Safe Control of Autonomous Vehicles in Overtaking Maneuvers Using Game-Theoretic Learning-based Predictive Controller

Shaowei Yuan, Jingjing Jiang, Sarah Spurgeon and Boli Chen

***Abstract*— This work proposes a safe control strategy for an autonomous vehicle to overtake a human-driven vehicle (HDV) using a predictive safety filter (PSF) mechanism that hierarchically combines an end-to-end Reinforcement Learning (RL) agent with a predictive controller. To create a more realistic RL environment, a Stackelberg game based on a first-principles model is employed to capture the HDV's real-time response during overtaking rather than relying on a predefined empirical or purely statistical driver model. In the lower layer, a distributionally robust chance-constrained predictive controller is implemented to manage uncertainties in HDV behavior, ensuring robust safety guarantees. The effectiveness of the proposed synthetic controller is verified in a gym environment with comparisons against traditional schemes.**

I. Introduction

Autonomous overtaking is one of the most common maneuvers, often involving complex interactions between CAVs and HDVs, making it a key focus of research in CAVs control [1]. Various approaches have been developed in the literature to address the overtaking problem, with model predictive control (MPC) standing out as one of the most widely used solutions due to its ability to handle safety constraints [2]–[4]. However, MPC methods heavily rely on prior modeling of the road environment, including obstacle vehicles, making it challenging to generalize to dynamic scenarios.

Machine learning methods enable agents to learn optimal policies or models from data, bypassing explicit modeling, which shows great promise in complex autonomous driving environments [5]–[8]. End-to-end RL, which directly uses raw sensor data as input and generates control actions, offers a straightforward solution and has been successfully implemented in autonomous overtaking, as demonstrated in [9]. Alternatively, RL can be applied to the vehicle trajectory planning layer, followed by a low-level controller, resulting in a hierarchical framework [10]. Methods that specifically integrate RL with MPC in such a framework can be found in [11]. However, current RL-based overtaking solutions lack interpretability and safety guarantees [6], which are essential for autonomous vehicles.

A key challenge in autonomous overtaking lies in modeling the interaction between the CAVs and the obstacle HDVs, which inherently involves uncertain human driver behaviors [1], [12]. Traditional methods assume that the HDVs follow empirical driver models, such as the Intelligent Driver Model (IDM) [13]. However, these models are typically based on deterministic rules derived from microscopic driving behavior and fail to accurately capture the uncertainties of real human drivers [12]. Approaches like deep learning [14] and inverse RL [15] have been used to model HDVs behaviors, but these methods require large datasets, which may not be feasible for individual drivers [16]. Game theory is emerging as a promising approach for modeling these interactions [4], [17], [18]. In particular, Stackelberg games, which model interactions as a leader-follower sequential decision-making process, have been successfully applied to solve overtaking and lane-changing problems in mixed-traffic scenarios utilizing MPC [3], [17]. Similarly to other MPC-based methods, these approaches rely on an accurate parametric state space model of the system, which may hinder their successful implementations in practice.

Recently, a promising solution that overcomes the foregoing limitations of MPC and RL is the introduction of PSF [19], where end-to-end RL determines the nominal control law, and MPC is adapted as a safety filter to enforce safety constraints. With its hierarchical configuration, this method leverages the strengths of end-to-end RL in managing complex environments while providing theoretical safety guarantees. Motivated by the PSF, this paper proposes a two-layer control framework, combining RL and MPC, to safely control a CAV to overtake an HDV, as illustrated in Fig. 1. For the sake of brevity, we will refer to the CAV as the ego vehicle (EV) and the HDV as the obstacle vehicle (OV) throughout the remainder of this article. The novelty of this work lies in two key aspects: 1) In the upper RL layer, instead of relying on a predefined driver model, we designed a Stackelberg game-based interaction model to simulate the OV's responses to the EV's movements, which is the key in autonomous overtaking; and 2) Once a control action is determined by the RL layer, it is filtered in the lower layer by a novel stochastic PSF (SPSF), which incorporates carefully designed chance constraints to account for the uncertainty of human behaviors, ensuring safety.

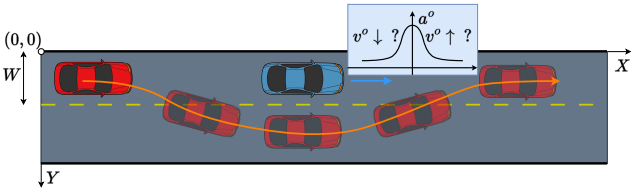


Fig. 1: Illustration of an overtaking problem. The EV (in red) begins positioned behind the OV (in blue). The left lane is denoted as the initial lane, while the right lane is the overtaking lane.

The rest of this paper is organized as follows. Section II explains the overtaking problem settings. In Section III, an overview of the controller framework and design of each module is provided. Moreover, simulation results are presented in Section IV. A conclusion and future research

S. Yuan, S. Spurgeon, and B. Chen are with the Department of Electronic and Electrical Engineering, University College London, London, UK (shaowei.yuan.22@ucl.ac.uk; s.spurgeon@ucl.ac.uk; boli.chen@ucl.ac.uk).

J. Jiang is with the Department of Aeronautical and Automotive Engineering, Loughborough University, UK (email: j.jiang2@lboro.ac.uk)

plan are provided in Section V.

Notations: Let $\mathbb{R}$, $\mathbb{R}_{>0}$ denote the real and strict positive real sets of numbers, respectively. $\mathbb{N}$ denotes the set of natural numbers, $\mathbb{N} = \{0, 1, 2, \ldots\}$ and, for any two integers m and n satisfying $m \leq n$, $\mathbb{N}_{[m,n]} = \{m, m+1, \ldots, n\}$. I_n denotes a $n \times n$ identity matrix. Given a vector $\mathbf{x} \in \mathbb{R}^n$, the Euclidean norm of $\mathbf{x}$ is denoted by $\|\mathbf{x}\|$, and $\|\mathbf{x}\|_W$ is the norm weighted by W. The superscripts $(\cdot)^e$ and $(\cdot)^o$ denote the variables corresponding to EV and OV, respectively. The Minkowski sum of two sets is defined by $\mathbb{A} \oplus \mathbb{B} := \{a + b : a \in \mathbb{A},\ b \in \mathbb{B}\}$. The distance between two sets is defined as $\text{dist}(\mathbb{A}, \mathbb{B}) := \min_r \{\|r\| : (\mathbb{A} \oplus r) \cap \mathbb{B} \neq \emptyset\}$. The operator $v \succeq 0$ means that all elements of v are non-negative. $\Pr_{[\mathbb{P}]}\{\cdot\}$ denotes the probability under distribution $\mathbb{P} \sim (\mu, \sigma^2)$, where μ and σ^2 are mean and variance, respectively.

II. Problem Formulation and Preliminaries

This work considers an overtaking problem on a two-lane, one-way straight highway of uniform lane width W, as shown in Fig 1. The EV starts the overtaking maneuver in the initial lane, changes its lane to the overtaking lane, and eventually merges back into the initial lane. The goal of the EV is to overtake the OV and drive comfortably without collision. The following assumptions are imposed.

Assumption 1: The OV remains in the center of its initial lane throughout the task with an initial speed strictly below the speed limit, that is, $v^o(0) < v_{\max}$.

Assumption 2: EV and OV have identical dimensions (vehicle length l and width w) and physical limits. The center of the mass coincides with the geometric center of the vehicle.

A. Vehicle Dynamics

As illustrated in Fig. 2, we use a kinematic bicycle model [20] to describe the dynamics of the EV. Let p_x^e and p_y^e

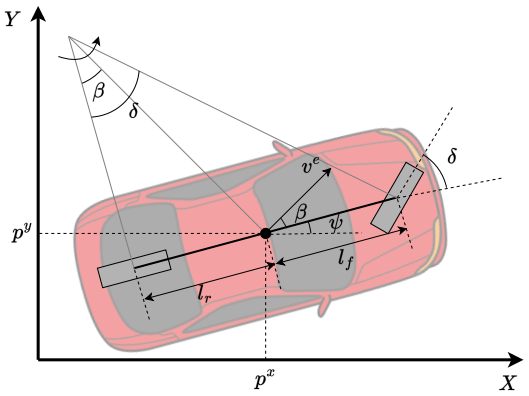


Fig. 2: Kinematic bicycle model of the EV.

represent the position of the EV's center of mass in a Cartesian coordinate system, where X denotes the longitudinal axis and Y the lateral axis. The vehicle speed is denoted by v^e, and ψ represents the orientation angle of the vehicle relative to the global X-axis. Define the state of the EV as $x^e = [p_x^e, p_y^e, v^e, \psi]^\top$ and the control input as $u^e = [a^e, \delta]^\top$. The dynamics of EV are modeled as

$$\dot{x}^e = f(x^e, u^e) = \begin{bmatrix} v^e \cos(\psi + \beta) \\ v^e \sin(\psi + \beta) \\ a^e \\ \dfrac{v^e}{l_r} \sin\beta \end{bmatrix}, \tag{1}$$

where $\beta = \arctan\left(\frac{l_r}{l_f + l_r} \tan\delta\right)$. According to Assumption 2, $l_r = l_f$ are the distances from the center to the rear and front axles, respectively. β is the side-slip angle. The state and input of the EV are respectively constrained by

$$\mathbb{X}^e = \{x^e \in \mathbb{R}^4 | v_{\min} \leq v^e \leq v_{\max}, \psi_{\min} \leq \psi \leq \psi_{\max}\}$$

and

$$\mathbb{U}^e = \{u^e \in \mathbb{R}^2 | a_{\min} \leq a^e \leq a_{\max}, \delta_{\min} \leq \delta \leq \delta_{\max}\}.$$

Under Assumption 1, OV can be modeled by

$$\dot{x}^o = g(x^o, u^o) = [v^o, a^o]^\top \tag{2}$$

with the state $x^o = [p_x^o, v^o]^\top$ and control input $u^o = a^o$. The convex polyhedral constraint on states is defined as $\mathbb{X}^o = \{x^o \in \mathbb{R}^2 | v_{\min} \leq v^o \leq v_{\max}\}$, and the convex polytopic constraint on control input is $\mathbb{U}^o = \{u^o \in \mathbb{R} | a_{\min} \leq a^o \leq a_{\max}\}$. $v_{\max}$ can be set to the legal speed limit while $v_{\min}$ is designed based on the driving environment. To facilitate control, it is reasonable to make the following assumption.

Assumption 3: The state variables x^e and x^o of the EV and the OV can be measured by the EV.

B. Road Object Description

The space occupied by a full-dimensional EV at time k is described as $\mathbb{E}(x_k^e) = p_k^e \oplus R(\psi_k)\mathbb{B}$ [21], where $p_k^e = [p_{x,k}^e, p_{y,k}^e]^\top$ is the position of EV, $\mathbb{B} = \{p \in \mathbb{R}^2 | Ap \leq b\}$ is a convex polytope, which describes the shape of the vehicle at the origin, and $R(\psi_k)$ is the rotation matrix. Then, it can be inferred that $\mathbb{E}(x_k^e) = \{p \in \mathbb{R}^2 | A_k^e(\psi_k) p \leq b_k^e(x_k^e)\}$ with

$$A_k^e(\psi_k) = \underbrace{\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}^\top}_{A} \underbrace{\begin{bmatrix} \cos\psi_k & \sin\psi_k \\ -\sin\psi_k & \cos\psi_k \end{bmatrix}}_{R(\psi_k)}, \tag{3a}$$

$$b_k^e(x_k^e) = \begin{bmatrix} p_{x,k}^e \cos\psi_k + p_{y,k}^e \sin\psi_k + \dfrac{l}{2} \\ -p_{x,k}^e \cos\psi_k - p_{y,k}^e \sin\psi_k + \dfrac{l}{2} \\ -p_{x,k}^e \sin\psi_k + p_{y,k}^e \cos\psi_k + \dfrac{w}{2} \\ p_{x,k}^e \sin\psi_k - p_{y,k}^e \cos\psi_k + \dfrac{w}{2} \end{bmatrix}. \tag{3b}$$

Similarly, the space occupied by OV at time k is described by $\mathbb{O}(x_k^o) = p_k^o \oplus \mathbb{B} = \{p \in \mathbb{R}^2 | Ap \leq b_k^o(p_k^o)\}$, where the position of OV is $p_k^o = [p_{x,k}^o, p_y^o]^\top$ with $p_y^o = W/2$ (remaining in the center of the initial lane) and

$$\begin{aligned} b_k^o(p_k^o) &= \underbrace{\begin{bmatrix} \frac{l}{2} & \frac{l}{2} & \frac{w}{2} & \frac{w}{2} \end{bmatrix}^\top}_{b} + A p_k^o \\ &= \begin{bmatrix} p_{x,k}^o + \frac{l}{2}, -p_{x,k}^o + \frac{l}{2}, p_y^o + \frac{w}{2}, -p_y^o + \frac{w}{2} \end{bmatrix}^\top. \end{aligned} \tag{4a}$$

Collision avoidance between the two vehicles is ensured if

$$\text{dist}(\mathbb{E}(x_k^e), \mathbb{O}(x_k^o)) \geq d_{\min}, \tag{5}$$

which is equivalent to [21]

$$\begin{cases} \lambda_k \succeq 0, \gamma_k \succeq 0, \\ -b_k^{e\top}\lambda_k - b_k^{o\top}\gamma_k \geq d_{\min}, \\ A_k^{e\top}\lambda_k + A^{\top}\gamma_k = \mathbf{0}, \\ \|A^{\top}\gamma_k\| \leq 1. \end{cases} \tag{6}$$

where $\lambda_k \in \mathbb{R}^4$, $\gamma_k \in \mathbb{R}^4$ are the two dual variables.

III. Methodology

The overall framework of the proposed control scheme is shown in Fig. 3. The RL agent generates the control input u^L based on observations and rewards from the environment. If u^L is verified as safe by the PSF, it is then forwarded to the EV for execution. If the PSF predicts that u^L would lead to unsafe behavior, u^L is adjusted to a safe control u^S, where the differences between u^L and u^S are minimized. As the key element of the environment, the OV's behavior during overtaking (when the EV passes through and merges back into the original lane) is characterized by a Stackelberg game-based model, which solves two sequential finite horizon optimal control problems (FHOCPs) to decide the control of the OV. In this section, we will elaborate on the design of each module in Fig. 3. The RL agent, reward function, and training environment, including the game-based model of the OV, are introduced in Section III-A. Section III-B presents a stochastic PSF to account for safety under the uncertainty of human driver behavior.

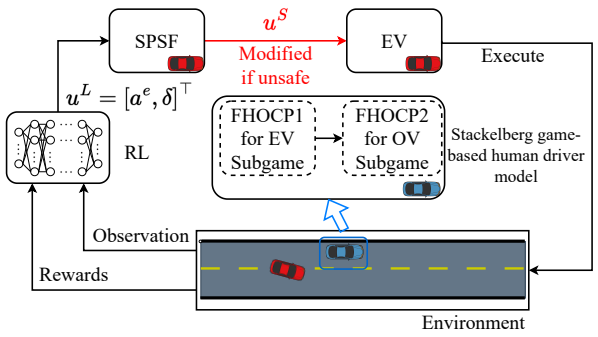


Fig. 3: Control scheme framework.

A. Deep RL-based Controller

Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm [22], an algorithm specific for tasks with continuous action space, is used for training. The action vector is defined as $\mathbf{a} = [a^e, \delta]^\top$, and the state vector is $\mathbf{s} = [p^{e\top}, p^{o\top}, \psi, v^e, v^o]^\top \in \mathbb{R}^7$. When EV runs off the road, the reward is set to 0. Otherwise, the reward is calculated by

$$r = r_{\text{collision}} + r_{\text{behavior}} + r_{\text{lane}} + r_{\text{overtaking}}, \tag{7}$$

and its components are designed as

$$r_{\text{collision}} = \begin{cases} -1, & \text{if a collision occurs,} \\ 0, & \text{otherwise,} \end{cases} \tag{8a}$$

$$r_{\text{overtaking}} = \begin{cases} 2, & \text{success overtaking,} \\ 0, & \text{otherwise,} \end{cases} \tag{8b}$$

$$r_{\text{lane}} = \frac{\alpha_1}{\alpha_2 + \alpha_3 \Delta y^2}, \tag{8c}$$

$$r_{\text{behavior}} = \omega_1 \tilde{v} - \omega_2 \tilde{a}^2 - \omega_3 \tilde{\delta}^2, \tag{8d}$$

where Δy is the lateral distance from the EV to the target lane center. The target lane is determined based on the relative longitudinal distance between the EV and the OV. When the EV is at least twice the vehicle length ahead of the OV, the target lane is set as the initial lane; otherwise, the target lane remains the overtaking lane. r_{lane} is to encourage the EV to stay in the center of the target lane. $\tilde{v}$ represents v^e normalized in the range $[0, 1]$, while $\tilde{a}$ and $\tilde{\delta}$ are the normalized versions of a^e and δ, respectively, mapped in the range $[-1, 1]$. r_{behavior} rewards higher speed to facilitate overtaking while penalizing aggressive acceleration and steering actions to promote smoother driving behavior. $\{\alpha_1, \alpha_2, \alpha_3\}$ and $\{\omega_1, \omega_2, \omega_3\}$ represent weights that need to be tuned.

The next step is to model the behavior of the OV during overtaking. Generally, OV is treated as a part of the environment in the RL training loop, and the OV is assumed to follow a certain model such as the IDM used in traffic flow modeling. However, such models cannot capture human behavior in response to an overtaking maneuver. In this work, we model the interaction between the OV and the EV as a Stackelberg game-based model to simulate human-like behavior of the OV. It is reasonable to assume that the interaction is triggered when the EV is ahead of the OV by half a vehicle length (i.e., $p^e_{x,k} \geq p^o_{x,k} + l/2$), as indicated by the EV's turn signal.

Definition 1: [17] The Stackelberg game is a sequential game between the leader $\mathcal{L}$ and follower $\mathcal{F}$, where $\mathcal{L}$ takes action first, and then $\mathcal{F}$ responds based on the action of $\mathcal{L}$. The Stackelberg equilibrium is the solution of the following optimization problem

$$u^*_{\mathcal{L}} = \arg\min_{u_{\mathcal{L}} \in \mathbb{U}_{\mathcal{L}}} \left(\min_{u_{\mathcal{F}} \in \mathbb{U}^*_{\mathcal{F}}} J_{\mathcal{L}}(x_{\mathcal{L}}, x_{\mathcal{F}}; u_{\mathcal{L}}, u_{\mathcal{F}}) \right), \tag{9a}$$

$$\mathbb{U}^*_{\mathcal{F}}(u_{\mathcal{L}}) \triangleq \{u^*_{\mathcal{F}} \in \mathbb{U}_{\mathcal{F}} : J_{\mathcal{F}}(x_{\mathcal{L}}, x_{\mathcal{F}}; u_{\mathcal{L}}, u^*_{\mathcal{F}}) \leq J_{\mathcal{F}}(x_{\mathcal{L}}, x_{\mathcal{F}}; u_{\mathcal{L}}, u_{\mathcal{F}}), \forall u_{\mathcal{L}} \in \mathbb{U}_{\mathcal{L}}\}, \tag{9b}$$

where x and u denote states and actions, respectively.

By assuming the OV follows a Stackelberg game with EV, where OV is the leader and EV is the follower. Through backward induction, the optimal control input u^{o*}_k is obtained by solving the following two cascaded FHOCPs.

1) EV Subgame: For all given possible actions of OV, EV minimizes the cost function J^e by finding the optimal control sequence $\{u^{e}_{i|k}{}^*\}, i \in \mathbb{N}_{[0,N-1]}$, which is the solution of the following FHOCP1,

$$\min_{u^e_{i|k}, u^o_{i|k}} \quad J^e = \sum_{i=0}^{N-1} \Big[Q_1 (v^o_{i|k} - v^e_{i|k}) + Q_2 {a^e_{i|k}}^2 + Q_3 \delta^2_{i|k} + Q_4 \psi^2_{i|k} - Q_5 \ln\left(\frac{d_{i|k}}{d_{\min}}\right) \Big] \tag{10a}$$

$$\text{s.t.} \quad x^e_{i+1|k} = x^e_{i|k} + f(x^e_{i|k}, u^e_{i|k}) T_s, \forall i \in \mathbb{N}_{[0,N-1]}, \tag{10b}$$

$$x^o_{i+1|k} = x^o_{i|k} + g(x^o_{i|k}, u^o_{i|k}) T_s, \tag{10c}$$

$$x^e_{i|k} \in \mathbb{X}^e, u^e_{i|k} \in \mathbb{U}^e, x^e_{0|k} = x^e(k), \tag{10d}$$

$$x^o_{i|k} \in \mathbb{X}^o, u^o_{i|k} \in \mathbb{U}^o, x^o_{0|k} = x^o(k), \tag{10e}$$

where N is the prediction horizon, and T_s is the sampling time. The first term in J^e is to encourage EV to overtake. When EV is slower than OV, the cost increases, and when EV is faster than OV, the cost decreases. The control input of EV and its heading angle are penalized for a smooth and comfortable overtaking maneuver. The last term uses

a log-barrier function to incentivize a safety gap with $d_{i|k} = \|p^e_{i|k} - p^o_{i|k}\|$. The five objectives are weighted by $Q_j \in \mathbb{R}_{>0}, \forall j \in \{1,\dots,5\}$. $x^e(k)$ and $x^o(k)$ are the states at current time k.

2) OV Subgame: Given $\{{u^e_{i|k}}^*\}$ of the EV, the optimal control action of the OV is obtained by solving the following FHOCP2, and then the state $\mathbf{s}$ gets updated:

$$\min_{u^o_{i|k}} J^o = \sum_{i=0}^{N-1} \left[-R_1 \ln\left(\frac{d_{i|k}}{d_{\min}}\right) + R_2 {a^o_{i|k}}^2 + R_3 (v^o_{i|k} - v^o(k))^2 \right] \tag{11a}$$

$$\text{s.t. } {x^e_{i+1|k}}^* = {x^e_{i|k}}^* + f({x^e_{i|k}}^*, {u^e_{i|k}}^*) T_s, \forall i \in \mathbb{N}_{[0,N-1]}, \tag{11b}$$

$$x^o_{i+1|k} = x^o_{i|k} + g(x^o_{i|k}, u^o_{i|k}) T_s, \tag{11c}$$

$$x^o_{i|k} \in \mathbb{X}^o, u^o_{i|k} \in \mathbb{U}^o, \tag{11d}$$

$${x^e_{0|k}}^* = x^e(k), x^o_{0|k} = x^o(k), \tag{11e}$$

where the third term represents the OV's intention to maintain its current speed. The three objectives are weighted by $R_j \in \mathbb{R}_{>0}, \forall j \in \{1,2,3\}$.

B. Stochastic Predictive Safety Filter

RL cannot provide guaranteed safety, therefore PSF is used to filter the probably unsafe input generated by the RL controller. The formulation of a nominal PSF is as follows:

$$\min_{u^e_{i|k}, \lambda_{i|k}, \gamma_{i|k}} \quad \|u^L(k) - u^e_{0|k}\|^2_{P_1} + \sum_{i=0}^{N-1} \|\Delta u^e_{i|k}\|^2_{P_2} \tag{12a}$$

$$\text{s.t. } x^e_{i+1|k} = x^e_{i|k} + f(x^e_{i|k}, u^e_{i|k}) T_s, \forall i \in \mathbb{N}_{[0,N-1]}, \tag{12b}$$

$$x^e_{i|k} \in \mathbb{X}^e, u^e_{i|k} \in \mathbb{U}^e, x^e_{0|k} = x^e(k), \tag{12c}$$

$$\lambda_{i|k} \succeq 0, \gamma_{i|k} \succeq 0, \tag{12d}$$

$$-{b^e_{i|k}}^\top \lambda_{i|k} - {b^o_{i|k}}^\top \gamma_{i|k} \geq d_{\min}, \tag{12e}$$

$${A^e_{i|k}}^\top \lambda_{i|k} + A^\top \gamma_{i|k} = \mathbf{0}, \tag{12f}$$

$$\|A^\top \gamma_{i|k}\| \leq 1, \tag{12g}$$

where $u^L(k)$ is the control of EV generated by RL controller at current time k. $\Delta u^e_{i|k} = u^e_{i|k} - u^e_{i-1|k}, \forall i \in \mathbb{N}_{[1,N-1]}$ and $\Delta u^e_{0|k} = u^e_{0|k} - u^e_{0|k-1}$ are the rate of change in the control of the EV. The objectives are weighted by $P_j \in \mathbb{R}_{>0}, \forall j \in \{1,2\}$, and $P_1 \gg P_2$. The velocity of OV is assumed to be constant during the entire prediction horizon. In this context, $b^o_{i|k}$ in the collision avoidance constraint over the prediction horizon is predefined based on (2) and (4). To ensure that all safety constraints, such as collision avoidance and physical limits, are satisfied in the presence of uncertainties from human driver behavior, we propose an SPSF framework below for robust safety guarantees.

To capture the uncertainty of human driver behavior, we make the following assumption to model OV's acceleration.

Assumption 4: The acceleration of the OV follows $a^o \sim \mathbb{P}(0, \sigma_a^2)$, where the distribution $\mathbb{P}$ is unknown and σ_a^2 is known from neutralized driving data specific to overtaking scenarios. The acceleration at each time step $a^o_{i|k}$ is independent and identically distributed (i.i.d.).

Due to the unknown distribution of a^o, the optimization problem should be solved considering the worst case, as reflected in $b^o_{i|k}$. Then, the following distributionally robust chance constraint is imposed in place of (12e)

$$\inf_{\mathbb{P} \in \mathcal{P}} \Pr_{[\mathbb{P}]} \{(12e)\} \geq 1 - \epsilon, \tag{13}$$

where $\epsilon \ll 1$ is the risk level, $\mathcal{P}$ is the ambiguity set of the uncertainty. In order to turn the chance constraint (13) into a computationally tractable form, we first write system (2) in a discrete-time form

$$x^o_{k+1} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} x^o_k + \begin{bmatrix} 0 \\ T_s \end{bmatrix} u^o_k = A_d x^o_k + B_d u^o_k, \tag{14}$$

and OV's position can be decomposed by

$$p^o_k = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} x^o_k + \begin{bmatrix} 0 \\ p^o_y \end{bmatrix} = C x^o_k + d. \tag{15}$$

Let $\mathbf{x}^o = \left[{x^o_{0|k}}^\top, {x^o_{1|k}}^\top, \cdots, {x^o_{N|k}}^\top \right]^\top$ and $\mathbf{u}^o = \left[a^o_{0|k}, a^o_{1|k}, \cdots, a^o_{N-1|k} \right]^\top$. Then, (14) can be written in a compact form

$$\mathbf{x}^o = \mathbf{A}_d x^o_{0|k} + \mathbf{B}_d \mathbf{u}^o, \tag{16}$$

where

$$\mathbf{A}_d = \begin{bmatrix} I_2 \\ A_d \\ {A_d}^2 \\ \vdots \\ {A_d}^N \end{bmatrix}, \mathbf{B}_d = \begin{bmatrix} \mathbf{0}_{2\times1} & \mathbf{0}_{2\times1} & \cdots & \mathbf{0}_{2\times1} \\ B_d & \mathbf{0}_{2\times1} & \cdots & \mathbf{0}_{2\times1} \\ AdB_d & B_d & \cdots & \mathbf{0}_{2\times1} \\ \vdots & \vdots & \ddots & \vdots \\ {A_d}^{N-1} B_d & {A_d}^{N-2} B_d & \cdots & B_d \end{bmatrix}.$$

Define the selection matrix as

$$S_i = \begin{bmatrix} \mathbf{0}_{2\times2i} & I_2 & \mathbf{0}_{2\times2(N-i)} \end{bmatrix}, \forall i \in \mathbb{N}_{[0,N]}, \tag{17}$$

such that $S_i \mathbf{x}^o = x^o_{i|k}$. By substituting (15), (16) and (17) into constraint (12e), we obtain

$$\begin{bmatrix} {\mathbf{u}^o}^\top & 1 \end{bmatrix} \begin{bmatrix} M_{i|k} \\ N_{i|k} \end{bmatrix} \leq 0 \Rightarrow \rho^\top \eta \leq 0, \tag{18}$$

where $M_{i|k} = {\mathbf{B}_d}^\top {S_i}^\top C^\top A^\top \gamma_{i|k}$ and $N_{i|k} = {b^e_{i|k}}^\top \lambda_{i|k} + b^\top \gamma_{i|k} + {x^o_{0|k}}^\top {\mathbf{A}_d}^\top {S_i}^\top C^\top A^\top \gamma_{i|k} + d^\top A^\top \gamma_{i|k} + d_{\min}$. Denote the mean vector and covariance matrix of $\mathbf{u}^o$ as $\mu = \mathbf{0}_{N\times1}$ and $\Sigma = \sigma_a^2 I_N$. Then,

$$\mathbf{E}(\rho) = \begin{bmatrix} \mu \\ 1 \end{bmatrix} = \boldsymbol{\mu}, \mathbf{Cov}(\rho) = \begin{bmatrix} \Sigma & \mathbf{0}_{N\times1} \\ \mathbf{0}_{1\times N} & 0 \end{bmatrix} = \boldsymbol{\Sigma}. \tag{19}$$

According to results in [23], constraint (13) is equivalent to the following second-order cone constraint

$$\sqrt{\frac{1-\epsilon}{\epsilon}} \left(\eta^\top \boldsymbol{\Sigma} \eta \right)^{1/2} + \boldsymbol{\mu}^\top \eta \leq 0. \tag{20}$$

The proposed control scheme is summarized in Algorithm 1.

IV. Numerical results

In this section, the proposed control scheme is evaluated in a gym environment to demonstrate its safety assurance and advantages compared to the control mechanism using an IDM-based OV model.

Algorithm 1 Safe Learning-Based Controller for Overtaking

Offline RL training
Initialize N, T_s, σ_a^2 and ϵ
Online
while $k \geq 0$ **do**
  Measure the states of the EV (x_k^e) and the OV (x_k^o)
  Generate u_k^L using the TD3 model subject to game-based environment given in (10) and (11)
  Construct (20) based on x_k^e and x_k^o
  if u^L satisfies safety constraints **then**
    Set the control input $u^S = u^L$
  else
    Modify u^L to a safe control u^S via the SPSF
  end if
  Apply u^S to the EV
end while

A. Simulation Settings

The simulation is conducted in `highway-env` [24], and the environment is set as shown in Fig. 1. It is worth-noting that the maximum speed limit $v_{\max}$ and the road boundaries are naturally enforced by the environment, therefore not repetitively addressed in the SPSF. The RL controller is trained by using the TD3 algorithm in `stable-baselines3` [25]. The learning rate is 0.0001, the buffer size is 10000, and the batch size is 64. γ of TD3 is 0.99 and τ is 0.001. An Ornstein-Uhlenbeck noise with variance 0.1 is added to action for better exploration, and the variance is reduced by 1e-5 at each time step after 80000 steps. The weights in the reward function are set as $\{\alpha_1, \alpha_2, \alpha_3\} = \{1, 1, 4\}$ and $\{\omega_1, \omega_2, \omega_3\} = \{0.4, 0.2, 1\}$. For stable training performance, the states and actions of the agent are normalized to $[-1, 1]$, and the reward is normalized to $[0, 1]$. The sampling time T_s is set to 0.1 s. The prediction horizon for FHOCP1 and FHOCP2 is $N = 10$, and for SPSF the prediction horizon is $N = 20$. The weights of FHOCP1 are set to $Q_j = \{1, 1, 1, 1, 2\}$. The weights for FHOCP2 are set as $R_j = \{20, 1, 1\}$. For SPSF, $P_1 = 100$ and $P_2 = 1$. The risk level $\epsilon = 0.01$. Variance $\sigma_a^2 = 0.01$. All optimization problems are solved by `CasADi` [26]. The lane width $W = 4$ m and the safety gap is set to $d_{\min} = 2$ m. The initial speed of EV is set to 25 m/s. For OV, the initial speed is randomly selected from $[21, 24]$ m/s. Rest of vehicle parameters are listed in Table I.

TABLE I: Vehicle Parameters

Symbol	Value	Description
l / w	5 / 2 m	Vehicle length/width
l_r / l_f	2 / 2 m	Wheel base length
$[v_{\min}, v_{\max}]$	$[20, 30]$ m/s	Speed range
$[\psi_{\min}, \psi_{\max}]$	$[-\pi/4, \pi/4]$ rad	Orientation angle range
$[a_{\min}, a_{\max}]$	$[-5, 5]\, m/s^2$	Acceleration range
$[\delta_{\min}, \delta_{\max}]$	$[-\pi/8, \pi/8]$ rad	Steering angle range

B. Simulation Results

Fig. 4 shows the convergence of the episode reward mean during training, reached after 100000 steps.

By incorporating both IDM and the game-based OV model in RL, it can be observed from Fig. 5 that the Stackelberg game-based OV exhibits more realistic behavior than the IDM-based OV. Specifically, in Fig. 5a, the OV maintains a constant speed and shows no reaction before the EV cuts in. After the EV merges back into the initial lane, the OV abruptly decelerates to give way to the EV, which is both unrealistic and unsafe in practice. In contrast, Fig. 5b demonstrates that the game-based OV initiates an interaction with the EV when it detects the intention of the EV roughly while the EV is ahead of the OV by half a vehicle length (e.g., through the turn signal of the EV), yielding a more reasonable and mild deceleration.

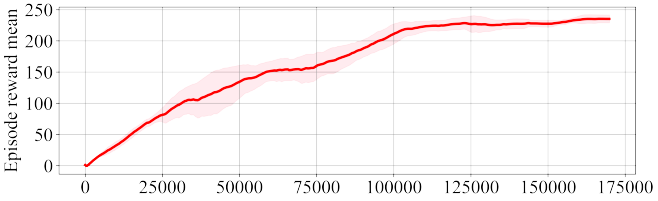
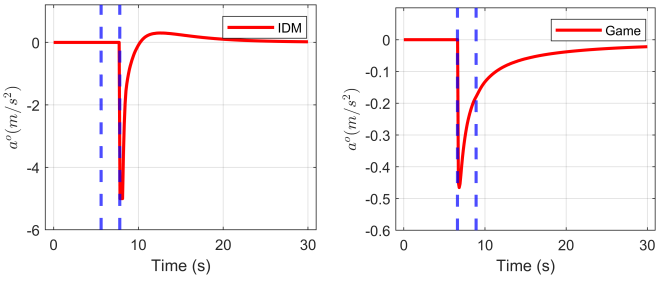


Fig. 4: The episode reward mean during the training process. The total time steps are set to 170000.



(a) (b)

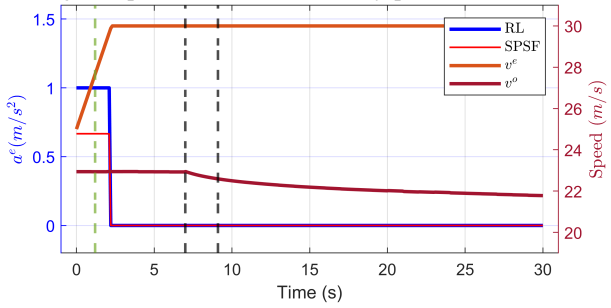
Fig. 5: Acceleration of OV during an overtaking process. (a) IDM-based OV driving behavior during overtaking; (b) Stackelberg game-based OV driving behavior during overtaking (in a noise-free setting). The left blue dashed line denotes the moment when EV is ahead of OV by half a vehicle length and starts merging back. The right blue dashed line denotes the end of overtaking maneuver.

Finally, taking into account human behavior uncertainties, the effectiveness of the proposed SPSF is shown in Fig. 6. When the action generated by the upper RL controller is verified as safe, the SPSF outputs the same action. When an unsafe action occurs, the SPSF modifies it to enforce safety constraints, as can be seen in Fig. 6c. For instance, the steering angle, when it exceeds the allowable limits at about $t = 8$ s, is robustly constrained to remain within the maximum feasible range.

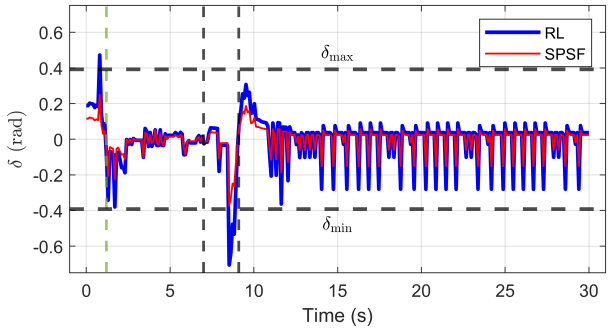
V. Conclusions

This work proposes a safe control framework for autonomous overtaking when uncertain human driver behaviors are presented. In the upper RL training process, a Stackelberg game-based OV model is incorporated in the environment to represent the interaction between EV and OV during the overtaking process. Moreover, for a trained agent, a SPSF which adopts a distributionally robust chance constraint is cascaded to the RL model to enforce collision avoidance and other safety-related constraints under uncertain human driver behaviors. The effectiveness of the proposed scheme is shown via simulation case studies in a gym environment. Future research work will consider more complex environments

(a) The vehicle trajectories during the overtaking process. The EV is in green and the OV is in yellow. The semi-transparent rectangles represent the vehicle history positions.



(b) Left: the acceleration of the EV; right: the speed of the EV and the OV.



(c) The steering angle of the EV.

Fig. 6: Vehicle trajectories, speeds and control signals during the overtaking. The three vertical dashed lines in (b) and (c) indicate the time instants when the EV switches to the overtaking lane, when it start merging back, and when it merges back into the initial lane, respectively.

including multiple OVs and lanes. Adaptive weights of the OV cost in the game environment will also be investigated to capture more realistic human driver behaviors.

REFERENCES

[1] S. S. Lodhi, N. Kumar, and P. K. Pandey, "Autonomous vehicular overtaking maneuver: A survey and taxonomy," *Vehicular Communications*, vol. 42, p. 100623, 2023.

[2] Y. Gao, F. J. Jiang, L. Xie, and K. H. Johansson, "Risk-aware optimal control for automated overtaking with safety guarantees," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 4, pp. 1460–1472, 2021.

[3] S. Yu, B. Chen, I. M. Jaimoukha, and S. A. Evangelou, "Game-theoretic model predictive control for safety-assured autonomous vehicle overtaking in mixed-autonomy environment," in *European Control Conference (ECC)*, 2024.

[4] X. Gong, S. Liang, B. Wang, and W. Zhang, "Game theory-based decision-making and iterative predictive lateral control for cooperative obstacle avoidance of guided vehicle platoon," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7051–7066, 2023.

[5] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 712–733, 2020.

[6] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-end autonomous driving: Challenges and frontiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[7] X. Hu, Y. Liu, B. Tang, J. Yan, and L. Chen, "Learning dynamic graph for overtaking strategy in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 11 921–11 933, 2023.

[8] A. Norouzi, H. Heidarifar, H. Borhan, M. Shahbakhti, and C. R. Koch, "Integrating machine learning and model predictive control for automotive applications: A review and future directions," *Engineering Applications of Artificial Intelligence*, vol. 120, p. 105878, 2023.

[9] M. Kaushik, V. Prasad, K. M. Krishna, and B. Ravindran, "Overtaking maneuvers in simulated highway driving using deep reinforcement learning," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1885–1890.

[10] Z. Gu, L. Gao, H. Ma, S. E. Li, S. Zheng, W. Jing, and J. Chen, "Safe-state enhancement method for autonomous driving via direct hierarchical reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 9, pp. 9966–9983, 2023.

[11] M. Al-Sharman, R. Dempster, M. A. Daoud, M. Nasr, D. Rayside, and W. Melek, "Self-learned autonomous driving at unsignalized intersections: A hierarchical reinforced learning approach for feasible decision-making," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 12 345–12 356, 2023.

[12] J. Li, C. Yu, Z. Shen, Z. Su, and W. Ma, "A survey on urban traffic control under mixed traffic environment with connected automated vehicles," *Transportation research part C: emerging technologies*, vol. 154, p. 104258, 2023.

[13] X. Chen, J. Wei, X. Ren, K. H. Johansson, and X. Wang, "Automatic overtaking on two-way roads with vehicle interactions based on proximal policy optimization," in *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2021, pp. 1057–1064.

[14] K. Sama, Y. Morales, H. Liu, N. Akai, A. Carballo, E. Takeuchi, and K. Takeda, "Extracting human-like driving behaviors from expert driver data using deep learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 9315–9329, 2020.

[15] Z. Huang, J. Wu, and C. Lv, "Driving behavior modeling using naturalistic human driving data with inverse reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 10 239–10 251, 2021.

[16] X. Di and R. Shi, "A survey on autonomous vehicle control in the era of mixed-autonomy: From physics-based to AI-guided driving policy learning," *Transportation research part C: emerging technologies*, vol. 125, p. 103008, 2021.

[17] Q. Zhang, R. Langari, H. E. Tseng, D. Filev, S. Szwabowski, and S. Coskun, "A game theoretic model predictive controller with aggressiveness estimation for mandatory lane change," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 1, pp. 75–89, 2019.

[18] M. Chen, J. Jiang, and B. Chen, "A game-based optimal and safe lane change control of autonomous vehicles in mixed traffic scenario," in *2024 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2024, pp. 52–57.

[19] K. P. Wabersich and M. N. Zeilinger, "A predictive safety filter for learning-based control of constrained nonlinear dynamical systems," *Automatica*, vol. 129, p. 109597, 2021.

[20] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 1094–1099.

[21] X. Zhang, A. Liniger, and F. Borrelli, "Optimization-based collision avoidance," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 972–983, 2020.

[22] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1587–1596.

[23] G. C. Calafiore and L. E. Ghaoui, "On distributionally robust chance-constrained linear programs," *Journal of Optimization Theory and Applications*, vol. 130, pp. 1–22, 2006.

[24] E. Leurent, "An environment for autonomous driving decision-making," https://github.com/eleurent/highway-env, 2018.

[25] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021.

[26] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.