxiliary reward. The reward function is applicable to various scenes and serves as a criterion for evaluating different algorithms.

3) To assess the potential collision conflict, this study develops a two-vehicle conflict detection method based on time-to-intersection crossing and vehicle-to-vehicle (V2V) communication information for intersection navigation, which is integrated to the constrained Markov decision process.

4) A novel safety RL algorithm is proposed that aims to balance the trade-off between vehicle traffic safety and policy learning efficiency. The algorithm demonstrates its potential to improve autonomous driving systems in complex and dynamic environments.

The paper is structured as follows. Section II formulates the vehicles traveling problem. Section III designs the constrained Markov decision process. Section IV proposes and derives the RL algorithms. Section V verifies the algorithms at different tasks. Finally, Section VI draws the conclusions and outlines the future works.

II. Vehicles Traveling in Intersection Environment

A traffic environment is constructed where a self-driving vehicle and other vehicles interact at a four-way intersection without traffic signals. A behavior controller is developed for the surrounding vehicles, with their speeds and positions randomly assigned. The decision-making elements for the self-driving vehicle are also defined, including state variables, control actions, reward function and transition model.

A. Traffic Navigation at Intersection

When a vehicle approaches an unsignalized intersection, its primary decision is whether to proceed along its planned route or to stop before the intersection. This decision is contingent on the established right-of-way rules. In urban settings, unsignalized intersections are typically governed by priority-based control or right-hand priority [29]. In this study, we adopt the scenario from [19], focusing on priority-based controlled intersections. Here, we designate the east-west road as the main thoroughfare and the north-south road as the subordinate route. Each lane leading to the intersection is defined with a length of 100 meters. At these priority-based intersections, vehicles on the secondary (north-south) road are required to yield to traffic on the primary (east-west) road.

This paper explores the decision-making process of a self-driving vehicle (referred to as the "ego vehicle") as it approaches an intersection from a secondary road, encountering crossing traffic from the primary road. It begins with the ego vehicle at the stop line, ready to perform various tasks like turning left, going straight, or turning right. The vehicle must assess the situation and avoid collisions with other vehicles, whose intentions are uncertain. Given the complexity and dynamic nature of this scenario, we frame the vehicle navigation task as a problem in reinforcement learning, where the ego vehicle must learn to optimally balance safety considerations and driving objectives.

The intersection scenario and all vehicles are modeled using the simulation of urban mobility (SUMO) simulator [30]. Repeated vehicles named flows are inserted randomly from the east and west ends of the road network, as shown in Fig. 1, and move according to a modification of the Krauss car following driver model [31]. The learning policy controls the ego vehicle, while other vehicles can respond to its behavior, such as braking or accelerating. However, these responses cannot prevent collisions entirely, which depend on the ego vehicle's actions. Each vehicle's position is updated at each tick based on its current state. A collision removes the ego vehicle from the network. The ego vehicle must reach its destination within a given time limit. Otherwise, it is removed from the network along with the episode.

B. Partially Observable Markov Decision Process

As described above, we define the reinforcement learning issue for intersection decision making as a POMDP to model system dynamics with a hidden Markov model that probabilistically relates unobservable system states to observations. As a formal description of a discrete-time POMDP, it can be referred to as a 7-tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{R}, \Omega, \gamma \rangle$, where $\mathcal{S}$, $\mathcal{A}$, $\mathcal{O}$ are the state space, action space, observation space accordingly, and $\mathcal{T}$, $\mathcal{R}$, Ω are the state transition function, reward function and observation model, respectively, while γ presents the discount rate. The ego vehicle tries to select actions so that the sum of the discounted rewards it receives over the future is maximized, defined as

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k} \tag{1}$$

where R_{t+k} is the reward at time step $t+k$.

The POMDP generalizes a Markov decision process (MDP) by allowing for partial observations of the environment state. An ego vehicle in a POMDP chooses an action at each time step based on its current observation, receives a reward and transitions to a new state. The state transition probability $\mathcal{T}(s'|s,a)$ and the observation probability $\Omega(o|s,a,s')$ are often unknown in real-world problems, but can be approximated by learning methods. The reward function $R(s,a,s')$ defines the immediate reward for each state-action pair. The goal of the ego vehicle is to maximize its expected future discounted reward by following a policy $\pi : \mathcal{S} \mapsto \mathcal{P}(\mathcal{A})$ that maps states to action probabilities. Policies are searched in a parameterized class $\Pi_\theta \subset \Pi$ (for example, a neural network class with weight θ). A policy in this class is denoted as $\pi(\theta)$ to indicate its dependence on θ. The expected future discounted reward of an ego vehicle that interacts with the environment under policy $\pi(\theta)$ is

$$R(\pi) = \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})\right] \tag{2}$$

where $\tau=(s_0,a_0,s_1,a_1,\ldots)$ represents the trajectory, $\tau\sim\pi$ indicates that the distribution on the trajectory is determined by the policy π, namely $s_0\sim p_0$, $a_t\sim\pi(\cdot|s_t)$, $s_{t+1}\sim P(\cdot|s_t,a_t)$.

1) State Space and Observation Space: The state space of the environment is a collection of the dominant and implicit states of all participating vehicles. Dominant states consist of immediately observable information, and implicit states cannot be directly observed but continue to influence the ongoing states, mostly referring to the intent of other vehicles. The whole state space of the environment is described as follows:

$$s=\left(s_{\text{term}},s_{\text{coll}},s_0^p,s_1^p,\ldots,s_{n_{\text{veh}}}^p,s_0^m,s_1^m,\ldots,s_{n_{\text{veh}}}^m\right) \tag{3}$$

which consists of the environmental indicator states including collision detector s_{coll} and terminal state s_{term}. These two are Boolean states, which take the value 1 when a collision occurs, or a terminal state is reached, and otherwise 0. The other two terms are the physical state s_i^p and the driver state s_i^m (driver model parameters) of the ego vehicle with index 0 and the n_{veh} surrounding vehicles in a traffic scene. The physical state describes a vehicle driving on the road by its continuous position, velocity, and heading

$$s_i^p=(x_i,y_i,v_i,\psi_i). \tag{4}$$

The driver state s_i^m is described by the model parameters

$$s_i^m=\left(v_i^{\text{set}},\vartheta_i,a_i^{\max},b_i,b_i^{\text{e}},\kappa_i,\sigma_i,\tau_i\right) \tag{5}$$

where v_i^{set} is the departing speed, ϑ_i is the minimum gap distance when standing, $a_i^{\max}$ is the acceleration ability of vehicles of this type, b_i is the desired deceleration, b_i^{e} describes the maximum deceleration ability of vehicles of this type in case of emergency, κ_i denotes the additional delay time before starting to drive after having had to stop, σ_i defines the driver imperfection (0 denotes perfect driving), and τ_i is the driver's desired and minimum time headway.

The observation model Ω assumes no sensor noise and full state awareness for the ego vehicle. The future paths of the surrounding vehicles are unknown. The observation space is relative to the ego vehicle. When departing from the lane and approaching the intersection, it can receive the global localization, speed, and heading of the oncoming vehicle in each time frame. The observation o includes the physical states of the surrounding vehicles and the ego vehicle's physical and driver states. Geometric representations have semantics such as object classes and spatial relationships, which semantic models emphasize [32], [33]. Therefore, a semantic representation is used for the observation space in this paper. The ego vehicle departs from a stop line at a distance d_0 from the intersection. The number of surrounding vehicles is dynamic, making it hard to describe a state with varying dimensions. To solve this problem and use a neural network input, a semantic scene graph is constructed for the ego vehicle, which is a fixed-size relational grid with only relevant relations.

We create a relational grid with the ego vehicle as its center, as shown in Fig. 2. In the real intersection scene, the global pose of each vehicle is determined based on the global coordinate system. The ego vehicle can obtain the information of surrounding vehicles via the vehicle-to-vehicle (V2V) communication technology. Therefore, the received information is further processed into relative pose to the ego vehicle, which contains relative position, relative angle and relative velocity. To make it easier to train, we define the semantic scene graph as a three-order tensor with the discrete feature, i.e., $o\in\mathbb{R}^{M\times N\times K}$ as shown in Fig. 3. M and N define the numbers of discrete granularity of relative longitudinal distance and horizontal distance, respectively. K is the dimension of the vector q that considers relative important information and a boolean state determining the presence or absence of the surround vehicle. In the study, $K=3$ and $q=(\Delta\psi,\Delta v,\odot)$, where $\Delta\psi$ is the heading relative to the ego vehicle, Δv is the longitudinal velocity relative to the ego vehicle and $\odot$ is the boolean indicator state.

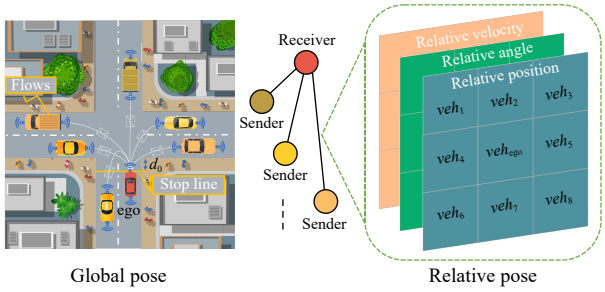


Fig. 2. Semantic scene graph.

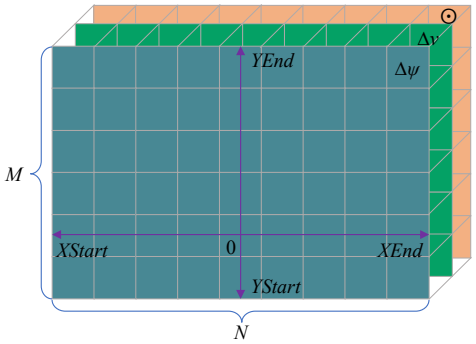


Fig. 3. Relative state information tensor.

2) Action Space: The ego vehicle's longitudinal and lateral movements are controlled by upper and lower commands, respectively. The upper command uses DRL to regulate the longitudinal motion, while the lower command implements a controller that maintains target speed and lane. The ego vehicle's route was predefined by an XML definition in SUMO. The steering angle is well controlled by the lower command within SUMO. The DRL only influences the longitudinal acceleration of the ego vehicle, which had a continuous action space $\mathcal{A}=[-u_{\text{decel}},u_{\text{accel}}]$.

To enhance the stability of DRL algorithms, action spaces are normalized, which adjusts the neural network outputs to a standard range, typically $[-1,1]$. This normalization aids in accelerating training and helps prevent divergence. For continuous-action DRL, outputs represent actions or parameters to construct actions, rather than having a node for each action. These normalized actions are then transformed back into actual control parameters through an inverse process applied

to the policy network's outputs

$$a_{\text{real}} = -u_{\text{decel}} + (a_{\text{out}} - (-1)) \times \frac{u_{\text{accel}} - (-u_{\text{decel}})}{1 - (-1)} \tag{6}$$

where a_{out} is the output action of the policy network and a_{real} is the real action to be taken by the ego vehicle.

3) Reward Function Tuning: In DRL, the agent refines its policy using rewards received from the environment. The neural network processes the state inputs, guided by the reward function, to estimate values for decision-making. The reward function's design, thus, is crucial. For intersection navigation, we focus on key objectives — safety, efficiency, and cooperation — split into mainline and auxiliary rewards.

i) The mainline reward is based on achieving qualitative goals, such as successfully navigating a two-dimensional task, winning a game of chess, or passing a level in a game. In our specific task, the mainline reward depends on three terminal states: arrival at the predefined destination (the primary goal), collision with another vehicle (defined conservatively as occupying its safe boundary), or running out of time before reaching the destination within the limited time frame τ_m. The mainline reward can be defined as follows:

$$r_m = \begin{cases} c_{\text{arr}} & \text{if arrival} \\ -c_{\text{col}} & \text{if collision} \\ -c_{\text{out}} & \text{if timeout, i.e., } \tau_e \geq \tau_m. \end{cases} \tag{7}$$

The ego vehicle may encounter more than one terminal state of an episode. Therefore, the mainline reward is calculated by: $r_m = c_{\text{arr}}\mathbf{1}[s \in S_{\text{arrival}}] - c_{\text{col}}\mathbf{1}[s \in S_{\text{collision}}] - c_{\text{out}}\mathbf{1}[s \in S_{\text{timeout}}]$, where $\mathbf{1}[\cdot]$ denotes a conditional judgment, which takes the value 1 if the condition in the parentheses holds, and 0 otherwise.

ii) Tasks that have high exploration difficulty and only offer a mainline reward often suffer from sparse reward issues. This lack of feedback signals can make learning difficult. To address this problem, auxiliary rewards or penalties can be added to the mainline reward to create a more robust reward function that guides the agent towards efficient exploration in the environment. Auxiliary rewards include efficiency and cooperation rewards. The former encourages the ego vehicle to reach its target quickly, usually by increasing speed, while the latter promotes collaboration with other agents in achieving shared goals. The penalty function for efficiency can be

$$r_e = -c_{\text{eff}} \frac{v_{\max} - v_0}{v_{\max}} \tag{8}$$

where c_{eff} is the absolute value of the penalty factor; $v_{\max}$ and v_0 are the maximum limit speed and current speed of the ego vehicle, respectively.

The cooperation reward, in our study, extends beyond the conventional focus on the autonomous agent's benefit, taking into account the broader impact of the ego vehicle's decisions on the surrounding traffic. This approach aligns with recent shifts in the field, as highlighted by key studies such as [34], which emphasize the significance of interactive behaviors in traffic dynamics. Accordingly, we have identified three behavior modes — waiting, braking, and emergency braking — that are typically exhibited by surrounding vehicles in response to the ego vehicle's actions. Therefore, a multi-condition judgement reward for cooperation can be described as

$$\begin{aligned} r_c = & -c_{\text{wab}}\mathbf{1}\left[s \in S_{\text{traffic\_waiting}} \cap S_{\text{traffic\_braking}}\right] \\ & - c_{\text{wob}}\mathbf{1}\left[s \in S_{\text{traffic\_waiting}} \cup S_{\text{traffic\_braking}}\right] \\ & + c_{\text{nwb}}\mathbf{1}\left[s \notin S_{\text{traffic\_waiting}} \cup S_{\text{traffic\_braking}}\right] \\ & - c_{\text{teb}}\mathbf{1}\left[s \in S_{\text{traffic\_emergencybraking}}\right] \end{aligned} \tag{9}$$

where c_* is the absolute value of the penalty factor or incentive factor. When the ego vehicle causes waiting, braking or emergency braking behavior to surrounding vehicles, we will give the appropriate penalty. Otherwise, it does not interfere with traffic movement and we will give encouragement as shown in the third term.

III. Constrained Markov Decision Process

One way to ensure the safety of the agent is to add constraints to the MDP framework and transform the problem into a constrained Markov decision process (CMDP). At this time, the goal of the agent is to maximize long-term rewards under the condition of meeting long-term risk constraints. This method can solve the above two problems at the same time. CMDP is an MDP with added constraints on long-term discounted risk. Specifically, the ordinary MDP is augmented with m risk functions $C_1, \ldots, C_m$, where each risk function $C_i : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the mapping from interaction data pairs to risk. According to (2), the long-term discount risk under the strategy π can be defined accordingly as $J_{C_i}(\pi) = \mathrm{E}_{\tau\sim\pi}[\sum_{t=0}^{\infty} \gamma^t C_i(s_t, a_t, s_{t+1})]$, corresponding to the restriction d_i. In the CMDP, the objective is to choose a strategy π that maximizes the long-term reward $J_R(\pi)$ while satisfying the constraint on the long-term risk $J_{C_i}(\pi) \leq d_i, \forall i \in [m]$, i.e.,

$$\begin{aligned} \pi^* &= \arg\max_{\pi \in \Pi_\theta} J_R(\pi) \\ &\text{s.t. } J_{C_i}(\pi) \leq d_i, \forall i \in [m]. \end{aligned} \tag{10}$$

The following describes the design method for the collision risk constraint for this environment based on the CMDP framework.

A. V2V Conflict Time Based on Motion Relationship

The navigation of intersections requires assessing potential conflicts with surrounding vehicles. This study aims to quantify the degree of conflict between vehicles using their position and operational status. A two-vehicle conflict detection method is developed to determine whether a conflict occurs. Previous studies have relied on time to collision (TTC) [35] as a conflict indicator, but TTC treats traffic flow as a scalar and cannot anticipate conflicts in advance. Therefore, this study proposes a time-to-intersection crossing (TIC) method based on the motion relationship of vehicles in intersection scenarios, using V2V communication information. The vehicle is modeled as a circle with center of mass located at (x, y) and diameter $2R$, as depicted in Fig. 4, for ease of computations. The corresponding vehicle information in the global coordinate system is then obtainable. To simplify TIC calculation, we derive the relationship of relative movement between the

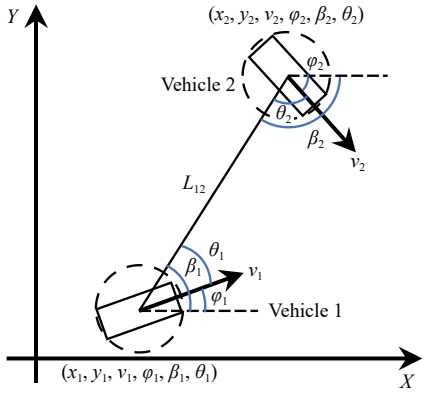


Fig. 4. Kinematic relationship between two vehicles.

two vehicles from Vehicle 1's coordinate view. This approach parallels the process of acquiring the observation space. The vehicle's velocity is represented by v, while φ denotes its heading angle. The relative position of Vehicle 2 with respect to Vehicle 1 is established as follows:

$$\begin{cases} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\varphi_1 & \sin\varphi_1 \\ -\sin\varphi_1 & \cos\varphi_1 \end{bmatrix} \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \end{bmatrix} \\ \varphi_r = \varphi_2 - \varphi_1 \\ v_x = v_2 \cos\varphi_r - v_1 \\ v_y = v_2 \sin\varphi_r \end{cases} \tag{11}$$

where the relative velocity of Vehicle 2 is v_r and its projection on L_{12} connecting the center points of the two vehicles is

$$\begin{cases} L_{12} = \sqrt{x^2 + y^2} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ v_L = v_x \cos\theta + v_y \sin\theta \\ \theta = \theta_2 - \theta_1 \end{cases} \tag{12}$$

where θ_1 and θ_2 are the angles between the speed directions of Vehicles 1 and 2 and the line connecting the two vehicles, respectively.

Next, we need to determine the calculation method of θ. First of all, it is specified that the direction of the angle is counterclockwise and positive. Secondly, define β as the angle between the global coordinate system and the connection between the two vehicles, specifically from the X axis to the connection. β_i is calculated as follows:

$$\beta_i = \text{atan2}\left(\frac{y_j - y_i}{x_j - x_i}\right) \tag{13}$$

where atan2(·) is four-quadrant arctangent function. Here if i = 1, then j = 2; if i = 2, then j = 1. We map the β value to the angle interal $[0, 360°]$. If $\beta_1 \geq 180°$, then $\beta_2 = \beta_1 - 180°$; if $\beta_1 < 180°$, then $\beta_2 = \beta_1 + 180°$.

As can be seen from Fig. 4,

$$\theta_i = \beta_i - \varphi_i \tag{14}$$

where φ_i is the information that vehicle i can obtain.

According to the previous analysis, the actual relative distance between the two vehicles is $\triangle L = L_{12} - 2R$. Therefore, the TIC between two vehicles can be obtained as

$$\text{TIC} = \frac{\triangle L}{v_L}. \tag{15}$$

There is no need to warn all vehicles during driving. Some vehicles do not have conflicts in traffic flow, so there is no need to calculate TIC. Next, consider and classify the potential collision relationship between vehicles according to the relative relationship between θ_1 and θ_2. Let $\theta > 0°$ indicate that the driving direction of the vehicle is on the right side of the central coordinate connection of the two vehicles; otherwise, if θ is negative, it means that the driving direction of the vehicle is on the left side of the central coordinate connection of the two vehicles. Set the range of θ is $[-180°, 180°)$, and when $\theta \geq 180°$, $\theta = \theta - 360°$; when $\theta < -180°$, $\theta = \theta + 360°$.

1) $|\theta_1 - \theta_2| = 0°$ indicates that the two vehicles are parallel and travelling in opposite directions, and there may be a frontal collision.

2) $|\theta_1 - \theta_2| = 180°$ means that the two vehicles are parallel and driving in the same direction, and there may be a rear-end collision.

3) If $0° < |\theta_1 - \theta_2| < 180°$ and θ_1 and θ_2 are with different signs, indicating that the two vehicles are on the same side and driving towards each other, and there may be a side collision.

4) For other cases, the vehicle is not dangerous.

This study scenario does not involve parallel and opposite-direction vehicles, so frontal collisions are excluded. The collision analysis considers Vehicle 2 moving from west to east and Vehicle 1 moving from south to north, as shown in Fig. 5. In the first scenario, Vehicle 2 has not crossed the intersection yet and θ_2 and θ_1 differ, indicating that both vehicles are on the same side and move in opposite directions, which may cause a side collision. In the second scenario, θ_1 is zero and reaches the boundary value for a potential collision with Vehicle 2. In the third scenario, both vehicles have θ values less than 0° with the same sign, indicating no conflict between their traffic flow directions and no collision risk.

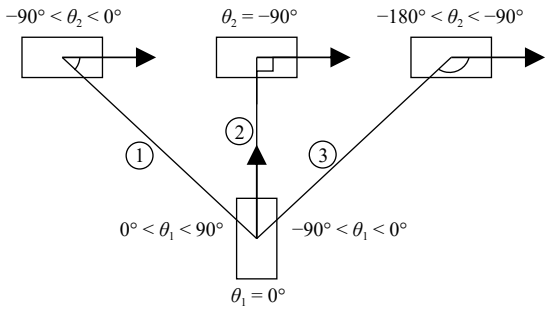


Fig. 5. Analysis of collision relationship at intersections.

B. Collision Risk Function

The collision risk function is defined based on the severity of potential collisions, and a quantitative index is designed to indicate the learning progress of the reinforcement learning algorithm. This index can be compared with TIC and time to avoidance (TTA) to evaluate the degree of collision risk. TTA is defined as

$$TTA = t_f + \frac{\delta v}{\mu g} \tag{16}$$

where t_f is the driver reaction time; δ is the deceleration factor, $\delta \in (0,1]$; μ is the friction coefficient of the vehicle tires on the road; and g is the acceleration of gravity.

Define the quantitative index of collision severity as

$$E = \frac{TIC}{TTA}. \tag{17}$$

When $0 < E \leq 0.5$, it is a serious conflict, and the collision risk is set to c_{E}; in other cases, it is set to 0.

IV. Reinforcement Learning Algorithms

The optimization objective of reinforcement learning based on static policy π, according to (2), combined with $V_\pi(s) = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty}\gamma^k R_{t+k} | S_t = s\right]$, can be expressed as

$$J(\pi) = \mathbb{E}_{s_0\sim\rho_0}\left[V_\pi(s_0)\right] \tag{18}$$

where s_0 is the initial state, ρ_0 is the initial state distribution, and $V_\pi(s_0)$ is the state value function of s_0 under policy π. In the learning process, after each policy update, the new policy π_{new} is improved compared with the old policy π_{old}

$$J(\pi_{\mathrm{new}}) = J(\pi_{\mathrm{old}}) + \mathbb{E}_{s_0,a_0,s_1,a_1,\ldots}\left[\sum_{t=0}^{\infty}\gamma^t A_{\pi_{\mathrm{old}}}(s_t,a_t)\right] \tag{19}$$

where $s_0 \sim \rho_0, a_t \sim \pi_{\mathrm{new}}(\cdot|s_t), s_{t+1} \sim P(s_{t+1}|s_t,a_t)$, with $P(s_{t+1}|s_t, a_t)$ being the environment transition probability, γ being the discount factor, $A_\pi(s_t,a_t)$ being the advantage function defined as $A_\pi(s_t,a_t) = \mathbb{E}_{s_{t+1}\sim P(s_{t+1}|s_i,a_t)}[r(s_t) + \gamma V_\pi(s_{t+1}) - V_\pi(s_t)]$.

Remark 1: To derive policy, we use the symbol (s_t,a_t). However, when dealing with POMDP, the ego vehicle samples are represented as (o_t,a_t) instead.

Equation (19) intuitively reveals that the amount of policy improvement is equal to the sum of expected discounted advantage functions over every state-action pair generated from π_{new}. The essential of reinforcement learning algorithms is: As long as π_{new} selects an action $A_{\pi_{\mathrm{old}}}(s_t,a_t) \geq 0$, π_{new} must be better than π_{old} (otherwise the policy has already converged to the optimal policy). However, in practical and interactive processes, the policy collected is that of π_{old} and π_{new} has a complicated coupling relationship with the policy distribution, making it difficult to optimize directly.

Parameterizing the policy as π_θ, the optimization problem can be approximated by solving the following optimization problem:

$$\begin{aligned}&\arg\max_\theta \mathbb{E}_{(s,a)\sim\tau_{\mathrm{old}}}\left[\frac{\pi_\theta(a|s)}{\pi_{\theta_{\mathrm{old}}}(a|s)}A_{\pi_{\theta_{\mathrm{old}}}}(s,a)\right]\\ &\text{s.t. } \mathbb{E}_{(s,a)\sim\tau_{\mathrm{old}}}\left[\left|\frac{\pi_\theta(a|s)}{\pi_{\theta_{\mathrm{old}}}(a|s)} - 1\right|\right] \leq \varepsilon.\end{aligned} \tag{20}$$

The final question above is transformed into a truncated target formula, which considers the constraints in the objective function, expands the absolute value, rearranges the items to become an unconstrained optimization problem

$$L^{\mathrm{CLIP}}(\theta) = \mathbb{E}_{(s,a)\sim\tau_{\mathrm{old}}}[\min(r(\theta)A, \mathrm{clip}(r(\theta), 1-\varepsilon, 1+\varepsilon)A)] \tag{21}$$

where, $r(\theta) = \pi_\theta(a|s)/\pi_{\theta_{\mathrm{old}}}(a|s)$, $A = A_{\pi_{\theta_{\mathrm{old}}}}(s,a)$, $\mathrm{clip}(x,l,h) = \max(\min(x,h),l)$, i.e., confine x within $[l,h]$. ε is a hyperparameter representing the range of clipping.

A. Sample Reuse Proximal Policy Optimization

Vanilla PPO is infeasible for experience replay in continuous action environment. The reason is that, in order to calculate the loss value, PPO uses the probability ratio $r(\theta)$, and assumes that the action is taken under $\pi_{\theta_{\mathrm{old}}}$. If this is not the case, and the parameter of the action sampling distribution has automatically changed after being recorded, then $\pi_{\theta_{\mathrm{old}}}(a_t|s_t)$ will quickly approach zero making $p_t(\theta)$ approach infinity. If the advantage function related to a_t is negative, the ratio will not be clipped and the loss value will be extremely large, thus resulting in gradient explosion. So here is to solve this shortcoming. We first give the total variance distance relationship between the visiting distributions and the total variance distance relationship between the policies.

Lemma 1: Consider any future new policy π and the old policy π_{old}, the total variational distance between the discounted future state visitation distributions $d_{\pi_{\mathrm{old}}}$ and d_π satisfies the following inequality:

$$D_{\mathrm{TV}}(d^\pi, d^{\pi_{\mathrm{old}}}) \leq \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_{\mathrm{old}}}}\left[D_{\mathrm{TV}}(\pi,\pi_{\mathrm{old}})(s)\right] \tag{22}$$

where $d^{\pi_{\mathrm{new}}} = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \Pr(s_t = s|\pi_{\mathrm{new}})$ (Pr as the state probability) is the normalized discounted visitation frequency, $D_{\mathrm{TV}}(d^\pi, d^{\pi_{\mathrm{old}}})$ is the total variational distance between the two discounted future state visitation distributions and $D_{\mathrm{TV}}(\pi, \pi_{\mathrm{old}})(s) = (1/2)\sum_a |\pi(a|s) - \pi_{\mathrm{old}}(a|s)|$ is the total variational distance between the two policies.

Proof: According to the above definition of d^π, it can be further transformed into $d^\pi = (1-\gamma)(I-\gamma P_\pi)^{-1}\rho_0$, where P_π is the transition matrix. Thus $d^\pi - d^{\pi_{\mathrm{old}}} = (1-\gamma)[(I-\gamma P_\pi)^{-1} - (I-\gamma P_{\pi_{\mathrm{old}}})^{-1}]\rho_0 = \gamma(I-\gamma P_\pi)^{-1}(P_\pi - P_{\pi_{\mathrm{old}}})d^{\pi_{\mathrm{old}}}$.

According to the Cauchy-Schwarz inequality, we obtain

$$\begin{aligned}D_{\mathrm{TV}}(d^\pi, d^{\pi_{\mathrm{old}}}) &= \frac{1}{2}\left\|d^\pi - d^{\pi_{\mathrm{old}}}\right\|_1\\ &= \frac{1}{2}\gamma\left\|(I-\gamma P_\pi)^{-1}(P_\pi - P_{\pi_{\mathrm{old}}})d^{\pi_{\mathrm{old}}}\right\|_1\\ &\leq \frac{1}{2}\gamma\|(I-\gamma P_\pi)^{-1}\|_1\left\|(P_\pi - P_{\pi_{\mathrm{old}}})d^{\pi_{\mathrm{old}}}\right\|_1.\end{aligned}$$

These two bounds are

$$\left\|(I-\gamma P_\pi)^{-1}\right\|_1 \leq \sum_{t=0}^{\infty}\gamma^t\|P_\pi\|_1^t = (1-\gamma)^{-1}$$

$$\begin{aligned}\left\|(P_\pi - P_{\pi_{\mathrm{old}}})d^{\pi_{\mathrm{old}}}\right\|_1 &= \sum_{s'}\left|\sum_s (P_\pi - P_{\pi_{\mathrm{old}}})(s'|s)\,d^{\pi_{\mathrm{old}}}(s)\right|\\ &\leq \sum_{s,a,s'} P(s'|s,a)|\pi(a|s) - \pi_{\mathrm{old}}(a|s)|\,d^{\pi_{\mathrm{old}}}(s)\\ &= \sum_{s,a}|\pi(a|s) - \pi_{\mathrm{old}}(a|s)|\,d^{\pi_{\mathrm{old}}}(s)\\ &= 2\mathop{\mathbb{E}}_{s\sim d^{\pi_{\mathrm{old}}}}\left[D_{\mathrm{TV}}(\pi,\pi_{\mathrm{old}})(s)\right].\end{aligned}$$

■

Remark 2: In Lemma 1, we present a bound on the total variation distance between the state visitation distributions of

any new policy π and the old policy π_{old}. This metric quantifies the dissimilarity in behavior between two policies over the long term, considering the discount factor γ. The lemma states that the discrepancy in visitation distributions, as measured by the total variation distance D_{TV}, is bounded by the expected total variation distance between the policies at each state, scaled by the factor $\gamma/(1-\gamma)$. This lemma is pivotal as it establishes a relationship that allows us to measure the impact of policy updates in terms of their influence on the distribution over future states visited by the agent.

Lemma 2: The following relation between the total variation distance between different policy distributions and the ratio of policies is satisfied:

$$\mathbb{E}_{s\sim d^{\pi_{\text{old}}}}\left[D_{\text{TV}}\left(\pi_{\text{new}},\pi_{\text{old}}\right)(s)\right]=\frac{1}{2}\mathbb{E}_{\substack{s\sim d^{\pi_{\text{old}}}\\ a\sim\pi_{\text{old}}}}\left[\left|\frac{\pi_{\text{new}}(a|s)}{\pi_{\text{old}}(a|s)}-1\right|\right]. \tag{23}$$

Proof: The proof is obtained according to the definition of D_{TV} and the importance sampling. ■

We next propose the sample reuse proximal policy optimization (SRPPO), an off-policy class of experience reuse algorithm, which considers the samples collected by the previous policy π_{old^-} for the current policy:

Theorem 1: Consider the current policy π_{old}, the previous policy π_{old^-} and any future new policy π_{new}, the new policy π_{new} has the following relation to the current policy π_{old} in terms of cumulative reward:

$$\begin{aligned}J(\pi_{\text{new}})-J(\pi_{\text{old}})\geq&\frac{1}{1-\gamma}\sum_i\kappa_i\mathbb{E}_{\substack{s\sim d^{\pi_i}\\ a\sim\pi_i}}\left[\frac{\pi_{\text{new}}(a|s)}{\pi_i(a|s)}A_{\pi_{\text{old}}}(s,a)\right]\\&-\frac{\gamma\epsilon}{(1-\gamma)^2}\sum_i\kappa_i\mathbb{E}_{\substack{s\sim d^{\pi_i}\\ a\sim\pi_i}}\left[\left|\frac{\pi_{\text{new}}(a|s)}{\pi_i(a|s)}-*\right|+|*-1|\right]\end{aligned} \tag{24}$$

where $*=\pi_{\text{old}}(a|s)/\pi_i(a|s)$, κ_i is the weight for policy π_i with the sum of the values equal to 1; when $i=0$, the policy refers to π_{old}; when $i=1$, the policy refers to π_{old^-}.

Proof: From (19), we can get

$$\begin{aligned}J(\pi_{\text{new}})-J(\pi_{\text{old}})&=\mathbb{E}_{s_0,a_0,s_1,a_1,\ldots}\left[\sum_{t=0}^{\infty}\gamma^tA_{\pi_{\text{old}}}(s_t,a_t)\right]\\&=\sum_{t=0}^{\infty}\gamma^t\sum_{s_t}\Pr(s_t|\pi_{\text{new}})\sum_{a_t}\pi_{\text{new}}(a_t|s_t)A_{\pi_{\text{old}}}(s_t,a_t)\\&=\sum_s\sum_{t=0}^{\infty}\gamma^t\Pr(\cdot)\sum_a\pi_{\text{new}}(a|s)A_{\pi_{\text{old}}}(s,a).\end{aligned} \tag{25}$$

Because $\sum_s\sum_{t=0}^{\infty}\gamma^t\Pr(\cdot)=\sum_{t=0}^{\infty}\gamma^t\sum_{s_t}\Pr(\cdot)=\sum_{t=0}^{\infty}\gamma^t=1/(1-\gamma)\neq 1$ ($\Pr(\cdot)$ refers to $\Pr(s_t=s|\pi_{\text{new}})$), for (25), to write it in the form of expectation, the distribution of s needs to be normalized

$$\begin{aligned}J(\pi_{\text{new}})-J(\pi_{\text{old}})=&\frac{1}{1-\gamma}\sum_s(1-\gamma)\sum_{t=0}^{\infty}\gamma^t\Pr(\cdot)\\&\times\sum_a\pi_{\text{new}}(a|s)A_{\pi_{\text{old}}}(s,a)=\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\pi_{\text{new}}}\\ a\sim\pi_{\text{new}}}}\left[A_{\pi_{\text{old}}}(s,a)\right].\end{aligned} \tag{26}$$

When considering the previous policy π_{old^-}, replacing the state action distribution from π_{new} to π_{old^-} yields the following result:

Let $\delta_f^{\pi_{\text{new}}}\in\mathbb{R}^{|S|}$ denote a vector with components $\delta_f^{\pi_{\text{new}}}(s)=\mathbb{E}_{a\sim\pi_{\text{new}}}[A_{\pi_{\text{old}}}(s,a)|s]$. Note that $\mathbb{E}_{\substack{s\sim d^{\pi_{\text{new}}}\\ a\sim\pi_{\text{new}}}}[A_{\pi_{\text{old}}}(s,a)]=\langle d^{\pi_{\text{new}}},\delta_f^{\pi_{\text{new}}}\rangle=\langle d^{\pi_{\text{old}^-}},\delta_f^{\pi_{\text{new}}}\rangle+\langle d^{\pi_{\text{new}}}-d^{\pi_{\text{old}^-}},\delta_f^{\pi_{\text{new}}}\rangle$. Then the Hölder inequality of the discrete form is applied to constrain it, and we have $|\langle d^{\pi_{\text{new}}}-d^{\pi_{\text{old}^-}},\delta_f^{\pi_{\text{new}}}\rangle|\leq\|d^{\pi_{\text{new}}}-d^{\pi_{\text{old}^-}}\|_p\|\delta_f^{\pi_{\text{new}}}\|_q$, where $p,q\geq 1$ and $1/p+1/q=1$. Therefore, $\mathbb{E}_{\substack{s\sim d^{\pi_{\text{new}}}\\ a\sim\pi_{\text{new}}}}[A_{\pi_{\text{old}}}(s,a)]\geq\langle d^{\pi_{\text{old}^-}},\delta_f^{\pi_{\text{new}}}\rangle-\|d^{\pi_{\text{new}}}-d^{\pi_{\text{old}^-}}\|_p\|\delta_f^{\pi_{\text{new}}}\|_q$.

According to the definition of $\delta_f^{\pi_{\text{new}}}$, the first term on the right side of the above inequality is $\langle d^{\pi_{\text{old}^-}},\delta_f^{\pi_{\text{new}}}\rangle=\mathbb{E}_{\substack{s\sim d^{\pi_{\text{old}^-}}\\ a\sim\pi_{\text{new}}}}[A_{\pi_{\text{old}}}(s,a)]$. Let $p=1$, $q=\infty$; (26) can be further expressed as

$$\begin{aligned}J(\pi_{\text{new}})-J(\pi_{\text{old}})\geq&\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\pi_{\text{old}^-}}\\ a\sim\pi_{\text{new}}}}\left[A_{\pi_{\text{old}}}(s,a)\right]\\&-\frac{1}{1-\gamma}\left\|d^{\pi_{\text{new}}}-d^{\pi_{\text{old}^-}}\right\|_1\left\|\mathbb{E}_{a\sim\pi_{\text{new}}}\left[A_{\pi_{\text{old}}}(s,a)\right]\right\|_\infty.\end{aligned} \tag{27}$$

Let $\|\mathbb{E}_{a\sim\pi_{\text{new}}}[A_{\pi_{\text{old}}}(s,a)]\|_\infty=\max_{s\in S}|\mathbb{E}_{a\sim\pi_{\text{new}}}[A_{\pi_{\text{old}}}(s,a)]|=\epsilon$. From Lemma 1, the following inequality can be obtained:

$$\begin{aligned}\|d^{\pi_{\text{new}}}-d^{\pi_{\text{old}^-}}\|_1&=2D_{\text{TV}}(d^{\pi_{\text{new}}},d^{\pi_{\text{old}^-}})\\&\leq\frac{2\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_{\text{old}^-}}}[D_{\text{TV}}(\pi_{\text{new}},\pi_{\text{old}^-})(s)].\end{aligned}$$

Then according to Lemma 2 and the inequality relation, we have

$$\begin{aligned}&2\mathbb{E}_{s\sim d^{\pi_{\text{old}^-}}}\left[D_{\text{TV}}\left(\pi_{\text{new}},\pi_{\text{old}^-}\right)(s)\right]\\&\quad\leq\mathbb{E}_{s\sim d^{\pi_{\text{old}^-}}}\left[\sum_a|\pi_{\text{new}}(a|s)-\pi_{\text{old}}(a|s)|\right]\\&\qquad+\mathbb{E}_{s\sim d^{\pi_{\text{old}^-}}}\left[\sum_a\left|\pi_{\text{old}}(a|s)-\pi_{\text{old}^-}(a|s)\right|\right]\\&\quad=\mathbb{E}_{\substack{s\sim d^{\pi_{\text{old}^-}}\\ a\sim\pi_{\text{old}^-}}}\left[\left|\frac{\pi_{\text{new}}(a|s)}{\pi_{\text{old}^-}(a|s)}-\frac{\pi_{\text{old}}(a|s)}{\pi_{\text{old}^-}(a|s)}\right|+\left|\frac{\pi_{\text{old}}(a|s)}{\pi_{\text{old}^-}(a|s)}-1\right|\right].\end{aligned}$$

The proof is obtained by considering the current policy π_{old} and then deriving similar results in combination. ■

Based on the aforementioned lower bound, the following optimization objective can be obtained:

Theorem 2: Denote the current policy as π_{old} and the previous policy as π_{old^-}, both parameterized by π_θ. The unconstrained optimization problem, considering the evolution of these policies, can be formulated as follows:

$$\begin{aligned}L_{\text{SRPPO}}^{\text{CLIP}}(\theta)=&\sum_i\kappa_i\mathbb{E}_{(s,a)\sim\tau_i}[\min(r_i(\theta)A,\text{clip}(r_i(\theta),\\&\varsigma_i-\varepsilon',\varsigma_i+\varepsilon')A)]\end{aligned} \tag{28}$$

where $r_i(\theta)=\pi_\theta(a|s)/\pi_{\theta_i}(a|s)$, $A=A_{\pi_{\theta_{\text{old}}}}(s,a)$, $\varsigma_i=\pi_{\theta_{\text{old}}}(a|s)/\pi_{\theta_i}(a|s)$.

Proof: Because $|\pi_{\text{old}}(a|s)/\pi_{\text{old}^-}(a|s)-1|$ is a constant for the data generated by π_{old^-}, it can be ignored in the optimization problem. According to the form in (20) and (21), with ς_i being the center, limiting the policy ratio in the interval $(\varsigma_i-\varepsilon',\varsigma_i+\varepsilon')$ guarantees that the difference between the new and old

policies will not be too large. This is verified by combining (21). ■

Theorem 2 formulates an optimization problem to evolve the policy parameters θ, using samples collected from previous policies. By comparing the current policy, π_{old}, with the previous one, π_{old^-}, it ensures informed updates that incorporate historical data, enabling a policy that learns from past experiences to enhance future performance.

B. Safe Proximal Policy Optimization

Constrained policy optimization (CPO) [36] finds feasible policies within the trust region and ensures monotonic improvement by solving quadratic optimization problems with appropriate approximations, while satisfying the constraints. The policy update form is derived from trust region policy optimization (TRPO) [37] for neural network policies, but it cannot be solved directly due to the high-dimensional parameter space with high computation cost. Therefore, Fisher information matrix and conjugate gradient are used to solve a linear objective optimization problem with linear and quadratic constraints. However, this approach still faces challenges such as approximation errors, complex high-dimensional Hessian matrix inversion and implementation difficulties. To address these issues, safe proximal policy optimization (Safe-PPO) is proposed. Π denotes the set of all static policies, Π_θ denotes feasible static policies in MDP and Π_C denotes feasible static policies in CMDP. In the local policy search in CMDP, the policy iteration must be feasible under CMDP, so the optimization should be performed on $\Pi_\theta \cap \Pi_C$.

Theorem 3: For parameterized strategies $\pi_{\theta_{\text{old}}}$ and π_θ in CMDP, the optimization seeks parameters θ that maximize expected returns within given constraints, ensuring policy improvements and adherence to predefined safety margins

$$\begin{aligned}&\arg\max_\theta \mathbb{E}_{(s,a)\sim\tau_{\text{old}}}\left[\frac{\pi_\theta(a|s)}{\pi_{\theta_{\text{old}}}(a|s)}A_R^{\pi_{\theta_{\text{old}}}}(s,a)\right]\\ &\text{s.t. } J_{C_i}(\pi_{\theta_{\text{old}}})+\frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim\tau_{\text{old}}}\left[\frac{\pi_\theta(a|s)}{\pi_{\theta_{\text{old}}}(a|s)}A_{C_i}^{\pi_{\theta_{\text{old}}}}(s,a)\right]\le d_i\\ &\quad \mathbb{E}_{s\sim d^{\pi_{\theta_{\text{old}}}}}\left[D_{\text{TV}}\left(\pi_\theta,\pi_{\theta_{\text{old}}}\right)(s)\right]\le\frac{\varepsilon}{2}.\end{aligned} \tag{29}$$

Proof: In refining our methodology for solving Problem (10), we draw upon Theorem 1 to define the objective function and the constraints of the policy optimization within a CMDP framework. In addressing the CMDP-specific trust region constraints, we consider the safety requirements of autonomous driving, encapsulated in $J_{C_i}(\pi)\le d_i, \forall i$. Here, the reward function traditionally used in trust region formulations is replaced with a risk function to align with these safety thresholds. This substitution reflects a deliberate shift in the optimization objective, balancing the pursuit of rewards with the imperative of risk aversion. The resulting risk function integrates safety considerations directly into the optimization process, guiding the development of policies that not only aim for high performance but also conform to essential safety standards. In (19), by substituting the reward function with the risk function and considering d_i, we have

$$J_{C_i}(\pi_{\text{new}})=J_{C_i}(\pi_{\text{old}})+\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\pi_{\text{new}}}\\ a\sim\pi_{\text{new}}}}\left[A_{C_i}^{\pi_{\text{old}}}(s,a)\right]\le d_i,\ \forall i.$$

Inspired by the deduction of Theorem 1, we obtain the upper bound

$$\begin{aligned}&J_{C_i}(\pi_{\text{new}})-\frac{\gamma\epsilon}{(1-\gamma)^2}\mathbb{E}_{\substack{s\sim d^{\pi_{\text{old}}}\\ a\sim\pi_{\text{old}}}}\left[\left|\frac{\pi_{\text{new}}(a|s)}{\pi_{\text{old}}(a|s)}-1\right|\right]\\ &\qquad\le J_{C_i}(\pi_{\text{old}})+\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\pi_{\text{old}}}\\ a\sim\pi_{\text{new}}}}\left[A_{C_i}^{\pi_{\text{old}}}(s,a)\right],\ \forall i.\end{aligned} \tag{30}$$

So as long as (30) satisfies less than or equal to d_i, and the residual constraint has been satisfied in the second constraint of (29), the constraints can be met. By applying importance sampling to (30), converting the sampling distribution of π_{new} to that of π_{old} and parameterizing the policy, we can obtain the form of (29). ■

Remark 3: Theorem 3 provides a structured approach to policy optimization in situations where safety and other operational constraints are critical. The expected advantage guides the optimization, pushing for policies that yield better performance compared to the past. Constraints are in place to prevent drastic deviations in behavior, ensuring that the new policy remains reliable and does not introduce excessive risk. This balance is crucial for the practical application of CMDP in autonomous navigation, where safety cannot be compromised.

Unlike previous work that approximates the non-convex problem within the trust region to the convex optimization problem by the Taylor expansion formula [36], [38], here, by combining the idea of proximal policy, the above constraint problem is converted to an unconstrained optimization problem, and the policy is still parameterized as π_θ

$$L_{\text{Safe-PPO}}^{\text{CLIP}}(\theta)=L^{\text{CLIP}}(\theta)-\varrho\sum_i^m\Phi(\hat{J}_{C_i}(\pi_\theta)) \tag{31}$$

where $L^{\text{CLIP}}(\theta)=\mathbb{E}_{(s,a)\sim\tau_{\text{old}}}[\min(r(\theta)A_R^{\pi_{\theta_{\text{old}}}}(s,a),\text{clip}(r(\theta),1-\varepsilon,1+\varepsilon)A_R^{\pi_{\theta_{\text{old}}}}(s,a))]$, ϱ is a penalty factor, the surrogate of the constraint condition $\hat{J}_{C_i}(\pi_\theta)=J_{C_i}(\pi_\theta)-d_i$ and the indicator function $\Phi(\cdot)$ is expressed as follows:

$$\Phi(\hat{J}_{C_i}(\pi_\theta))=\begin{cases}0, & \text{if } \hat{J}_{C_i}(\pi_\theta)\le 0\\ L_{C_i}^{\text{CLIP}}(\theta), & \text{if } \hat{J}_{C_i}(\pi_\theta)>0\end{cases}$$

where $L_{C_i}^{\text{CLIP}}(\theta)=\mathbb{E}_{(s,a)\sim\tau_{\text{old}}}[\max(r(\theta)A_{C_i}^{\pi_{\theta_{\text{old}}}}(s,a),\text{clip}(r(\theta),1-\varepsilon,1+\varepsilon)A_{C_i}^{\pi_{\theta_{\text{old}}}}(s,a))+(1-\gamma)(J_{C_i}(\pi_{\theta_{\text{old}}})-d_i)]$.

Multiplying both sides by $(1-\gamma)$ according to the constraint in (29), then moving terms leads to a relationship with 0. Therefore, as an indicator function, the penalty term is only considered when the constraint is not satisfied. This updating method simplifies the calculation, since the second item in $L_{C_i}^{\text{CLIP}}(\theta)$ is only used to determine whether the constraint needs to be considered, but in the actual updating process, it is only related to the first item.

C. Safe Sample Reuse Proximal Policy Optimization

Building upon the idea of reusing samples collected by the previous policy π_{old^-} in SRPPO, we propose a new off-policy

algorithm called safe sample reuse proximal policy optimization (Safe-SRPPO) that aims to reuse samples safely in an off-policy manner.

Corollary 1: Consider the current policy π_{old} and the previous policy π_{old^-}, parameterizing the policy by π_θ, the off-policy safety constrained optimization problem can be expressed as

$$L^{\text{CLIP}}_{\text{Safe-SRPPO}}(\theta) = L^{\text{CLIP}}_{\text{SRPPO}}(\theta) - \varrho \sum_i \kappa_i \sum_j^m \Phi(\hat{J}^i_{C_j}(\pi_\theta)) \tag{32}$$

when $i=0$, the policy refers to $\pi_{\theta_{\text{old}}}$; when $i=1$, the policy refers to $\pi_{\theta_{\text{old}^-}}$. ϱ is the safe policy penalty factor. The substitute function of the constraint $\hat{J}^i_{C_j}(\pi_\theta) = J^i_{C_j}(\pi_\theta) - d_j$ and the indicator function $\Phi(\cdot)$ are expressed as follows:

$$\Phi(\hat{J}^i_{C_j}(\pi_\theta)) = \begin{cases} 0, & \text{if } \hat{J}^i_{C_j}(\pi_\theta) \le 0 \\ L^{\text{CLIP}}_{i,C_j}(\theta), & \text{if } \hat{J}^i_{C_j}(\pi_\theta) > 0 \end{cases}$$

where $L^{\text{CLIP}}_{i,C_j}(\theta) = \mathbb{E}_{(s,a)\sim\tau_i}[\max(r_i(\theta)A^{\pi_{\theta_{\text{old}}}}_{C_j}(s,a), \text{clip}(r_i(\theta), \varsigma_i - \varepsilon', \varsigma_i + \varepsilon')A^{\pi_{\theta_{\text{old}}}}_{C_j}(s,a)) + (1-\gamma)(J_{C_j}(\pi_{\theta_i}) - d_j)]$.

Algorithm 1 shows the pseudocode for Safe-SRPPO.

Algorithm 1 Safe-SRPPO

Input: Current policy $\pi_{\theta_{\text{old}}}$; Previous policy $\pi_{\theta_{\text{old}^-}}$; Initial critic value network parameters ϕ_0; Initial risk value network parameters $\xi^j_0\ \forall j$

for $k = 0, 1, 2, \ldots$ **do**
  for actor $= 1, 2, \ldots, N$ **do**
    Interact with environment using policy $\pi_{\theta_{\text{old}}}$ to collect training trajectory samples $\mathcal{D}_k = \{\tau_i\}$;
    Calculate $\hat{R}_t$, and based on critic value function V_{ϕ_k}, calculate reward-based advantage estimate function $\hat{A}^R_t$;
    Calculate $\hat{C}^j_t$, and based on risk value function $V_{\xi^j_k}$, calculate risk-based advantage estimate function $\hat{A}^{C_j}_t$;
  end
  Optimize the objective function:
  for $l = 1, 2, \ldots, L$ **do**
    Calculate $L^{\text{CLIP}}_{\text{SRPPO}}(\theta)$, $L^{\text{CLIP}}_{i,C_j}(\theta)$ for $\pi_{\theta_{\text{old}}}$ based on $\mathcal{D}_k$, and for $\pi_{\theta_{\text{old}^-}}$ based on $\mathcal{D}_{k-1}$, where $i=0$ for $\pi_{\theta_{\text{old}}}$ and $i=1$ for $\pi_{\theta_{\text{old}^-}}$;
    Update policy network parameters: $\theta \leftarrow \theta + \eta \nabla L^{\text{CLIP}}_{\text{Safe-SRPPO}}(\theta)$;
    if $\frac{1}{|\mathcal{D}_k|T}\sum_{\tau\in\mathcal{D}_k}\sum_{t=0}^{T} D_{\text{KL}}\left(\pi_\theta \| \pi_{\theta_{\text{old}}}\right)[s_t] > \delta$ **then**
      break
    end
  end
  $\theta_{\text{old}^-} \leftarrow \theta_{\text{old}}$; $\theta_{\text{old}} \leftarrow \theta$;
  Fit value functions using MSE regression:
  $\hat{L}_V(\phi) = \frac{1}{|\mathcal{D}_k|T}\sum_{\tau\in\mathcal{D}_k}\sum_{t=0}^{T}\left(V_\phi(s_t) - \hat{R}_t\right)^2$;
  $\hat{L}_V(\xi_j) = \frac{1}{|\mathcal{D}_k|T}\sum_{\tau\in\mathcal{D}_k}\sum_{t=0}^{T}\left(V_{\xi_j}(s_t) - \hat{C}^j_t\right)^2$;
  Update value network parameters: ϕ; ξ_j;
end

V. Simulation and Analysis

This section investigates the performance of the algorithms on an unsignalized intersection scenario (as shown in Fig. 1) using Gym [39], a general reinforcement learning platform. The SUMO simulation environment is built by connecting Gym's interface to an XML configuration file that specifies the intersection design, vehicle parameters, traffic density and route information. Traci is used to extract relevant state values for each vehicle from SUMO and we develop the algorithms based on PyTorch [40], a neural network framework. Fig. 6 shows the overall simulation logic.

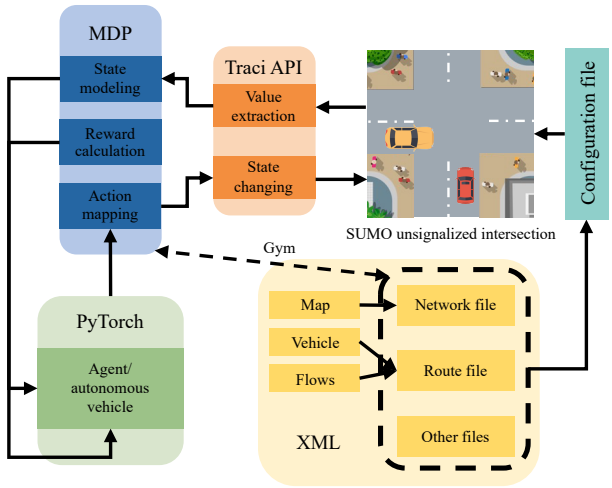


Fig. 6. Simulation training framework.

A. Parameter Settings

1) Vehicle Tracking Model Parameters: We employed the Krauss vehicle following model within the SUMO environment to simulate dynamic traffic flows. This model was chosen for its capacity to produce smooth speed transitions and maintain close distances during deceleration in single-lane traffic scenarios. To add realism, vehicles were generated at the east and west ends of lanes with speeds exceeding 10 m/s, and their arrival time was randomized with a probability of $h = 0.15$ per second, introducing variability and mimicking real-world traffic conditions. Table I shows the parameters of the vehicle following model.

TABLE I
Krauss Model Parameters

Description	Symbol	Value
Minimal safety distance	ϑ	2.5 m
Departing speed	v_i^{set}	$\geq$ 10 m/s
Maximum acceleration	$a_{\max}$	2.6 m/s^2
Expected deceleration	b	4.5 m/s^2
Maximum deceleration in emergency cases	b_e	9 m/s^2
Extra delay time before driving after stopping	κ	0.1 s
Imperfection of the driver's driving behavior	σ	0.5
Minimal following time gap	τ_i	1 s

2) Simulation Training Parameters: In this simulation scenario, the discrete granularity of relative longitudinal distance and relative transverse distance in the semantic scene graph are $M = 18$ and $N = 28$, respectively. The relative coordinate range for the start and end transverse coordinates is $[XStart, XEnd] = [-80, 80]$, and the relative coordinate range for the start and end longitudinal coordinates is $[YStart, YEnd] =$

$[-5, 40]$. The continuous speed space for the self-driving vehicle is $\mathcal{A} = [-4, 4]$.

The specific values of the reward function parameters are shown in Table II. The starting point of the specific numeric settings is to make the absolute value of the reward function around 1. Therefore, the relationship between the terminal state related factors of the main-line reward is designed as shown in Table II. If the final mainline reward is diluted, the number of episodes of running is increased, and the vehicle hardly moves. Therefore, it is necessary to ensure that the boundary range of the auxiliary reward/0.08 is greater than the absolute value of the main-line reward (note that when γ is 0.95, $0.95^{50} = 0.08$. Similarly, if γ is 0.99, then $0.99^{50} = 0.6$). Therefore, the numerical values of the auxiliary reward factors designed in Table II can meet different γ values.

TABLE II
REWARD FUNCTION PARAMETERS

Description	Symbol	Value
Arrival factor of mainline reward	c_{arr}	1.0
Collision factor of mainline reward	c_{col}	2.0
Timeout factor of mainline reward	c_{out}	1.0
Episode limit time length	τ_m	128
Efficiency factor of auxiliary reward	c_{eff}	0.01
Wait and brake factor of auxiliary reward	c_{wab}	0.01
Wait or brake factor of auxiliary reward	c_{wob}	0.005
No wait or brake factor of auxiliary reward	c_{nwb}	0.01
Emergency brake factor of auxiliary reward	c_{teb}	0.01

The number of interactive periods (epochs, corresponding to the number of policy updates) is 250. The complete parameter settings are shown in Table III.

Remark 4: To accelerate the learning process, we use parallel actors throughout the entire training process. Specifically, 16 actors learn simultaneously. During each iteration, every actor interacts with the environment independently and collects data for one epoch. The collected data is then used to calculate a local gradient, which is combined with other local gradients to obtain a global gradient through averaging.

B. Tasks Evaluation and Analysis

There are six metrics to evaluate algorithms: 1) Average cumulative return, which measures the average reward obtained by the ego vehicle per episode; 2) Average cumulative risk, which measures the average risk of collision for the ego vehicle per episode; 3) Success rate, which measures how often the ego vehicle reaches its destination within an epoch; 4) Collision rate, which measures how often the ego vehicle collides with any other vehicles before reaching its destination per epoch; 5) Output action value, which measures the value of the action executed by the policy network after receiving an observation value, ranging from [−1, 1] after applying a Tanh activation function; and 6) Average episode length, which measures how many steps it took for the ego vehicle to complete an episode, including both successful and failed cases.

TABLE III
TRAINING HYPERPARAMETERS

Description/symbol	Value
Hidden layer number	2
Hidden units number	128
Numbers of epochs	250
Steps per epoch	2^{14}
Number of steps	250×2^{14}
Target clipping factor ε	0.2
Sample reuse target clipping factor ε'	0.1
Learning rate of policy optimizer	0.0003
Learning rate of value function optimizer	0.001
Discount factor γ	0.99
GAE factor λ	0.97
KL divergence boundary value δ	0.012
Maximum number of gradient descent steps	80
Simulation step length	0.2 s
Policy weight κ	{0.5, 0.5}
Safe policy penalty factor ϱ	10
Random vehicle arrival probability per second h	0.15

We train the ego vehicle with these four RL algorithms to complete three tasks, i.e., left turn, going straight and right turn. The ego vehicle starts at the stop line and observes the whole traffic environment, interacts with surrounding vehicles, and eventually travels 40 meters to reach the destination through different reinforcement learning algorithms and performing the corresponding output actions. Taking into account the safe policy, the collision risk c_{E} value is set to 2.0 and the long-term risk constraint is set to 0.2 for these tasks.

1) Left Turn Task: In our simulation of a self-driving vehicle tasked with navigating an intersection safely, we observed notable differences in the convergence performance of various algorithms (Fig. 7). All algorithms utilized our semantic scene representation to capture traffic interaction information and demonstrated effective task completion, as indicated by stable average cumulative returns (Fig. 7(a)) and high success rates (over 0.95) (Fig. 7(c)). PPO, while effective, showed a higher collision risk (approximately 1.9) and collision rate (about 0.03), highlighting safety concerns. SRPPO, employing an experience reuse method, led to more aggressive driving behavior, reflected in slightly lower returns (around 0.92) and success rates (0.95). Safe-PPO marked a significant advancement in safety, reducing both collision risk (below 0.5) and collision rate (under 0.01). However, it demonstrated a slower convergence rate compared to SRPPO, with marginally lower cumulative returns (approximately 0.96) and success rates (about 0.97). Its cautious approach, characterized by longer episode durations (about 50 steps or 10 s), suggests a trade-off between safety and efficiency. The standout performer in our simulations was Safe-SRPPO, which excelled in both safety metrics and convergence efficiency (Figs. 7(b) and 7(d)). It achieved the highest average cumulative return (1.04) and success rate (0.99), while maintaining

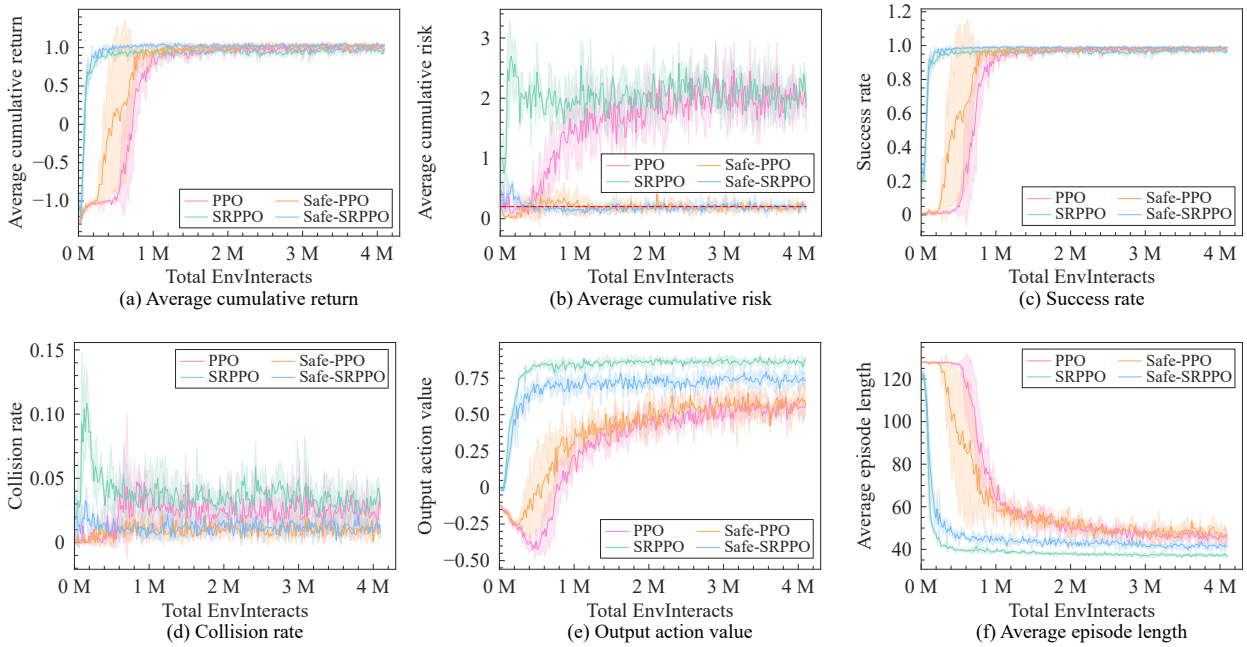


Fig. 7. Training results of left turn task.

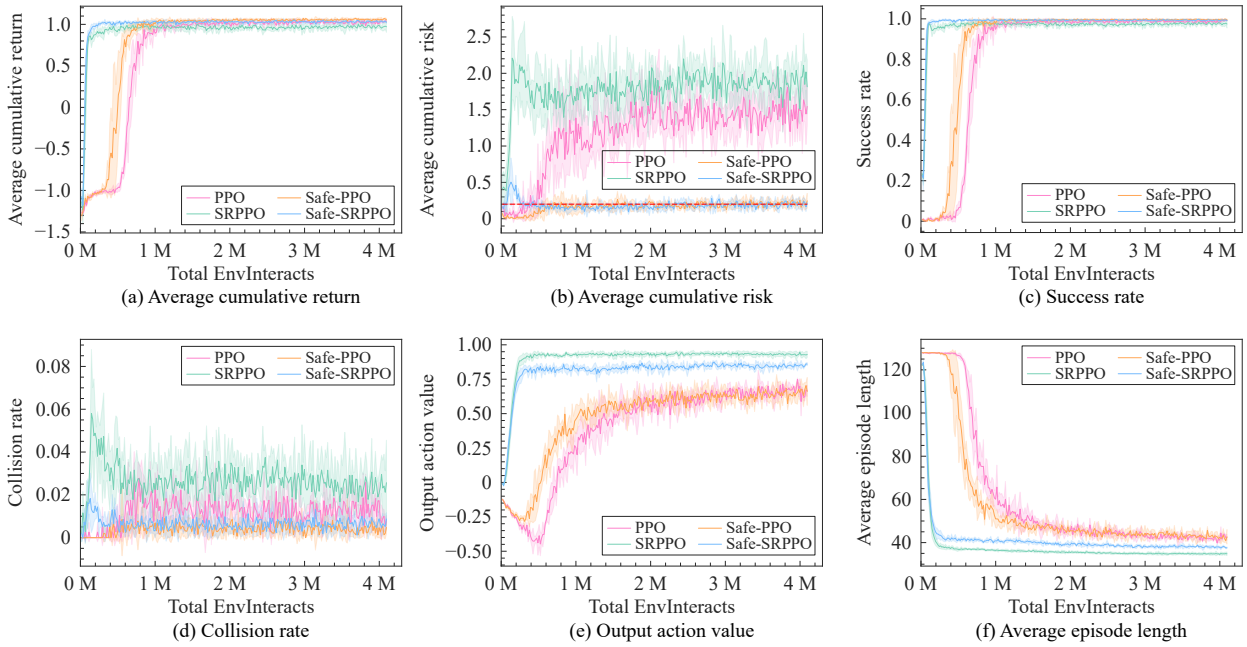


Fig. 8. Training results of going straight task.

the lowest average cumulative risk value and collision rate (below 0.2 and under 0.01, respectively). This algorithm uniquely combined the safety-oriented policy of Safe-PPO with the experience reuse strategy of SRPPO, resulting in optimized policy adjustments without significant deviations from the original policy. This approach not only enhanced sample efficiency but also improved exploration, leading to more robust and quicker convergence.

2) Going Straight Task: In this task, the ego vehicle only needs to consider how to traverse the intersection, without considering the following behavior. As shown in Figs. 8(a) and 8(c), in going straight task, the average cumulative return curves could converge and have a high success rate under different algorithms; as shown in Figs. 8(b) and 8(d), the average cumulative risk value and collision rate are lower than the corresponding values of turn left task, so from the concrete execution behavior, as shown in Figs. 8(e) and 8(f), the output action values increase and the average episode length decreases correspondingly.

3) Right Turn Task: When making a right turn, the distance required for turning is shorter and following behavior needs to be implemented after successful turn. This task is similar to high-speed merging scene. As shown in Figs. 9(a) and 9(c), in the right turn task, both average accumulated return and success rate can converge under different algorithms. Figs. 9(b) and 9(d) show that the average accumulated risk values are

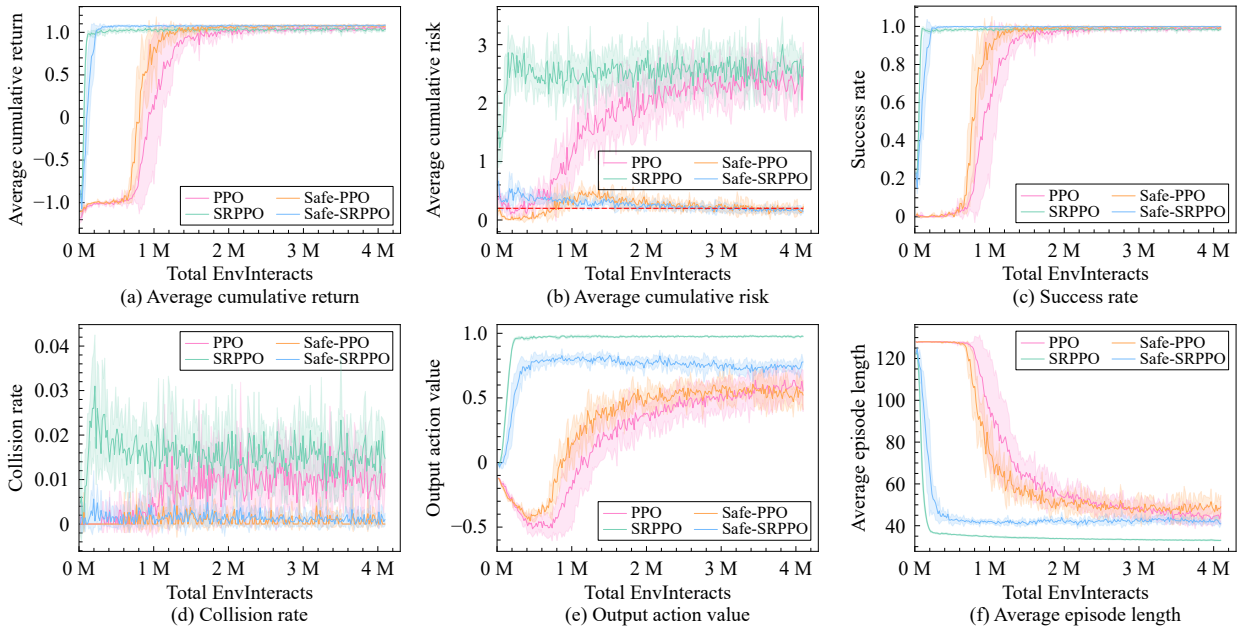


Fig. 9. Training results of right turn task.

lower than those of left turn and going straight tasks, and the collision rate is also lower and close to zero compared with those of left turn and going straight tasks. Besides, from the perspective of concrete action outputs (Figs. 9(e) and 9(f)), the convergence trend of outputs of different algorithms in left turn and going straight tasks is similar, and the same goes for the average episode length.

Under different tasks, PPO converges to stable policies but requires relatively high numbers of interactions. SRPPO increases sampling efficiency but has higher collision rate and risk values, indicating a more aggressive policy. Safe-PPO reduces collision rate by applying a risk function as constraint, but still requires high numbers of interactions for convergence. Safe-SRPPO combines experience reuse and safety constraint policy, overcoming these weaknesses and enabling safe and efficient intersection passing. All algorithms achieve stable and good performance under our MDP and training frameworks, converging to average cumulative returns of 1 with low collision rates.

4) Policy Transfer Effect Among Different Tasks: We evaluate the transfer of policy knowledge by testing a set of 12 policies. Each policy is initially trained on one of three distinct driving tasks. We use neural networks to directly map the output of these policies to vehicle actions. To determine how effective each policy is when faced with tasks it was not trained on, we conduct 1000 test episodes for every combination of policy and task. This comprehensive testing approach allows us to thoroughly assess the adaptability and generalization capability of the policies across different task environments. Figs. 10 and 11 show the success rate and average episode return, respectively. The four algorithms are represented by circular graph with three concentric circles, from outside to inside, representing that the algorithm is trained under left turn task (LT), going straight (GS) and right turn (RT) tasks respectively; The radial direction corresponds to testing the above polices under LT, GS and RT tasks in a

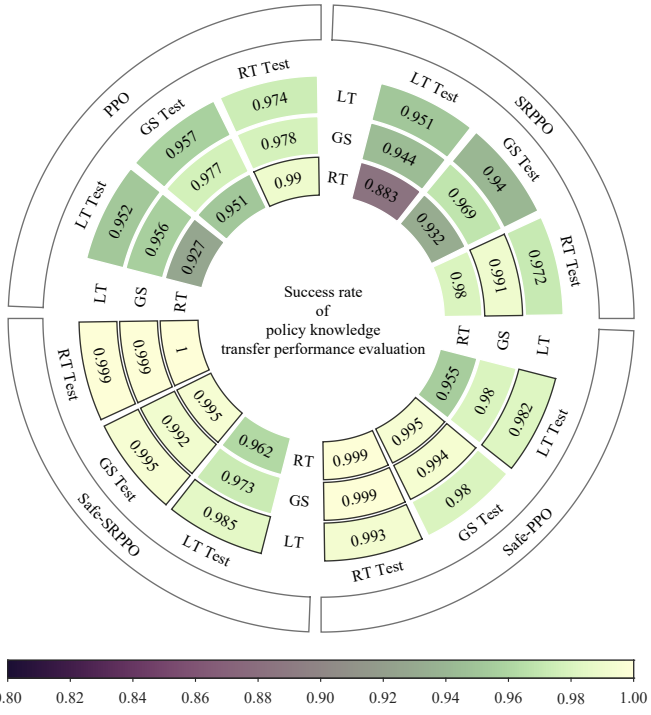


Fig. 10. Success rate of policy transfer test.

clockwise direction. The success rates above 0.98 are highlighted with black frames. All four algorithms achieved success rates higher than 85%, demonstrating good policy transferability across different tasks. Safe-SRPPO has the best transfer performance, especially on the RT task, where it achieves a near-perfect success rate. The RT task is also the easiest for all algorithms to transfer to, followed by the GS task. Safe-PPO performs exceptionally well on these two tasks, with a success rate close to 100%. When testing on the same task as training, all algorithms performed well, with success rates above 95%. Fig. 11 shows the average episode

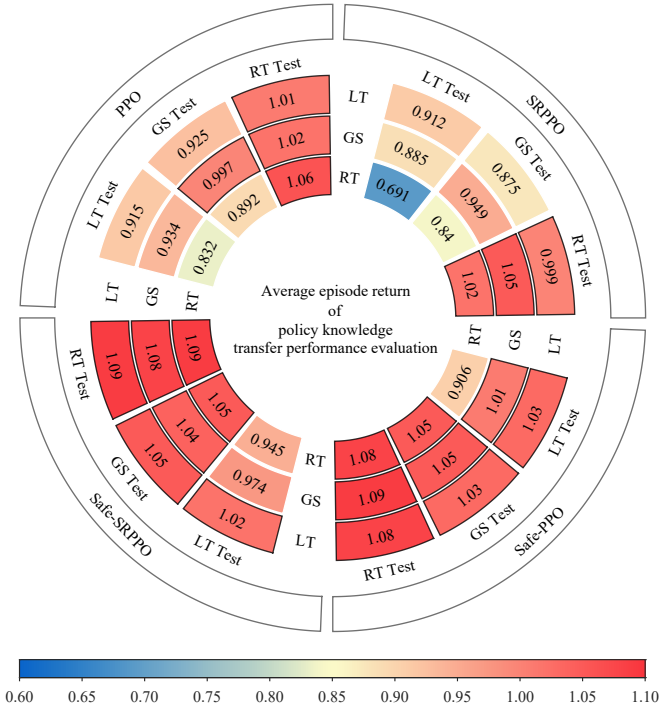


Fig. 11. Average episode return of policy transfer test.

return over 1000 episodes. The highest average reward (1.09) was obtained by Safe-SRPPO tested on the RT task. The lowest average reward (0.691) was obtained by SRPPO trained on the RT task and tested on the LT task. This is because SRPPO converged to a large action value that reflects an aggressive driving style, which differs from the optimal path for the LT task and does not consider safety collision warnings.

5) Parallel Number Impact Analysis: We train our Safe-SRPPO algorithm on the left turn task using different numbers of CPUs and its impact on training time and performance shown in Fig. 12. The algorithm quickly converged to a stable value under all parallel settings (Fig. 12(a)). The training time varied from 36 minutes (32 CPUs) to 2 hours 6 minutes (4 CPUs), with a negative correlation with CPU number (Fig. 12(b)). The exception was 48 CPUs, which took longer than expected due to resource limitations on our server. The best performance was achieved with 16 CPUs, which resulted in the fastest convergence rate and moderate time consumption.

VI. Conclusion and Future Work

This paper proposes an RL framework for decision-making at unsignalized intersections with both self-driving and manually-driven vehicles. The framework consists of three steps: 1) Building a mixed traffic scenario and setting up three intersection tasks (left turn, going straight, right turn); 2) Establishing a POMDP, designing a semantic scene graph based on the observations of self-driving vehicles, and developing a reward function that balances qualitative targets and effective exploration; and 3) Proposing a CMDP to ensure safe driving of the self-driving vehicle and a method to calculate the time of V2V conflict based on motion relations for quantitative collision risk. The framework is evaluated in the SUMO unsignalized intersection environment with four RL algorithms under different intersection tasks. The experimental results show that all four RL algorithms can complete the tasks successfully, with the safe-SRPPO algorithm achieving the best performance with a success rate close to 100%. This study focuses on single-vehicle decision-making in dynamic traffic environments, provides essential insights into RL-based decision-making at unsignalized intersections, underscoring its potential to enhance traffic safety and efficiency. While our research initially concentrates on the autonomous navigation principles of an individual vehicle, it sets a foundation for expanding into more complex scenarios involving multiple agents. Future work will not only explore interactions among multiple vehicles in traffic but also extend the proposed framework to address intricate and lifelike traffic situations, including environments with multiple intersections, diverse pedestrian activities, and varied traffic light systems. This progression will involve the application of alternative RL algorithms and the refinement of semantic scene graph structures, aiming to significantly improve learning efficiency and generalization capabilities of autonomous vehicles. These advancements are essential for adapting to a broad range of traffic conditions in urban, suburban, and rural areas, and for integrating these methodologies into various vehicle types and traffic systems, thereby enhancing the comprehensiveness and applicability of our research in the dynamic field of autonomous driving.

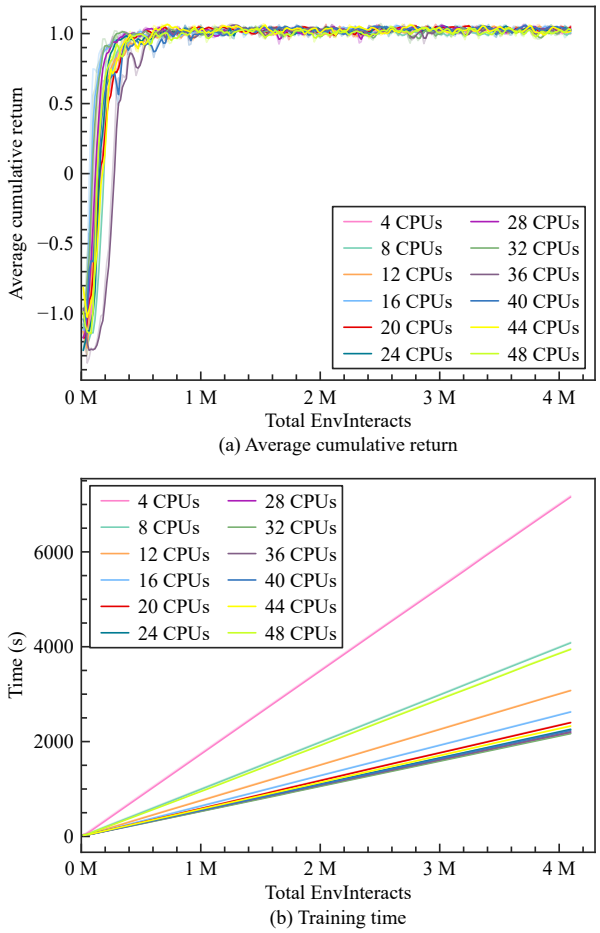


Fig. 12. Training results with different number of CPUs using the Safe-SRPPO algorithm.

REFERENCES

[1] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annu. Rev. Control Rob. Auton. Syst.*, vol. 1, no. 1, pp. 187–210, 2018.

[2] S. Mariani, G. Cabri, and F. Zambonelli, "Coordination of autonomous vehicles," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–33, Jan. 2022.

[3] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *J. Artif. Intell. Res.*, vol. 31, pp. 591–656, Mar. 2008.

[4] X. Chen, M. Hu, B. Xu, Y. Bian, and H. Qin, "Improved reservation-based method with controllable gap strategy for vehicle coordination at non-signalized intersections," *Physica A*, vol. 604, p. 127953, Oct. 2022.

[5] R. Tian, N. Li, I. Kolmanovsky, Y. Yildiz, and A. R. Girard, "Game-theoretic modeling of traffic in unsignalized intersection network for autonomous vehicle control verification and validation," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2211–2226, Mar. 2022.

[6] N. Li, Y. Yao, I. Kolmanovsky, E. Atkins, and A. R. Girard, "Game-theoretic modeling of multi-vehicle interactions at uncontrolled intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1428–1442, Feb. 2022.

[7] S. Yan, T. Welschehold, D. Buscher, and W. Burgard, "Courteous behavior of automated vehicles at unsignalized intersections via reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 1, pp. 191–198, Oct. 2021.

[8] N. Parvez Farazi, B. Zou, T. Ahamed, and L. Barua, "Deep reinforcement learning in transportation research: A review," *Transp. Res. Interdiscip. Persp.*, vol. 11, p. 100425, Sept. 2021.

[9] F. Azadi, N. Mitrovic, and A. Stevanovic, "Impact of shared lanes on performance of the combined flexible lane assignment and reservation-based intersection control," *Transp. Res. Rec.*, vol. 2676, no. 12, pp. 51–68, Dec. 2021.

[10] Z. Guo, D. Sun, and L. Zhou, "Game algorithm of intelligent driving vehicle based on left-turn scene of crossroad traffic flow," *Comput. Intell. Neurosci.*, vol. 2022, p. e9318475, Sept. 2022.

[11] P. Hang, C. Lv, and X. Chen, "Human-like decision making for autonomous vehicles with noncooperative game theoretic method," *Human-Like Decision Making and Control for Autonomous Driving*. CRC Press, 2022.

[12] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[13] D. P. Bertsekas, "Feature-based aggregation and deep reinforcement learning: A survey and some new implementations," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 1, pp. 1–31, Jan. 2019.

[14] Z. Zhu and H. Zhao, "A survey of deep rl and il for autonomous driving policy learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, p. 14, Sept. 2022.

[15] 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 Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 12, pp. 5526–5538, Dec. 2021.

[16] J. Wu, Z. Huang, W. Huang, and C. Lv, "Prioritized experience-based reinforcement learning with human guidance for autonomous driving," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 1, pp. 855–869, Jan. 2024.

[17] Y. Wang, S. Hou, and X. Wang, "Reinforcement learning-based birdview automated vehicle control to avoid crossing traffic," *Comput.-Aided Civ. Infrastruct. Eng.*, vol. 36, no. 7, pp. 890–901, Jul. 2021.

[18] C. Huang, R. Zhang, M. Ouyang, P. Wei, J. Lin, J. Su, and L. Lin, "Deductive reinforcement learning for visual autonomous urban driving navigation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 12, pp. 5379–5391, Dec. 2021.

[19] 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 Trans. Veh. Technol.*, vol. 71, no. 1, pp. 41–52, Jan. 2022.

[20] Z. Qiao, K. Muelling, J. Dolan, P. Palanisamy, and P. Mudalige, "POMDP and hierarchical options MDP with continuous actions for autonomous driving at intersections," in *Proc. 21st Int. Conf. Intelligent Transportation Systems*, Nov. 2018, pp. 2377–2382.

[21] Y. Ren, J. Duan, S. E. Li, Y. Guan, and Q. Sun, "Improving generalization of reinforcement learning with minimax distributional soft actor-critic," in *Proc. IEEE 23rd Int. Conf. Intelligent Transportation Systems*. IEEE, 2020, pp. 1–6.

[22] H. Seong, C. Jung, S. Lee, and D. H. Shim, "Learning to drive at unsignalized intersections using attention-based deep reinforcement learning," in *Proc. IEEE Int. Intelligent Transportation Systems Conf.*. Indianapolis, USA: Oct. 2021, pp. 559–566.

[23] M. Martinson, A. Skrynnik, and A. I. Panov, "Navigating autonomous vehicle at the road intersection simulator with reinforcement learning," in *Proc. Russian Conf. Artificial Intelligence*, ser. Lecture Notes in Computer Science, S. O. Kuznetsov, A. I. Panov, and K. S. Yakovlev, Eds. Cham: Springer, 2020, pp. 71–84.

[24] R. Bautista-Montesano, R. Galluzzi, K. Ruan, Y. Fu, and X. Di, "Autonomous navigation at unsignalized intersections: A coupled reinforcement learning and model predictive control approach," *Transp. Res. Part C Emerg. Technol.*, vol. 139, p. 103662, Jun. 2022.

[25] E. Candela, O. Doustaly, L. Parada, F. Feng, Y. Demiris, and P. Angeloudis, "Risk-aware controller for autonomous vehicles using model-based collision prediction and reinforcement learning," *Artif. Intell.*, vol. 320, p. 103923, Jul. 2023.

[26] M. Selim, A. Alanwar, M. W. El-Kharashi, H. M. Abbas, and K. H. Johansson, "Safe reinforcement learning using data-driven predictive control," in *Proc. 5th Int. Conf. Communications, Signal Processing, and Their Applications*, Dec. 2022, pp. 1–6.

[27] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," arXiv preprint arXiv: 1707.06347 [cs], Jul. 2017.

[28] L. Zhang, R. Zhang, T. Wu, R. Weng, M. Han, and Y. Zhao, "Safe reinforcement learning with stability guarantee for motion planning of autonomous vehicles," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 12, pp. 5435–5444, Jul. 2021.

[29] T. De Ceunynck, E. Polders, S. Daniels, E. Hermans, T. Brijs, and G. Wets, "Road safety differences between priority-controlled intersections and right-hand priority intersections: behavioral analysis of vehicle-vehicle interactions," *Transp. Res. Rec.*, vol. 2365, no. 1, pp. 39–48, Jan. 2013.

[30] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lucken, J. Rummel, P. Wagner, and E. Wiessner, "Microscopic traffic simulation using SUMO," in *Proc. 21st Int. Conf. Intelligent Transportation Systems*, Nov. 2018, pp. 2575–2582.

[31] J. Song, Y. Wu, Z. Xu, and X. Lin, "Research on car-following model based on SUMO," in *Proc. IEEE/Int. Conf. Advanced Infocomm Technology*, Nov. 2014, pp. 47–55.

[32] H. Li, G. Yu, B. Zhou, P. Chen, Y. Liao, and D. Li, "Semantic-level maneuver sampling and trajectory planning for on-road autonomous driving in dynamic scenarios," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1122–1134, Feb. 2021.

[33] Y. Qin, W. Hua, J. Jin, J. Ge, X. Dai, L. Li, X. Wang, and F.-Y. Wang, "AUTOSIM: Automated urban traffic operation simulation via meta-learning," *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 9, pp. 1871–1881, 2023.

[34] R. K. R. Pallavali, "Synchronous intelligent intersections for sustainable urban mobility," Ph.D. dissertation, Universidade do Porto, Portugal, 2023.

[35] S. Das and A. K. Maurya, "Defining time-to-collision thresholds by the type of lead vehicle in non-lane-based traffic environments," *IEEE Trans. Intell. Transport. Syst.*, vol. 21, no. 12, pp. 4972–4982, Dec. 2020.

[36] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *Proc. 34th Int. Conf. Machine Learning*. PMLR, Jul. 2017, pp. 22–31.

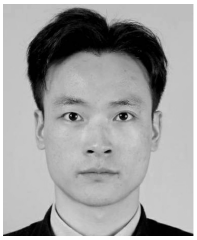
[37] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. 32nd Int. Conf. Machine Learning*. PMLR, Jun. 2015, pp. 1889–1897.

[38] T.-Y. Yang, J. Rosca, K. Narasimhan, and P. J. Ramadge, "Projection-based constrained policy optimization," in *Proc.8th Int. Conf. Learning Representations*, Apr. 2020.

[39] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, *OpenAI Gym*, Jun. 2016.

[40] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T.

Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., 2019.



Xiaolong Chen received the B.E. degree from Fuzhou University in 2016 and the Ph.D. degree from Hunan University in 2023, respectively. His research interests include cooperative control methods for connected and automated vehicles, single-agent and multi-agent reinforcement learning, as well as decision control for signal-free intersections. He has served as a Peer Reviewer for several prestigious journals such as *IEEE Transactions on Intelligent Transportation Systems*, *IEEE/ASME Transactions on Mechatronics*, *IEEE Transactions on Computing*, *IEEE Internet of Things Journal,* and *IEEE Transactions on Vehicular Technology*.

Biao Xu (Member, IEEE) received the B.E. degree and the Ph.D. degree from Tsinghua University, in 2013 and 2018, respectively. From 2016 to 2017, he was a Visiting Scholar with University of Washington, USA. He is currently an Associate Research Fellow at the College of Mechanical and Vehicle Engineering, Hunan University. His research interests include connected and automated vehicles, vehicle control, and V2I cooperation. Dr. Xu was the recipient of Best Paper Award in 14th Intelligent Transportation Systems Asia-Pacific Forum in 2015, and Best Paper Award in the 2017 IEEE Intelligent Vehicle Symposium.

Manjiang Hu received the B.Tech. degree and the Ph.D. degree from Jiangsu University, in 2009 and 2014, respectively. He worked as a Postdoctor in the Department of Automotive Engineering, Tsinghua University from 2014 to 2017. He is currently a Professor with the College of Mechanical and Vehicle Engineering, Hunan University. His research interests include cooperative driving assistance technology and vehicle control.

Yougang Bian (Member, IEEE) received the B.E. and Ph.D. degrees from Tsinghua University in 2014 and 2019, respectively. He was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of California at Riverside, USA, from 2017 to 2018. He is currently an Associate Professor at Hunan University. His research interests include distributed control, cooperative control, and their applications to connected and automated vehicles. Dr. Bian was a recipient of the Best Paper Award at the 2017 IEEE Intelligent Vehicles Symposium.

Yang Li received the B.E. degree from Chongqing University in 2014 and the Ph.D. degree from Tsinghua University in 2020, respectively. She is now working as an Assistant Professor with Hunan University. Her current research interests include deep reinforcement learning, multi-agent reinforcement learning, safe reinforcement learning, and transfer learning in the decision-making of automated vehicles.

Xin Xu (Senior Member, IEEE) received the Ph.D. degree in control science and engineering from the College of Mechatronics and Automation, National University of Defense Technology (NUDT) in 2002. He has been a Visiting Professor with Hong Kong Polytechnic University, Hong Kong, China; the University of Alberta, Canada, University of Guelph, Canada, and the University of Strathclyde, U.K. He is currently a Professor with the College of Intelligence Science and Technology, NUDT. His research interests include intelligent control, reinforcement learning, approximate dynamic programming, machine learning, robotics, and autonomous vehicles. Dr. Xu was the recipient of the National Science Fund for Outstanding Youth in China and the Second-Class National Natural Science Award of China. He has served as an Associate Editor or a Guest Editor for *Information Sciences, International Journal of Robotics and Automation*, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *Intelligent Automation and Soft Computing, International Journal of Adaptive Control and Signal Processing*, and *Acta Automatica Sinica*. He is a Member of the IEEE CIS Technical Committee on Approximate Dynamic Programming and Reinforcement Learning and the IEEE RAS Technical Committee on Robot Learning.