MeUAL: Model-enhanced Uncertainty-aware Safe Reinforcement Learning for Safety-critical Autonomous Highway Overtaking

Sunan Zhang, Bingbing Li, Boli Chen, *Senior Member, IEEE*, Bo Hu, Chen Sun, and Weichao Zhuang, *Member, IEEE*

***Abstract*—Decision-making and control are the core functionalities of high-level autonomous driving systems. Existing mainstream research, including modular and end-to-end paradigms, typically employ conservative strategies that compromise driving efficiency. However, driving efficiency constitutes a critical constraint on the transition of autonomous vehicles from mere operability to practical utility. Autonomous overtaking systems serve as a typical means to improve driving efficiency. Nevertheless, in stochastic and uncertain traffic scenarios, achieving safe and efficient continuous autonomous overtaking remains a significant challenge. In this context, this paper proposes a decision-making and control framework based on MeUAL to achieve the optimal trade-off between overtaking risk and efficiency. First, at the decision-making layer, a safe reinforcement learning method based on Uncertainty-aware Augmented Lagrangian (UAL) is developed to provide global overtaking guidance. Subsequently, the motion planning and control layer based on Model Predictive Control (MPC) closely tracks the UAL-generated guidance, while preserving the safety and constraint guarantees inherent to traditional MPC. Finally, a Policy Switching Mechanism (PSM) triggered by the safety epistemic uncertainty threshold is designed for the MeUAL-driven autonomous overtaking system. Experimental results demonstrate that MeUAL outperforms baseline algorithms with respect to reward-cost balance, sample efficiency, and learning stability. Moreover, in various test scenarios that are distinct from the training distribution, MeUAL-PSM exhibits strong robustness and interpretable overtaking maneuvers through flexible policy switching.**

***Index Terms*—Decision-making and control, autonomous overtaking, safe reinforcement learning, policy switching mechanism.**

I. Introduction

AUTONOMOUS Driving (AD) technology enables vehicles to independently execute driving tasks, thereby providing safer, more efficient, and more convenient transportation solutions [1]. In advanced AD systems, the decision-making and control modules perform a central function analogous to the human brain, converting environmental perception data into actionable commands [2]. Consequently, their performance directly determines the capability of the AD system to handle dynamic driving environments [3].

Traditional decision-making and control systems typically adopt a modular design, decomposing the system into a series of functionally independent modules, such as prediction, decision-making, trajectory planning, and control [4]. This solution has been widely applied in the industry due to its strong interpretability. However, the dependence of each module on extensive manual design, combined with their vulnerability to the long-tail effect, renders it difficult to cover all potential driving scenarios. To overcome this limitation, both academia and industry have increasingly explored more adaptive data-driven approaches, with end-to-end decision-making and control methods emerging as a central trend in AD research [5], [6].

The end-to-end paradigm directly maps perception data to the desired control actions through a deep neural network-based policy, which can be divided into two categories: Imitation Learning (IL) and Reinforcement Learning (RL). IL methods are widely used in autonomous driving tasks due to their effectiveness in replicating human driving behavior [7]. However, these methods typically depend on diverse and high-quality driving data, but data from rare scenarios are often scarce and challenging to obtain, resulting in limited performance under such conditions [8]. In contrast, RL methods gradually optimize policies through trial and error within high-fidelity simulation environments or real-world scenarios, thereby enabling autonomous learning in previously unknown situations [9]. Early applications of RL in AD typically employed deep Q-networks or actor-critic methods to learn policies in discrete or continuous action spaces. These studies have achieved remarkable results in tasks such as ramp merging [10], highway lane changing [11], [12], and unprotected left turns at unsignalized intersections [13]. However, even well-trained RL agents face significant challenges in ensuring policy safety due to the absence of safety constraints. Consequently, Safe RL, which integrates safety constraints into policy learning, has emerged as a paradigm in RL applications.

The safe RL method based on constrained policy optimization utilizes constrained optimization techniques to jointly optimize safety and performance, where performance is encoded through a reward function and safety is expressed

This work was supported in part by the National Natural Science Foundation of China under Grant 52441204, 52172383, and in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX24_0376. *(Corresponding author: Weichao Zhuang.)*

Sunan Zhang, Bingbing Li, Weichao Zhuang are with the School of Mechanical Engineering, Southeast University, Nanjing 211189, China (e-mail: sunzhang@seu.edu.cn; bingbli@seu.edu.cn; wezhuang@seu.edu.cn).

Boli Chen is with the Department of Electronic and Electrical Engineering, University College London, WC1E 6BT London, U.K. (e-mail: boli.chen@ucl.ac.uk).

Bo Hu is with Key Laboratory of Advanced Manufacturing Technology for Automobile Parts, Ministry of Education, Chongqing University of Technology, Chongqing 400054, China (e-mail: b.hu@cqut.edu.cn).

Chen Sun is with Department of Data and Systems Engineering, University of Hong Kong, Pok Fu Lam, Hong Kong, China (e-mail: c87sun@hku.hk).

as a constraint [14]. Achiam et al. [15] first proposed a Constrained Policy Optimization (CPO) method that combines Trust Region Policy Optimization (TRPO) [16] with safety constraints. This method monotonically improves the policy and ensures the satisfaction of constraints in each policy update. The Projection-based Constrained Policy Optimization (PCPO) [17], derived from CPO, adopts a two-step approach: first, the policy is optimized using the TRPO method, and then the policy is projected into the feasible region to satisfy safety constraints. Although the aforementioned safety-enhanced TRPO methods have theoretical guarantees in terms of safety, they often suffer from high computational overhead and substantial sample interaction requirements. Another type of Lagrangian-based method converts safety constraints into adjustable penalty terms and embeds them into the optimization objective by constructing a primal-dual optimization framework. This method adopts an alternating update mechanism for the policy network and Lagrangian multipliers, theoretically converging to the optimal policy that satisfies the safety constraints [14]. However, previous studies have pointed out that in the primal-dual optimization framework, the instability of Lagrange multiplier updates may cause policy oscillations, thus affecting the safety of the final policy [18], [19]. To suppress the oscillations of the Lagrange multiplier, Stooke et al. [20] used PID control to stabilize the update process of the Lagrange multiplier. Additionally, Liu et al. [21] decomposed the safe RL problem into two stages: convex optimization and supervised learning from the perspective of probabilistic inference, effectively improving the stability of policy updates. In summary, safe RL methods based on constrained policy optimization perform well in balancing safety and performance. However, these methods typically incorporate safety constraints into the objective function, leading to only a soft enforcement of safety constraints, rather than ensuring strict adherence to safety [22].

To ensure the safety of RL policies in real-world deployments, a growing body of research has focused on restricting the exploration behavior of RL policies to safe regions. The first category of approaches involves safety filtering, which guarantees constraint satisfaction by modifying control outputs in real time. Methods such as Control Barrier Function (CBF)-based quadratic programs (QP) [23] and Hamilton–Jacobi (HJ) reachability filters [24] act as correction layers after the RL controller, making minimal interventions to enforce safety constraints. However, because such safety filters operate independently of the RL controller, they often lead to myopic and suboptimal decisions. The second category of methods is online optimization [25], [26], which typically naturally incorporates hard safety constraints while following the high-level guidance provided by the RL controller, thereby enhancing the overall safety of the system and improving the sample efficiency of RL policy learning. Nevertheless, when RL policies encounter Out-Of-Distribution (OOD) scenarios relative to the training data, overall system performance may degrade substantially or even fail completely. To address this gap, the third category of methods is the hybrid framework of RL and alternative policies [27], [28]. These methods first quantify the epistemic uncertainty of the RL policy and switch to an alternative policy when the uncertainty is high, thereby compensating for the vulnerability of RL policies in handling OOD scenarios. Although the aforementioned methods have driven significant progress in RL applications, they all focus on traditional RL approaches and do not encompass constraint-based policy optimization within the context of Safe RL. Furthermore, few prior studies have applied quantified uncertainty to Safe RL-based AD systems.

To this end, this paper proposes a Model-enhanced Uncertainty-aware Augmented Lagrangian (MeUAL) approach for decision-making and control in autonomous highway overtaking scenarios. This approach incorporates epistemic uncertainty into the update of a safe RL policy based on the augmented Lagrangian through deep integration techniques. After training convergence, epistemic uncertainty is further utilized as a safety threshold for detecting OOD scenarios, thereby enabling the design of a policy-switching mechanism. The main contributions of this research are summarized as follows:

1) **System Framework Design:** A model-augmented safe RL framework is introduced, which integrates Lagrangian-based safe RL decision-making with MPC-based motion control to jointly handle both soft and hard constraints in autonomous highway overtaking scenarios.
2) **Constrained Policy Optimization:** An uncertainty-aware augmented Lagrangian method is proposed, which integrates cost Q-function estimation with augmented Lagrangian optimization to enhance both constraint satisfaction and the learning stability of the policy.
3) **Policy Switching Mechanism:** A policy-switching mechanism triggered by epistemic uncertainty or MPC infeasibility is developed, enabling robust and interpretable fallback to a conservative backup strategy under challenging scenarios.

The remainder of the paper is organized as follows. Section II introduces the proposed decision-making and control framework. Section III describes the implementation details for its application to highway autonomous driving tasks. Experimental results and discussions are presented in Section IV. Finally, the conclusion is drawn in Section V.

II. Decision-Making and Control Framework Based on MeUAL

A. Framework

As illustrated in Fig. 1, this paper proposes a decision-making and control framework based on MeUAL. In this framework, the UAL policy is responsible for generating global decision guidance with uncertainty awareness at the decision layer, while the MPC optimizer translates the global guidance into optimal control instructions that satisfy actual constraints at the motion planning and control layer. Together, these two components form the MeUAL policy.

Notably, the motivation for introducing MPC lies in addressing key concerns regarding the application of safe RL in AD. Specifically, AD tasks entail numerous hard physical constraints—such as speed limits, road boundaries, and

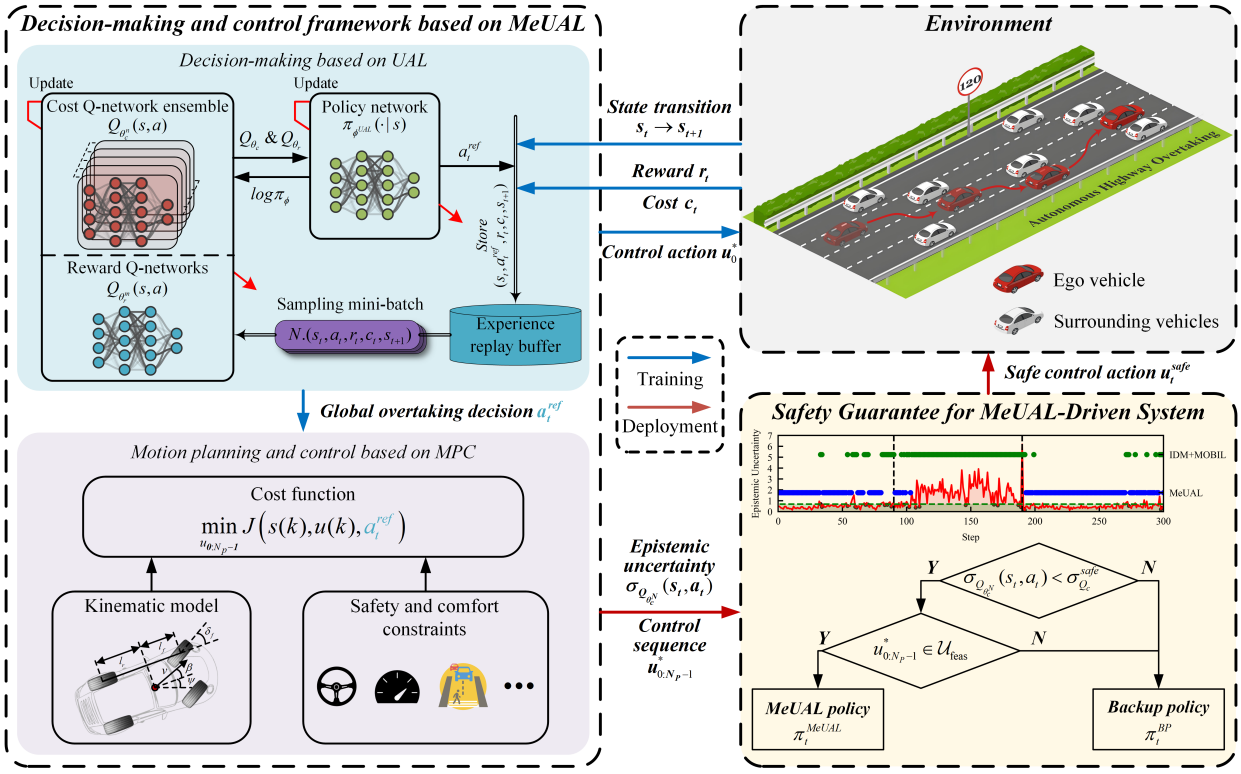


Fig. 1. Decision-making and control framework based on MeUAL.

acceleration/steering limits with rate bounds. Handling these constraints indirectly via reward or cost penalties complicates multi-objective optimization and risks failing to satisfy safety-critical constraints. By contrast, MPC enforces multiple hard constraints explicitly, thereby reducing the state space that safe RL agents need to explore, i.e., mapping the original high-dimensional space (often physically infeasible) into a feasible subspace. In turn, this enables safe RL to concentrate on tradeoffs between safety and performance rather than struggling to comply with multiple strict constraints.

Specifically, the environment first provides the state s_t of the ego vehicle and surrounding vehicles to the decision-making and control framework. Subsequently, the decision-making layer samples continuous decision actions $a_t^{ref} \sim \pi_{\phi^{UAL}}$ from the UAL policy. The decision action contains the reference speed and the reference lateral position, that is, $a_t^{ref} = [v_t^{ref}, y_t^{ref}]^T$. Next, the motion planning and control layer uses the global guidance as the input of the MPC optimizer and solves the formulated constrained optimization problem to generate feasible optimal control actions $u_0^* = [u^a, u^{\delta_f}]^T$, where u^a represents the vehicle acceleration and u^{δ_f} represents the front wheel steering angle, considering the realistic constraints related to vehicle kinematics, traffic rules, and ride comfort. Then, the UAL agent obtains the immediate reward r_t and the immediate cost c_t from the environment through the defined reward function and cost function, and the environment transfers to the next state s_{t+1}. During the interaction between the MeUAL policy and the environment, the tuples $(s_t, a_t^{ref}, r_t, c_t, s_{t+1})$ generated are stored in the experience replay buffer for the UAL agent's learning.

In addition, this paper designs a policy switching mechanism based on a safety epistemic uncertainty threshold trigger for the MeUAL-driven automatic overtaking system. When the MeUAL policy cannot provide a feasible action, the backup policy will take over to ensure that the vehicle can still maintain the performance lower limit in unfamiliar scenarios, thereby ensuring the reliability of vehicle operation.

B. Decision-Making Based on UAL

1) SAC-Lag-Based Safe RL Algorithm: SAC-Lag is a safety-constrained RL method that introduces Lagrangian relaxation technology based on SAC. The goal of SAC-Lag is to find an optimal policy that maximizes the expected cumulative reward while satisfying safety constraints and entropy constraints, as follows:

$$\pi^* = \arg\max \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[Q_r^\pi(s,a)]$$
$$s.t. \begin{cases} \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[Q_c^\pi(s,a)] \le d \\ \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[-\log(\pi(a|s))] \ge \mathcal{H}_0 \end{cases} \quad (1)$$

where $\mathcal{H}_0$ is the minimum entropy threshold, which limits the minimum randomness of the policy. It should be noted that we have a constraint on the expected cost Q value on the trajectory, and a local constraint on the policy entropy at each time step.

We train the reward Q-network $Q_{\theta_r}(s,a)$ and the cost Q-network $Q_{\theta_c}(s,a)$ to estimate the state-action Q-values of reward and cost, and train a policy network π_ϕ to generate the action distribution under a given state. To effectively balance reward, safety, and exploration, we employ a standard Lagrangian approach—primal-dual policy optimization. By introducing the Lagrange multiplier λ and the entropy weight

α, the constrained optimization problem (1) is transformed into an unconstrained optimization problem, and the following minimum-maximum problem is iteratively solved:

$$\min_{\alpha\geq 0}\min_{\lambda\geq 0}\max_{\pi}\mathbb{E}_{s,a}\Big[-\alpha\left(\log\left(\pi_\phi(a|s)\right)+\mathcal{H}_0\right)+Q_{\theta_r}(s,a) -\lambda\left(Q_{\theta_c}(s,a)-d\right)\Big] \tag{2}$$

where $\mathbb{E}_{s,a}[\cdots]:=\mathbb{E}_{s\sim\rho_\pi,a\sim\pi(\cdot|s)}[\cdots]$, θ_r, θ_c and ϕ represent the parameters of the reward Q-network $Q_{\theta_r}(s,a)$, the cost Q-network $Q_{\theta_c}(s,a)$ and the policy network π_ϕ, respectively.

2) Uncertainty-aware Cost Q-value Evaluation: In the iterative update process of (2), when $Q_{\theta_c}(s,a)>d$, the optimal Lagrange multiplier $\lambda^*=+\infty$, leading to policy updates that prioritize minimizing the cost Q-value estimate $Q_{\theta_c}(s,a)$. However, $Q_{\theta_c}(s,a)$ actually contains random noise caused by the function approximator. We assume that $Q_{\theta_c}(s,a)$ has zero-mean noise ϵ_{Q_c}:

$$Q_{\theta_c}(s,a)=\tilde{Q}_c(s,a)+\epsilon_{Q_c} \tag{3}$$

where $\tilde{Q}_c(s,a)$ represents the true cost Q value.

Previous studies have shown that under minimization, the presence of noise can lead to an underestimate of the cost Q value because it cannot preserve its zero-mean property [29], i.e.,

$$\mathbb{E}_{\epsilon_{Q_c}}[\min_{a_t}Q_{\theta_c}(s,a)]\leq\min_{a_t}\tilde{Q}_c(s,a) \tag{4}$$

Although this underestimation bias may be small at each update, it can be further accumulated through temporal difference learning, which may lead to larger underestimation bias and suboptimal policy updates [30].

To address the issue of cost Q-value underestimation, we propose an uncertainty-aware cost Q-value estimation method. Inspired by the work [31], we use a deep ensemble approach to quantify the epistemic uncertainty of the cost Q-network. Specifically, a single cost Q-network is expanded into a cost Q-network ensemble, as follows:

$$\left\{Q_{\theta_c^n}\right\}_{n=1}^N=\left\{Q_{\theta_c^n},n=1,2,\cdots,N\right\} \tag{5}$$

where N represents the number of cost Q-networks in the ensemble network, and in this study $N=6$. These networks have the same architecture and are trained on the same dataset, but with different initial weights. Each network estimates the cost Q value independently and in parallel, so the mean and standard deviation of the ensemble cost Q-network can be calculated using the outputs of the N cost Q-networks:

$$\mu_{Q_{\theta_c^N}}(s,a)=\frac{1}{N}\sum_{n=1}^{N}Q_{\theta_c^n}(s,a) \tag{6}$$

$$\sigma_{Q_{\theta_c^N}}(s,a)=\sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(Q_{\theta_c^n}(s,a)-\mu_{Q_{\theta_c^N}}(s,a)\right)^2} \tag{7}$$

The standard deviation $\sigma_{Q_{\theta_C^N}}$ reflects the estimation differences of different cost Q-networks for the same state-action pair, quantifying the epistemic uncertainty of the cost Q-network ensemble. When the cost Q-network ensemble lacks confidence in a particular state-action pair, $\sigma_{Q_{\theta_C^N}}$ will increase. Based on this, the uncertainty-aware cost Q value estimate can be defined as:

$$Q^U_{\theta_c^N}(s,a)=\mu_{Q_{\theta_c^N}}(s,a)+k\sigma_{Q_{\theta_c^N}}(s,a) \tag{8}$$

where k is a hyperparameter used to adjust the impact of epistemic uncertainty on the cost Q-value estimation. By adding $\mu_{Q_{\theta_C^N}}$ to the mean $k\sigma_{Q_{\theta_c^N}}$, the underestimation bias of the cost Q-value can be effectively compensated, thereby reducing potential safety risks caused by this bias.

3) Local Policy Convexification Using Augmented Lagrangian: When the estimated cost Q-value exceeds the threshold $Q_{\theta_c}(s,a)>d$, but the true cost Q-value is actually below the threshold $\tilde{Q}_c(s,a)<d$, the Lagrange multiplier λ will be updated in the direction completely opposite to the true direction, which will become a misleading penalty weight during the optimization of the policy π. Thus, the Lagrangian method relies on accurate cost Q-value estimates, making their application in off-policy settings challenging: it requires backpropagating gradients from multiple Q-value functions to the policy network, potentially causing oscillations in policy learning. To stabilize policy learning, we modify the original optimization objective (2) of SAC-Lag by applying the Augmented Lagrangian Method (ALM) as follows:

$$\min_{\alpha\geq 0}\min_{\lambda\geq 0}\max_{\pi}\mathbb{E}_{s,a}\Big[-\alpha\left(\log\left(\pi_\phi(a|s)\right)+\mathcal{H}_0\right)+Q_{\theta_r}(s,a) -\frac{1}{2c}\left(\max\left\{0,\lambda-c\left(d-Q^U_{\theta_c^N}(s,a)\right)\right\}^2-\lambda^2\right)\Big] \tag{9}$$

where $Q_{\theta_c}(s,a)$ is replaced by the uncertainty-aware cost Q value $Q^U_{\theta_c^N}(s,a)$.

To clarify the update mechanism of Lagrange multiplier λ and policy π, (9) is expanded as follows:

$$\min_{\alpha\geq 0}\min_{\lambda\geq 0}\max_{\pi}\mathbb{E}_{s,a} \begin{cases} -\alpha\left(\log\left(\pi_\phi(a|s)\right)+\mathcal{H}_0\right)+Q_{\theta_r}(s,a)-\lambda\left(Q^U_{\theta_c^N}(s,a)-d\right) \\ -\dfrac{c}{2}\left(Q^U_{\theta_c^N}(s,a)-d\right)^2, if\dfrac{\lambda}{c}>d-Q^U_{\theta_c^N}(s,a) \\ -\alpha\left(\log\left(\pi_\phi(a|s)\right)+\mathcal{H}_0\right)+Q_{\theta_r}(s,a)+\dfrac{\lambda^2}{2c}, \text{otherwise} \end{cases} \tag{10}$$

where c is a positive penalty coefficient. Under standard assumptions of inequality constraints, adding the quadratic penalty term $\frac{c}{2}\left(Q^U_{\theta_c^N}(s,a)-d\right)^2$ to the SAC-Lag objective preserves the optimality properties of the original constrained problem (i.e., optimal policy and Lagrangian multipliers remain unchanged) [32]. Additionally, when the original constrained optimization problem inherently possesses a convex structure (e.g., convex constraints and concave objectives), the quadratic penalty term helps maintain this convexity in the augmented Lagrangian formulation, thereby improving numerical stability during training. The inequality $\frac{\lambda}{c}\leq d-Q^U_{\theta_c^N}(s,a)$ indicates that the uncertainty cost Q-value remains below the threshold, thereby excluding the cost Q-value from the optimization objective.

4) Design and Training of Q-value Networks and Policy Networks: Similar to the SAC-Lag algorithm, dual Q-learning and target networks are used to solve the problems of overestimation bias and unstable target values in reward Q-learning. We use two reward Q-networks $Q_{\theta_r^1}$ and $Q_{\theta_r^1}$ with parameters θ_r^1 and θ_r^2 to estimate the reward Q value, and their parameters can be trained by minimizing the Bellman residual:

$$\mathcal{L}_Q(\theta_r^m)=\mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{D}}\left[\frac{1}{2}\left(Q_{\theta_r^m}(s_t,a_t)-y(r_t,s_{t+1})\right)^2\right], \quad m\in\{1,2\} \tag{11}$$

where $\mathcal{D}$ is the experience replay buffer that stores interaction data. The temporal difference (TD) target for the reward $y(r_t, s_{t+1})$ is estimated as follows:

$$\begin{aligned} y(r_t,s_{t+1}) = r_t+\gamma\left(\min_{m=1,2} Q_{\bar{\theta}_r^m}(s_{t+1},\widetilde{a}_{t+1})\right) \\ -\alpha\log\pi_{\phi^{UAL}}(\widetilde{a}_{t+1}|s_{t+1}) \end{aligned} \tag{12}$$

where $\widetilde{a}_{t+1}\sim\pi_{\phi^{UAL}}(\cdot|s_{t+1})$, $\bar{\theta}_r^m$ represents the parameters of the target reward Q-network. Drawing on the idea of dual Q learning, we use the smaller value of the two target reward Q networks $Q_{\bar{\theta}_r^1}$ and $Q_{\bar{\theta}_r^2}$ to estimate the reward Q value of the next state-action pair $(s_{t+1}, \tilde{a}_{t+1})$, thereby effectively avoiding the overestimation of the reward Q value.

The loss function of the cost Q-network ensemble is as follows:

$$\mathcal{L}_Q(\theta_c^n)=\mathbb{E}_{(s_t,a_t,c_t,s_{t+1})\sim\mathcal{D}}\left[\frac{1}{2}\left(Q_{\theta_c^n}(s_t,a_t)-y(c_t,s_{t+1})\right)^2\right], \quad n\in\{1,2,\cdots,N\} \tag{13}$$

where the TD target for the cost $y(c_t, s_{t+1})$ can be estimated as follows:

$$y(c_t,s_{t+1}) = c_t+\gamma Q^U_{\bar{\theta}_c^N}(s_{t+1},\tilde{a}_{t+1}) \tag{14}$$

where $\bar{\theta}_c^N$ is the parameter of the target cost Q-network ensemble. In the early stage of training, due to the lack of sufficient training data, the differences between multiple cost Q-networks are large. Therefore, we use uncertainty-aware cost Q value $Q^U_{\bar{\theta}_c^N}$ to replace the original cost Q value. By considering the uncertainty in the cost Q estimate, $Q^U_{\bar{\theta}_c^N}$ can be used as an approximate upper limit of the cost Q value with high confidence, thereby alleviating the adverse effects of underestimation of the cost Q value.

We use a policy network with parameters ϕ^{UAL} to approximate the policy function of UAL. According to the expanded optimization objective of UAL (10), the loss function of the policy network can be expressed as:

$$\begin{aligned} &\mathcal{L}_\pi(\phi^{UAL}) = \mathbb{E}_{s_t\sim\mathcal{D},\epsilon_t\sim\mathcal{N}} \\ &\begin{cases} \alpha\log\left(\pi_{\phi^{UAL}}(\tilde{a}_t|s_t)\right)-\min_{m=1,2} Q_{\theta_r^m}(s_t,\tilde{a}_t) \\ +\left[\lambda-c\left(d-\frac{Q^U_{\theta_c^N}(s_t,\tilde{a}_{\phi^{UAL},t})}{2}\right)\right], if\frac{\lambda}{c}>d-Q^U_{\theta_c^N}(s_t,\tilde{a}_t) \\ \alpha\log\left(\pi_{\phi^{UAL}}(\tilde{a}_t|s_t)\right)-\min_{m=1,2} Q_{\theta_r^m}(s_t,\tilde{a}_t), otherwise \end{cases} \end{aligned} \tag{15}$$

where $\tilde{a}_t$ is obtained through the reparameterization trick, which helps to reduce the variance of the gradient estimate and is defined as:

$$\tilde{a}_t = f_{\phi^{UAL}}(\epsilon_t;s_t) \tag{16}$$

where $\epsilon_t\in\mathbb{R}^{\dim(\mathcal{A})}$ is an input noise vector sampled from a fixed distribution. Since the policy $\pi_{\phi^{UAL}}(\cdot\mid s_t)$ is a Gaussian distribution, $f_{\phi^{UAL}}(\epsilon_t;s_t)$ can be expressed as:

$$\begin{aligned} f_{\phi^{UAL}}(\epsilon_t;s_t) = \tanh(\mu_{\phi^{UAL}}(s_t)+\sigma_{\phi^{UAL}}(s_t)\odot\epsilon_t), \\ \epsilon_t\sim\mathcal{N}(0,I_{\dim(\mathcal{A})}) \end{aligned} \tag{17}$$

where $\mu_{\phi^{UAL}}(s_t)\in\mathbb{R}^{\dim(\mathcal{A})}$ and $\sigma_{\phi^{UAL}}(s_t)\in\mathbb{R}^{\dim(\mathcal{A})}$ represent the mean and standard deviation of the policy $\pi_{\phi^{UAL}}(\cdot\mid s_t)$, respectively, $\odot$ represents the Hadamard product, ϵ_t follows the standard normal distribution $\mathcal{N}(0, I_{\dim(\mathcal{A})})$, and $\dim(\mathcal{A})$ is the dimension of the action space. The loss function of the enhanced Lagrangian multiplier λ_{AL} and entropy weight α is modified as follows:

$$\mathcal{L}(\lambda_{AL})=\mathbb{E}_{s_t\sim\mathcal{D},a_t\sim\pi_{\phi^{UAL}}(\cdot|S_t)}\left[\lambda_{AL}\left(d-Q^U_{\theta_c^N}(s_t,a_t)\right)\right] \tag{18}$$

$$\mathcal{L}(\alpha)=\mathbb{E}_{s_t\sim\mathcal{D},a_t\sim\pi_{\phi^{UAL}}(\cdot|S_t)}\left[-\alpha\left(\log\left(\pi_{\phi^{UAL}}(a_t|s_t)\right)+\mathcal{H}_0\right)\right] \tag{19}$$

Therefore, when the constraints are violated, i.e., the current policy is unsafe or entropy is insufficient, λ_{AL} and α will be adjusted accordingly.

C. Motion Planning and Control Based on MPC

1) Vehicle Model: The bicycle kinematic model is used to simulate the ego vehicle's motion, as shown in Fig. 2, and the discrete-time model can be described as:

$$\begin{aligned} s_e(k+1) &= f_{ego}\left(s_e(k),u(k)\right) \\ &= \begin{bmatrix} x(k)+v(k)\cos\left(\psi(k)+\beta(k)\right)\Delta t \\ y(k)+v(k)\sin\left(\psi(k)+\beta(k)\right)\Delta t \\ v(k)+u^a(k)\Delta t \\ \psi(k)+\frac{v(k)}{l_r}\sin\beta(k)\Delta t \end{bmatrix} \end{aligned} \tag{20}$$

where the vehicle state is $s_e=[x,y,v,\psi]^T$, x and y represent the longitudinal and lateral positions of the vehicle, v is the vehicle speed, and ψ is the yaw angle. u^a denotes the vehicle acceleration, and β is the sideslip angle at the center of mass, which relates to the front-wheel steering angle u^{δ_f} as:

$$\beta(k)=\arctan\left(\frac{l_r}{l_f+l_r}\tan u^{\delta_f}(k)\right) \tag{21}$$

where l_f and l_r denote the distances from the center of mass to the front axle and rear axle, respectively. u is the control action vector, $u=\left[u^a,u^{\delta_f}\right]^T$.

2) Cost Function: To incentivize the ego vehicle to closely track the continuous decision actions $a_t^{ref}=[v_t^{ref},y_t^{ref}]^T$ generated by the UAL-based decision layer, while enhancing driving comfort, the objective function is given by:

$$\begin{aligned} J\left(s_k,u_k,a_t^{ref}\right)=\sum_{k=0}^{N_p-1}\Bigg(\left\|v_t^{ref}-v_k\right\|_{q_v}^2+\left\|y_t^{ref}-y_k\right\|_{q_y}^2 \\ +\left\|\Delta u_k^a\right\|_{q_a}^2+\left\|\Delta u_k^\delta\right\|_{q_\delta}^2\Bigg) \end{aligned} \tag{22}$$

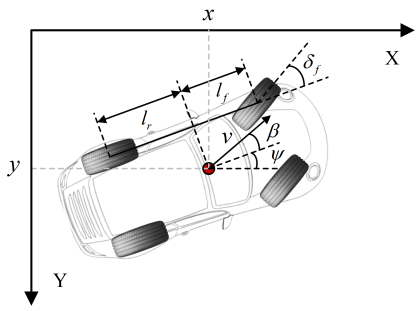


Fig. 2. Bicycle kinematic model

where v_t^{ref} and y_t^{ref} are the reference velocity and reference lateral position respectively. $\mathcal{Q} = \{q_v, q_y, q_a, q_\delta\}$ denotes the set of cost weights. The first and second cost terms aim to encourage the motion planning and control layer to closely track reference signals from the decision layer, while the third and fourth terms prevent excessive acceleration and front wheel steering changes.

3) MPC Formulation: We formulate the motion planning and control problem as a rolling horizon trajectory optimization problem, as follows:

$$\min_{u_{0:N_p-1}} \quad J(s_k, u_k, a_t^{ref}) \tag{23a}$$

$$s.t. \quad s_e(k+1) = f_{\text{ego}}\left(s_e(k), u(k)\right) \tag{23b}$$

$$y_{\text{left}}^{\text{road}} \leq y_e(k) \leq y_{\text{right}}^{\text{road}} \tag{23c}$$

$$v_{\min}^{\text{road}} \leq v_e(k) \leq v_{\max}^{\text{road}} \tag{23d}$$

$$c_k^{i,j}(s_k) > 1, \quad \forall j \in \{1, \ldots, n_c\} \tag{23e}$$

$$u_{\min} \leq u(k) \leq u_{\max} \tag{23f}$$

$$\Delta u_{\min} \leq \Delta u(k) \leq \Delta u_{\max} \tag{23g}$$

where N_p is the prediction horizon, y_{left}^{road} and y_{right}^{road} are the lateral positions of the left and right sides of the road, v_{min}^{road} and v_{max}^{road} are the minimum and maximum speed limits of the road, u_{min} and u_{max} are the minimum and maximum values of the ego vehicle's actions (acceleration and front-wheel steering angle), and Δu_{min} and Δu_{max} are the minimum and maximum values of the rate of change of the ego vehicle's actions (acceleration jerk and front-wheel steering jerk). (23b) adopts the bicycle kinematics model, (23c) and (23d) ensure that the ego vehicle stays within the road boundaries and adheres to the road speed limits during its motion, (23e) represents the dynamic collision avoidance constraint [33], and comfort constraints (23f) and (23g) are used to limit the actions and their rates of change, minimizing abruptness during the ego vehicle's movement. By solving the constrained optimization problem (23), the optimal control sequence $u_{0:N_p-1}^*$ can be obtained. The first optimal control action u_0^* is then used as the control command for the ego vehicle at the current time step.

Finally, the pseudocode of the proposed MeUAL is shown in Algorithm 1.

Algorithm 1: MeUAL

Input: Initialize reward Q-networks $Q_{\theta_r^1}$, $Q_{\theta_r^2}$, cost Q-network ensemble $\left\{Q_{\theta_c^n}\right\}_{n=1}^N$ and policy network $\pi_{\phi^{UAL}}$ with random parameters θ_r^1, θ_r^2, ϕ^{UAL}; Initialize target networks $\bar{\theta}_r^1 \leftarrow \theta_r^1$, $\bar{\theta}_r^2 \leftarrow \theta_r^2$, $\{\bar{\theta}_c^n\}_{n=1}^N \leftarrow \{\theta_c^n\}_{n=1}^N$

Output: $Q_{\theta_r^1}$, $Q_{\theta_r^2}$, $\left\{Q_{\theta_c^n}\right\}_{n=1}^N$, $\pi_{\phi^{UAL}}$, u_0^*

for *each iteration* **do**
  for *each environment step* **do**
    Select decision action $a_t^{ref} \sim \pi_{\phi^{UAL}}(\cdot|s_t)$
    Solve the optimization problem of (23) without collision constraints (23e): $u_{0:N_p-1}^* = MPC(s_k, u_k, a_t^{ref})$
    Execute control action u_0^*
    Observe reward r_t, cost c_t and next state s_{t+1}
    Store transition $(s_t, a_t^{ref}, r_t, c_t, s_{t+1})$ in $\mathcal{D}$
  end
  for *each gradient step* **do**
    Sample mini-batch of N transitions $(s_t, a_t^{ref}, r_t, c_t, s_{t+1})$ from $\mathcal{D}$
    Calculate TD target for rewards and costs $y(r_t, s_{t+1})$, $y(c_t, s_{t+1})$ based on (12) and (14)
    Update reward Q networks $Q_{\theta_r^1}$, $Q_{\theta_r^2}$ based on (11): $\theta_r^m \leftarrow \theta_r^m - \alpha_r \nabla_{\theta_r^m} \mathcal{L}_Q(\theta_r^m), m \in \{1, 2\}$
    Update cost Q-network ensemble $\left\{Q_{\theta_c^n}\right\}_{n=1}^N$ based on (13): $\theta_c^n \leftarrow \theta_c^n - \alpha_c \nabla_{\theta_c^n} \mathcal{L}_Q(\theta_c^n), n \in \{1, 2, \cdots, N\}$
    Update policy network based on (15): $\phi^{UAL} \leftarrow \phi^{UAL} - \alpha_\pi \nabla_{\phi^{UAL}} \mathcal{L}_\pi(\phi^{UAL})$
    Update augmented lagrange multiplier λ_{AL} based on (18): $\lambda_{AL} \leftarrow \lambda_{AL} - \alpha_\lambda \nabla_{\lambda_{AL}} L(\lambda_{AL})$
    Update entropy weight α based on (19): $\alpha \leftarrow \alpha - \alpha_\alpha \nabla_\alpha \mathcal{L}(\alpha)$
    Update target networks via soft update: $\bar{\theta}_r^m \leftarrow \tau\theta_r^m + (1-\tau)\bar{\theta}_r^m, m \in \{1, 2\}$ $\bar{\theta}_c^n \leftarrow \tau\theta_c + (1-\tau)\bar{\theta}_c^n, n \in \{1, 2, \cdots, N\}$
  end
end

D. Safety Guarantee for MeUAL-Driven Automatic Overtaking System

1) Safety Epistemic Uncertainty Threshold: Ideally, the UAL-based decision policy is expected to output optimal actions for any given state. However, in real-world deployment, the ego vehicle frequently encounters OOD scenarios (i.e., beyond the training distribution), where the UAL policy may become fragile and unsafe. To address this issue, the policy not only outputs actions but also provides associated risk information. Specifically, the epistemic uncertainty $\sigma_{Q_{\theta_c^N}}$ quantified by the cost Q-network ensemble (see Section II-B) is employed as a statistical indicator of safety during policy deployment. As the MeUAL algorithm is trained, $\sigma_{Q_{\theta_c^N}}$ gradually converges, allowing the determination of a risk-sensitive safety threshold $c_{Q_c}^{safe}$ (further explained in Section IV-C). When the estimated uncertainty exceeds this threshold, the decision action is considered potentially unsafe. It is important to note that this threshold offers a probabilistic safety bound

derived from the statistical properties of the cost Q-network ensemble, rather than a strict theoretical guarantee as in control-theoretic approaches such as CBF or HJ reachability. Nevertheless, a key advantage of this safety threshold is that it can be pre-adjusted to balance safety and efficiency under OOD scenarios, whereas classical control-theoretic methods may become overly conservative.

2) Policy Switching Mechanism: To enhance the reliability of the MeUAL-driven overtaking system, we incorporate a Policy Switching Mechanism (MeUAL-PSM). As defined in (24), when the epistemic uncertainty of the UAL policy exceeds the safety threshold or the MPC optimization becomes infeasible, the action output by MeUAL is regarded as untrustworthy or unexecutable. In such cases, control authority is transferred to a conservative backup policy π_t^{BP}, implemented by combining the Intelligent Driving Model (IDM) and Minimizing Overall Braking Induced by Lane Changes (MOBIL) [12], [28]. Although IDM+MOBIL is a relatively classical and simplified model, it can still provide a practical lower bound on safety performance, ensuring that the ego vehicle maintains fundamental behaviors such as collision avoidance and lane keeping even in challenging highway scenarios, e.g., dense traffic flows and high observation noise. However, similar to other fallback strategies, as a model-based policy it requires additional optimization to handle extreme scenarios (e.g., aggressive cut-ins, emergency braking). Nevertheless, this is not the focus of the present study, but rather a conservative baseline used to demonstrate the effectiveness of PSM.

$$u_t^{safe} = \begin{cases} \pi_t^{MeUAL}, & \text{if } \sigma_{Q_{\theta_c^N}}(s_t, a_t) < \sigma_{Q_c}^{safe} \wedge u^*_{0:N_p-1} \in \mathcal{U}_{\text{feas}} \\ \pi_t^{BP}, & \text{otherwise} \end{cases} \tag{24}$$

where $\mathcal{U}_{\text{feas}}$ is the feasible solution set. u_t^{safe} is the safety action output by MeUAL-PSM. The pseudocode of MeUAL-PSM is shown in Algorithm 2.

Algorithm 2: MeUAL-PSM

Input: Initialize cost Q-network ensemble $\{Q_{\theta_c^n}\}_{n=1}^N$ and policy network $\pi_{\phi^{UAL}}$ with offline training parameters $\{\theta_c^n\}_{n=1}^N$, ϕ^{UAL}; Initialize backup policy π_t^{BP}, safety epistemic uncertainty threshold $\sigma_{Q_c}^{safe}$

Output: safe action u_t^{safe}

for *each environment step* **do**
- Observe driving environment state s_t
- Select decision aciton $a_t^{ref} \sim \pi_{\phi^{UAL}}(\cdot|s_t)$
- Calculate epistemic uncertainty of cost Q-network ensemble based on (10)
- **if** $\sigma_{Q_{\theta_c^N}}(s_t, a_t) < \sigma_{Q_c}^{safe}$ **then**
  - Solve optimization problem of (23) with collision constraints (23e): $u^*_{0:N_p-1} = MPC(s_k, u_k, a_t^{ref})$
  - **if** $u^*_{0:N-1} \in \mathcal{U}_{\text{feas}}$ **then**
    - $u_t^{safe} = u_0^*$
  - **else**
    - $u_t^{safe} = \pi_t^{BP}$
  - **end**
- **else**
  - $u_t^{safe} = \pi_t^{BP}$
- **end**

end

III. Application to Autonomous Highway Overtaking

A. Driving Scenario Setup

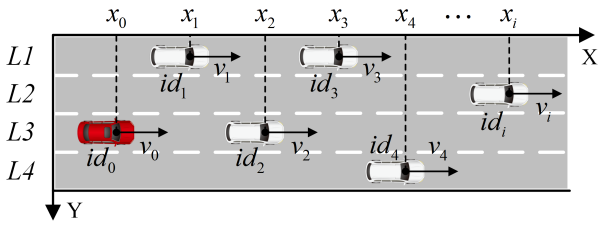


Fig. 3. Schematic diagram of a four-lane highway. The red car represents the ego vehicle, while the white cars represent the surrounding vehicles.

To evaluate the performance of the proposed MeUAL and baseline algorithms, a four-lane highway scenario is constructed using the highway-env simulator [34]. The driving scenario consists of a four-lane highway with a lane width of 4 m , as shown in Fig. 3. In this scenario, the length and width of the vehicle are set to 5 m and 2 m, respectively. The surrounding vehicles are controlled by a driver model that includes the IDM for longitudinal car-following and the MOBIL for lateral lane-changing maneuvers. To create an overtaking traffic environment, the initial speed v_0 of the ego vehicle is set to 25 m/s, the initial speeds v_i of the surrounding vehicles are randomly generated in the range of [21, 24] m/s, and the slower surrounding vehicles are placed in front of the ego vehicle. To make the traffic environment more challenging, the politeness coefficient of the MOBIL model is set to 0, that is, the vehicle does not consider the impact on surrounding vehicles when changing lanes, and the acceleration coefficient δ_i of the IDM model is randomly selected in the range of [3.5, 4.5] m/s^2. The initial driving lane id_i of all vehicles is randomly selected from the four available lanes $[L1, L2, L3, L4]$. The initial longitudinal position x_i can be described as follows:

$$\Delta x_i = \varepsilon(m + v_i) \tag{25}$$

$$x_i = \begin{cases} n\Delta x_i, & if \quad i = 0 \\ x_{i-1} + \Delta x_i, & if \quad i > 0 \end{cases} \tag{26}$$

where m is a constant used to adjust the longitudinal position offset Δx_i between vehicles, ε is a random number uniformly distributed between 0.9 and 1.1, used to add noise to Δx_i. n is a constant used to control the initial position x_0 of the ego vehicle. In this paper, i takes the value of 20, indicating that there are 20 surrounding vehicles.

To verify the effectiveness of the proposed policy switching mechanism in a more realistic simulation platform, we reproduce the same highway simulation scenario in CARLA, as

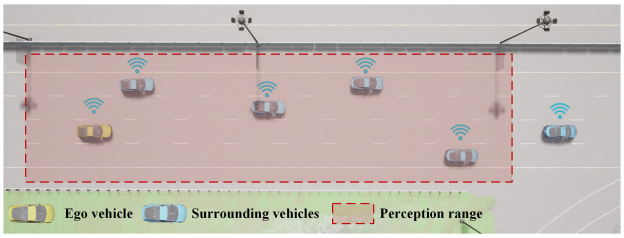


Fig. 4. The highway simulation scenario reconstructed in CARLA.

shown in Fig. 4. In this scenario, the following assumptions are made:

• The ego vehicle can accurately obtain the state information (e.g., position, velocity, heading) of itself and surrounding vehicles within the perception range.

• Gaussian noise is added to the state information of the ego vehicle and surrounding vehicles to simulate perception-level noise.

Automatic overtaking is an essential strategy for improving the driving efficiency of autonomous vehicles. However, because it necessitates maximizing speed and continuously maneuvering around slower vehicles in dynamic and uncertain traffic conditions, the risk of collisions significantly increases due to insufficient safe distances from surrounding vehicles. The central challenge in this task is to find the optimal trade-off between driving risk and overtaking performance.

B. Design of Driving Risk Field

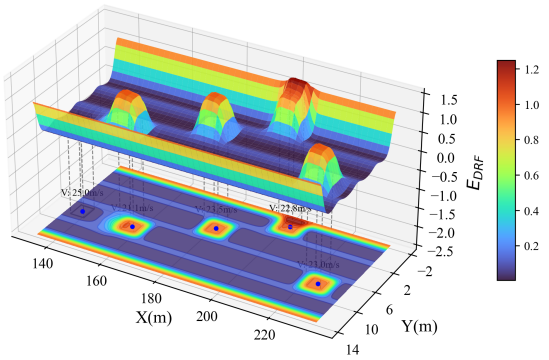


Fig. 5. The driving risk field of a four-lane highway

In order to quantify the driving risk level of the ego vehicle in a dynamic driving environment, this study subdivides the Driving Risk Field (DRF) into static vehicle risk field, dynamic vehicle risk field, road boundary risk field, and road marking risk field.

1) Static Vehicle Risk Field: The static vehicle risk field is constructed based on the relative distance and the vehicle's dimensions. The smaller the relative distance, the larger the risk field. In addition, a two-dimensional Gaussian function with a high-order center distance is used to characterize the risk field of the surrounding vehicle dimensions. The high-order center distance flattens the peak of the function, so that the surface of each surrounding vehicle has a similar risk field. The static vehicle risk field E_{sta} is defined as follows:

$$\begin{cases} E_{sta} = A_{sta} \exp\left(-\left(\frac{(x-x_{sur})^2}{\sigma_x^2}\right)^{\beta} - \left(\frac{(y-y_{sur})^2}{\sigma_y^2}\right)^{\beta}\right) \\ \sigma_x = k_x L_{sur} \\ \sigma_y = k_y W_{sur} \end{cases} \tag{27}$$

where A_{sta} is the static vehicle risk field coefficient, (x, y) is the position coordinate of a point on the highway, (x_{sur}, y_{sur}) is the position coordinate of the center point of the surrounding vehicles, β is the shape factor used to adjust the distribution shape of the surface risk field of the surrounding vehicles, L_{sur} is the body length of the surrounding vehicles, and W_{sur} is the body width of the surrounding vehicles.

2) Dynamic Vehicle Risk Field: The construction of the dynamic vehicle risk field is based on the approach direction, relative distance and absolute value of relative speed of the ego vehicle relative to the surrounding vehicles. The closer the ego vehicle is to the surrounding vehicles, the smaller the relative distance is, and the greater the absolute value of the relative speed is, the greater the risk of collision is, so the dynamic vehicle risk field formed behind the surrounding vehicles is also greater. The dynamic vehicle risk field E_{dyn} is defined as follows:

$$\begin{cases} E_{dyn} = A_{dyn} \frac{\exp\left(-\left(\frac{(x-x_{sur})^2}{\sigma_v^2}\right)^{\beta} - \left(\frac{(y-y_{sur})^2}{\sigma_y^2}\right)^{\beta}\right)}{1+\exp(-rel_v(x-x_{sur}-\alpha L_{sur} rel_v))} \\ \sigma_v = k_v |v - v_{sur}| \\ rel_v = \begin{cases} 1, v_{sur} \geq v \\ -1, v_{sur} < v \end{cases} \end{cases} \tag{28}$$

where A_{dyn} is the dynamic vehicle risk field coefficient, σ_v is a function of the absolute value of the relative velocity between the ego vehicle and surrounding vehicles, k_v is the relative velocity coefficient, rel_v is used to describe the relative velocity direction between the ego vehicle and surrounding vehicles, and α is used to adjust the influence of the $L_{sur} rel_v$ term.

It should be noted that when the yaw angle ψ of the surrounding vehicles is not 0, the risk field (x, y, E) in (27) and (28) needs to be deflected by the corresponding angle, which is defined as follows:

$$\begin{cases} x' = (x - x_{sur})\cos\psi - (y - y_{sur})\sin\psi + x_{sur} \\ y' = (x - x_{sur})\sin\psi + (y - y_{sur})\cos\psi + y_{sur} \\ E' = E \end{cases} \tag{29}$$

3) Road Boundary Risk Field: The closer the vehicle is to the road boundary, the greater the risk of running out of the lane. Therefore, the risk field E_b generated by the road boundary is defined as follows:

$$E_b = A_b \exp\left(-\frac{(l_{pos} - l_b)^2}{2\sigma_b^2}\right) \tag{30}$$

4) Road Marking Risk Field: Considering that the closer the ego vehicle is to the road marking line, the greater the risk of collision with the adjacent vehicle. Therefore, in order to encourage the ego vehicle to drive on the center line of each

lane as much as possible, the risk field E_l formed by the lane marking line is defined as follows:

$$E_l = A_l \exp\left(-\frac{(l_{pos} - l_l)^2}{2\sigma_l^2}\right) \tag{31}$$

Therefore, the total driving risk field E_{DRF} of the ego vehicle is the superposition of the static vehicle risk field, the dynamic vehicle risk field, the road boundary risk field and the road marking risk field, as shown in Fig. 5, as follows:

$$E_{\mathrm{DRF}} = E_{sta} + E_{dyn} + E_b + E_l \tag{32}$$

C. CMDP Formulation

1) State Space: Assume that the state information of the ego vehicle and surrounding vehicles can be accurately obtained. The state space includes the ego vehicle state S_e and the surrounding vehicle state S_i. The ego vehicle state S_e consists of the following parameters: longitudinal position x_e, lateral position y_e, longitudinal velocity v_{x_e}, lateral velocity v_{y_e}, cosine value of the yaw angle $\cos\psi_e$ and sine value of the yaw angle $\sin\psi_e$. The surrounding vehicle state S_i includes: relative longitudinal position Δx_{ie}, relative lateral position Δy_{ie}, relative longitudinal velocity $\Delta\nu_{x_{ie}}$, relative lateral velocity$\Delta\nu_{y_{ie}}$, cosine value of the yaw angle $\cos\psi_i$ and sine value of the yaw angle $\sin\psi_i$. The state space is defined as follows:

$$\begin{cases} S = (S_e, S_i) \\ S_e = (x_e, y_e, v_{x_e}, v_{y_e}, \cos\psi_e, \sin\psi_e) \\ S_i = (\Delta x_{ie}, \Delta y_{ie}, \Delta v_{x_{ie}}, \Delta v_{y_{ie}}, \cos\psi_i, \sin\psi_i) \\ -l_1 < \Delta x_{ie} < l_2, i \leq 4 \end{cases} \tag{33}$$

where l_1 and l_2 represent the detection distances backward and forward, respectively. The constraint $i \leq 4$ means that the states of up to 4 vehicles that satisfy the constraint $-l_1 < \Delta x_{ie} < l_2$ will be added to the surrounding vehicle state S_i. If the number of surrounding vehicles that satisfy the constraint is less than 4, the corresponding surrounding vehicle state will be filled with 0. In order to improve the stability of the training process, each element of the state space is mapped to the range of [-1,1] for normalization.

2) Action Space: In this study, the action space A of the decision layer consists of the reference velocity v^{ref} and the reference lateral position y^{ref}, which are defined as follows:

$$A = \left(v^{ref}, y^{ref}\right) \tag{34}$$

Note that the action output by the decision policy is normalized to the range of [-1, 1] through the tanh activation function. When used as a reference signal for the motion planning and control layer, the decision action is further mapped to the physically feasible range.

3) Reward Function Design: In the context of safe RL, the performance of the vehicle (efficiency and comfort) and the driving risk (safety) are distinguished by designing reward functions and cost functions. Specifically, the reward function includes efficiency rewards and comfort rewards, which are used to evaluate the performance of the vehicle in the driving task, as follows:

$$r = w_e r_{effi} + w_c r_{comf} \tag{35}$$

where w_e and w_c are the weight parameters of efficiency reward r_{effi} and comfort reward r_{comf} respectively. The design concept of efficiency reward and comfort reward is as follows:

• *Efficiency Reward:* In order to prevent the ego car from reducing the risk of collision by driving at a lower speed, a speed-related efficiency reward is designed to encourage the ego car to increase its speed as much as possible, as follows:

$$r_{effi} = -1 + \frac{2(v_e - v_{min})}{v_{max} - v_{min}} \tag{36}$$

• *Comfort Reward:* In order to improve driving comfort, a comfort reward is set that comprehensively considers the acceleration change rate and the front wheel angle change rate to avoid excessive acceleration, braking, or steering, as follows:

$$r_{comf} = -(w_a \Delta u^a + w_{\delta_f} \Delta u^{\delta_f}) \tag{37}$$

It should be noted that, since autonomous highway overtaking requires carefully balancing safety and efficiency, the weight of the comfort reward related to action smoothness is deliberately set to a relatively small value in this paper.

4) Cost Function Design: In the process of interaction between the safe RL agent and the environment, in addition to receiving a reward signal, an additional cost signal is obtained. A well-designed cost function can help the self-driving car comprehensively evaluate the driving risks in the process of completing driving tasks in a dynamic driving environment. Conventional safe RL settings usually give unsafe situations a sparse cost, which means that the safe RL agent will not receive a cost signal in most time steps, thus affecting the convergence speed and safety of the safe RL policy. Therefore, in this study, the cost function consists of a dense cost c_{DRF} based on DRF and a sparse cost of running out of the lane c_{out} and colliding c_{col}, as follows:

$$\begin{cases} c_t = c_{DRF} + c_{out} + c_{col} \\ c_{DRF} = E_{DRF} \\ c_{out} = \begin{cases} 5, if \text{ out} \\ 0, \text{otherwise} \end{cases} \\ c_{col} = \begin{cases} 5, if \text{ crashed} \\ 0, \text{otherwise} \end{cases} \end{cases} \tag{38}$$

D. Implementation Details

1) Training and deployment Details: The learning-based RL algorithm is implemented using PyTorch, while the MPC solver is built using the open source CasADi tool. The MeUAL algorithm and the baseline algorithms are trained on a computer equipped with an Intel i9-14900K CPU. The converged policy is deployed on an NVIDIA Jetson Orin NX platform to assess real-time execution efficiency. During the entire training process, the RL agent carries out 300,000 training steps. The policy operates at a control frequency

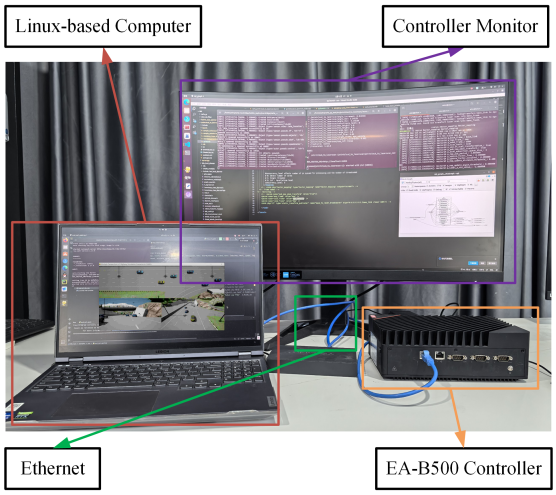


Fig. 6. Processor-in-the-Loop Test Platform for Deploying Decision-Making and Control policies

of 10 Hz, and each episode lasts 20 seconds. An episode terminates when the ego vehicle collides with surrounding vehicles, departs from its lane, or when the episode duration reaches its maximum limit. When the driving scenario is reset, the configuration parameters (including initial vehicle speed v_i, the IDM acceleration coefficient δ_i, initial driving lane id_i, and initial longitudinal position x_i) are updated randomly.

Processor-in-the-Loop (PIL) testing is designed to evaluate the real-time performance of pre-trained decision-making and control policies when deployed on a physical controller. As illustrated in Fig. 6, the PIL test platform comprises four core components: an EA-B500 controller equipped with the NVIDIA Jetson Orin NX (for policy execution), a controller display (for status monitoring), a Linux-based computer hosting the CARLA high-fidelity simulation platform (for scenario simulation), and an Ethernet (for data transmission). To ensure low-latency data interaction, the EA-B500 controller and Linux-based computer exchange environmental states and control commands over Ethernet, using the Real-Time Publish-Subscribe (RTPS) protocol.

2) Comparison Metrics:

• *Average Episode Reward (AER):* This metric denotes the average total reward per episode for the policy across multiple test episodes. A higher value indicates better overtaking performance.

• *Average Episode Cost (AEC):* This metric denotes the average total cost per episode incurred by the policy across multiple test episodes. A lower value indicates stronger adherence to safety constraints.

• *Success Rate (SR):* This metric represents the proportion of collision-free episodes among all test episodes. A higher success rate indicates a safer policy.

• *Violation Rate (VR):* This metric is defined as the ratio of the total number of hard physical constraint violations to the total number of steps. A lower violation rate indicates stronger adherence to hard physical constraints.

• *Average Speed (AS):* This metric denotes the average vehicle speed across multiple test episodes, reflecting the policy's driving efficiency.

• *Activation Rate (AR):* This metric refers to the ratio of backup policy takeover steps to total steps, reflecting the backup policy's usage frequency.

3) Comparison Baselines:

• *CPO [15]:* CPO is a constrained RL algorithm based on constrained policy search, guaranteeing near-constraint satisfaction at each iteration.

• *PPO-L [35]:* PPO-L is an on-policy safe RL algorithm that combines Proximal Policy Optimization (PPO) with the Lagrangian method.

• *SAC-L [36]:* SAC-L is an off-policy safe RL algorithm that incorporates Lagrangian relaxation into Vanilla-SAC.

• *CVPO [21]:* CVPO is a constrained RL algorithm based on the Expectation-Maximization approach, which naturally embeds constraints into the policy learning process.

• *MeSAC-L:* MeSAC-L is a model-enhanced safe RL algorithm that extends the framework based on MPC and SAC in [26] by incorporating Lagrangian optimization.

• *IDM+MOBIL:* The IDM+MOBIL model introduced in Section III-A serves as a comparative baseline.

IV. Experimental Results and Discussion

A. Performance of MeUAL

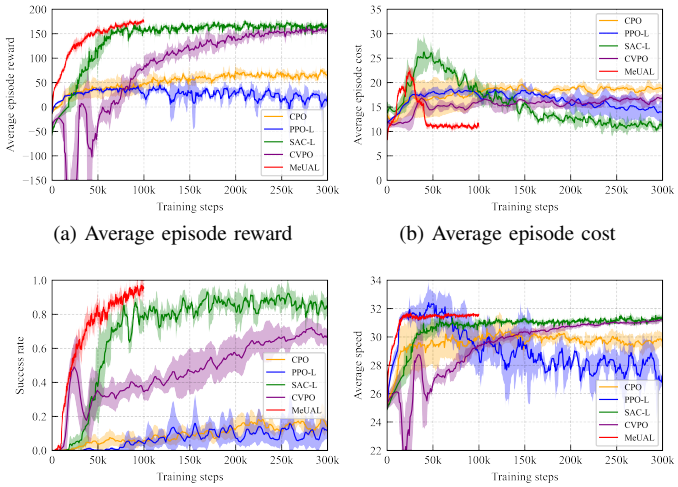
In this experiment, all algorithms are independently trained using five random seeds to evaluate the performance advantage of MeUAL over baseline algorithms (CPO, PPO-L, SAC-L, and CVPO). As shown in Fig. 7, MeUAL exhibits clear advantages in the learning curves across different metrics. Regarding average episode reward, MeUAL converges to the highest level and maintains stability, while SAC-L and CVPO perform similarly but at slightly lower levels, and CPO and PPO-L perform poorly. With respect to average episode cost, MeUAL steadily converges to the lowest level, highlighting its safety advantage, whereas CPO, PPO-L, and CVPO maintain consistently high costs. As for the success rate, MeUAL rapidly surpasses all baseline algorithms. Concerning the average speed, MeUAL is on par with SAC-L and CVPO, and significantly higher than CPO and PPO-L. Notably, at 50,000 training steps, MeUAL's learning curves for various evaluation metrics significantly outperform those of the other baseline algorithms. This improvement arises because MPC shrinks the state space that the safe RL agent can explore, thereby reducing the need for interaction samples. The quantitative results in Table I further confirm these observations: MeUAL achieves the highest reward, success rate, and speed, the lowest cost, and the smallest variance across all metrics. In summary, the proposed MeUAL outperforms baseline algorithms in terms of reward–cost balance, sample efficiency, and learning stability.

B. Ablation Study on the Key Components of MeUAL

In this experiment, we performed an ablation study on the key components of the MeUAL algorithm. Fig. 8 and Table II present the training process and evaluation results, respectively. As shown in Fig. 8(b) and Table II, the average episode costs of MeSAC-UL and MeUAL are both lower than those of MeSAC-L and MeSAC-AL, indicating

TABLE I
EVALUATION RESULTS OF MEUAL AND BASELINE METHODS ON HIGHWAY TRAINING SCENARIOS IN HIGHWAY-ENV

Algorithms	AER ↑	AEC ↓	SR (%) ↑	AS (m/s) ↑
CPO	67.3±8.6	18.8±1.0	16.1±7.0	29.7±0.7
PPO-L	14.1±18.5	14.3±2.3	10.6±7.7	27.4±1.3
SAC-L	164.8±7.4	11.5±1.1	83.8±7.4	31.3±0.3
CVPO	157.1±7.2	16.3±1.0	68.6±6.4	31.1±0.2
MeUAL	175.0±5.1	10.8±0.8	95.4±5.6	31.5±0.1



(a) Average episode reward (b) Average episode cost

(c) Success rate (d) Average Speed

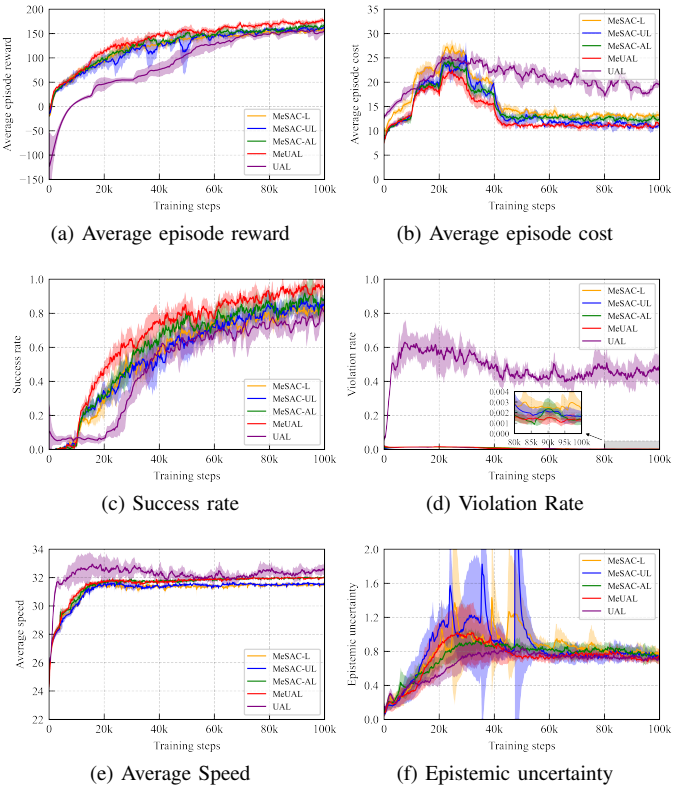
Fig. 7. Learning curves of MeUAL and baseline algorithms. The solid line corresponds to the average of five independent trainings, and the shaded area corresponds to the 95% confidence interval. Since the MeUAL has achieved satisfactory convergence in the early stages, we terminated the training of the MeUAL algorithm early at 100,000 steps to save computing resources.

that uncertainty-aware cost Q-value estimation can enhance policy constraint satisfaction. Moreover, Fig. 8(f) illustrates that the epistemic uncertainty of MeSAC-L and MeSAC-UL exhibits significant fluctuations during training, which verifies the inherent issue of policy oscillation in Lagrangian-based constrained policy optimization. In contrast, MeSAC-AL and MeUAL effectively alleviate policy learning instability by incorporating an enhanced Lagrangian mechanism.

Furthermore, Fig. 8 and Table II demonstrate that within only 100k training steps, MeUAL achieves a 17.3% higher success rate and a substantially lower average episode cost compared with UAL, indicating clear safety performance improvements. Notably, as seen in Fig. 8(d), the violation rate of hard physical constraints decreases by 47% and remains nearly eliminated when using the MPC-enhanced safe RL framework. Benefiting from MPC's capability to explicitly enforce multiple physical constraints, MeUAL delivers consistent gains in both learning efficiency and overall safety.

C. Safety Threshold Determination

As mentioned above, the epistemic uncertainty of the proposed MeUAL converges to a lower level at the end of training. As observed in Fig. 9, during the initial training phase, the cognitive uncertainty increases dramatically and the confidence interval remains broad. This indicates that the cost



(a) Average episode reward (b) Average episode cost

(c) Success rate (d) Violation Rate

(e) Average Speed (f) Epistemic uncertainty

Fig. 8. Learning curves for different ablation variants of MeUAL. MeSAC-AL sets the conservative parameter k in (8) to 0, and MeSAC-UL sets the convexity parameter c in (10) to 0.

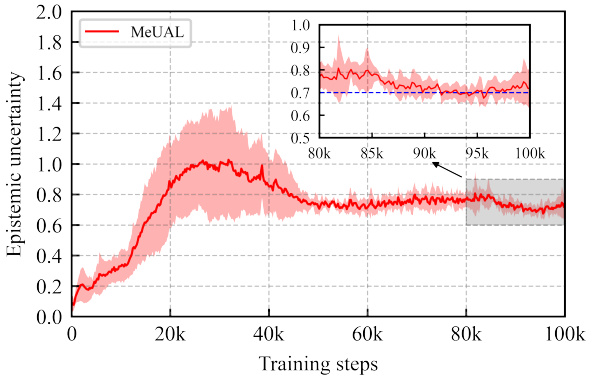


Fig. 9. Learning curve of epistemic uncertainty for MeUAL

Q-network ensemble has not been fully learned in the initial stage and lacks sufficient cognition of driving risks. With the continuous collection of environmental interaction samples and the iterative update of the cost Q-network ensemble, the epistemic uncertainty gradually converges to 0.7, and the confidence interval gradually narrows. This shows that the cognitive divergence of the cost Q-network ensemble for the same state-action pair is decreasing, and the integrated network becomes more confident. It is noteworthy that the epistemic uncertainty dose not converge to 0, which reflects the inherent uncertainty of the highway overtaking task. The safety threshold of epistemic uncertainty, σ_{Qc}^{safe}, plays a critical role in the design of MeUAL-PSM. To determine an appropriate

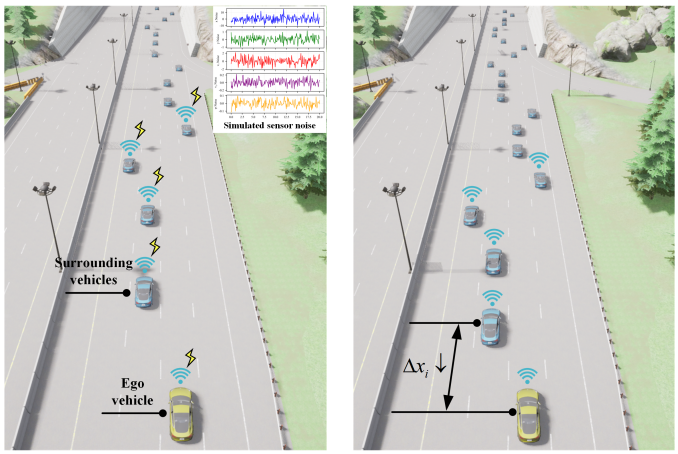
TABLE II
EVALUATION RESULTS OF DIFFERENT ABLATION VARIANTS OF MEUAL ON HIGHWAY TRAINING SCENARIOS IN HIGHWAY-ENV

Algorithms	Components				Performance Metrics				
	MPC	UAL			AER ↑	AEC ↓	SR (%) ↑	VR (%) ↓	AS (m/s) ↑
		L	U	A					
MeSAC-L	✓	✓			157.0±3.8	13.1±0.6	82.0±5.1	0.3±0.1	31.5±0.2
MeSAC-UL	✓	✓	✓		163.8±6.8	11.1±0.5	84.3±5.9	0.2±0.1	31.5±0.1
MeSAC-AL	✓	✓		✓	164.0±5.7	12.5±0.9	86.2±7.1	0.1±0.0	32.0±0.0
MeUAL	✓	✓	✓	✓	175.0±5.1	10.8±0.8	95.4±5.6	0.1±0.0	32.0±0.0
UAL		✓	✓	✓	152.8±11.9	19.6±1.0	78.1±8.5	47.1±8.7	32.5±0.4

value, we evaluated three thresholds (0.6, 0.7, and 0.8), and the results are summarized in Table III. Compared to MeUAL, MeUAL-PSM0.7 reduces the average episode cost by 29.6% and further improves the success rate, although the average speed is reduced by 7.8%. This shows the effectiveness of MeUAL-PSM0.7 in dealing with uncommon scenarios. Accordingly, a safety threshold of $\sigma_{Qc}^{safe} = 0.7$ is adopted in the following experiments.

TABLE III
EVALUATION RESULTS OF MEUAL-PSM WITH DIFFERENT EPISTEMIC UNCERTAINTY THRESHOLDS ON HIGHWAY TRAINING SCENARIOS IN HIGHWAY-ENV

Algorithms	AEC ↓	AS (m/s) ↑	SR (%) ↑	AR (%)
MeUAL (w/o PSM)	10.8±0.8	32.0±0.0	95.4±5.6	/
MeUAL-PSM0.6	6.6±0.1	28.4±0.4	100.0±0.0	35.0±3.0
MeUAL-PSM0.7	7.6±0.4	29.5±0.3	100.0±0.0	25.0±2.0
MeUAL-PSM0.8	8.3±0.3	30.1±0.3	96.0±3.0	18.0±2.0



(a) Sensor noise+40% (b) Traffic density+50%

Fig. 11. Highway simulation scenarios in CARLA. The WiFi symbols represent vehicle-to-vehicle communication, and the lightning symbols indicate simulated sensor noise injected into the states of both the ego vehicle and surrounding vehicles.

D. Testing with Unfamiliar Cases

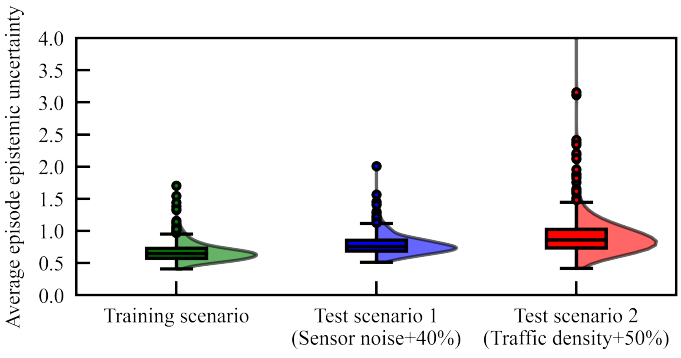


Fig. 10. Distribution of average episode epistemic uncertainty of MeUAL in the training scenario and two test scenarios (high sensor noise and high traffic density).

To evaluate the effectiveness of MeUAL-PSM in handling unfamiliar scenarios, we design two highway test scenarios in CARLA that are far from the training distribution. As can be seen from Fig. 10, compared with the training scenarios, the average episode epistemic uncertainty of the test scenarios with 40% sensor noise added and the test scenarios with 50% increased traffic density are distributed in higher and more dispersed areas.

Due to the sensitivity of RL policies to noise in state inputs, it is usually assumed that the input state is accurate. Therefore, in the first test scenario, Gaussian noise is added to the states of the ego vehicle and the four surrounding vehicles it observes (longitudinal position x, lateral position y, longitudinal velocity v_x, lateral velocity y_x, and yaw angle ψ) to simulate perception-level noise, as shown in Fig. 11(a). The vehicle state with noise is represented as:

$$X_{noise} = X + pN(0, \sigma^2) \tag{39}$$

where $X = \left[x, y, v_x, v_y, \psi\right]^T$, p is the noise ratio, set at 40%, and $\mathcal{N}(0, \sigma^2)$ denotes a Gaussian distribution with a mean of 0 and a covariance matrix $\sigma^2 = \text{diag}\left(\sigma_x^2, \sigma_y^2, \sigma_{v_x}^2, \sigma_{v_y}^2, \sigma_\psi^2\right)$. The specific parameter settings are: $\sigma_x = 10\,\text{m}, \sigma_y = 1\,\text{m}, \sigma_{v_x} = 2\,\text{m/s}, \sigma_{v_y} = 0.2\,\text{m/s}, \sigma_\psi = 0.1\,\text{rad}$. As shown in Table IV, which reports the evaluation results for the test scenario with 40% sensor noise, MeUAL consistently outperforms CVPO, SAC-L, and MeSAC-L in terms of average episode cost, average speed, and success rate, demonstrating its robustness under noisy conditions. Compared to MeUAL, MeUAL-PSM further reduces the average episode cost by 48.9% and achieves a success rate of 98%. In addition, the activation rate of the backup policy π_t^{BP}, implemented by the IDM+MOBIL combination, reaches 49%. These results clearly

TABLE IV
EVALUATION RESULTS OF MEUAL-PSM AND BASELINE METHODS ON DIFFERENT HIGHWAY TEST SCENARIOS IN CARLA

Cases	Algorithms	AEC ↓	AS (m/s) ↑	SR (%) ↑	AR (%)
Test scenario 1 (Sensor noise+40%)	CVPO	15.5±1.2	31.0±0.2	71.0±3.2	/
	SAC-L	13.1±0.4	30.9±0.0	77.0±4.0	/
	MeSAC-L	13.2±0.8	31.3±0.0	78.0±7.0	/
	MeUAL	10.6±0.4	31.6±0.0	84.0±2.0	/
	MeUAL-PSM	5.4±0.3	27.9±0.3	98.0±2.0	49.0±2.0
	IDM+MOBIL	2.3±0.2	24.1±0.3	100.0±0.0	100.0±0.0
Test scenario 2 (Traffic density+50%)	CVPO	25.1±0.8	29.5±0.2	34.9±3.5	/
	SAC-L	25.3±1.0	29.5±0.1	36.0±4.0	/
	MeSAC-L	23.1±0.6	30.2±0.0	43.0±5.0	/
	MeUAL	22.9±0.8	30.6±0.0	58.0±5.0	/
	MeUAL-PSM	5.3±0.2	28.0±0.3	99.0±1.0	50.0±2.0
	IDM+MOBIL	2.5±0.1	21.1±0.2	100.0±0.0	100.0±0.0

validate the effectiveness of MeUAL-PSM in improving safety and reliability in simulated sensor noise scenarios.

To evaluate the performance of MeUAL-PSM in the dense traffic scenarios, we increased the traffic density by 50% in the highway simulation scenario constructed in CARLA (i.e., shortened the longitudinal distance between vehicles), as illustrated in 11(b). Table IV shows that the safety (in terms of average episode cost and success rate) of CVPO, SAC-L, MeSAC-L, and MeUAL deteriorates significantly. This indicates that RL policies trained under regular traffic conditions pose greater safety risks in high-density traffic scenarios. However, compared to MeUAL, MeUAL-PSM reduces the average episode cost by 76.8% and improves the success rate by 41%. Furthermore, relative to IDM+MOBIL, MeUAL-PSM achieves a 32.7% increase in average speed. These results demonstrate that the proposed MeUAL-PSM effectively improves driving efficiency while maintaining safety in this unseen high-density traffic scenarios.

E. Analysis of the Overtaking Behavior of MeUAL-PSM

In this experiment, a highly challenging testing scenario is designed in CARLA. Environmental complexity is increased significantly by introducing 40% sensor noise and raising traffic density by 50%. Meanwhile, episode duration is extended from 20s to 30s, which enables a complete observation of overtaking behaviors and policy switching processes.

As shown in Fig. 12(e) and (i), when four vehicles occupied parallel positions ahead of the ego vehicle, the MeUAL-controlled ego vehicle failed to decelerate substantially, ultimately colliding at 29.5 m/s. In contrast, Fig. 13(b) demonstrates that the MeUAL-PSM-controlled ego vehicle reduced its speed to 24.9 m/s within timesteps 100-110 while maintaining a safe distance from preceding vehicles. Fig. 14 reveals that the epistemic uncertainty value of MeUAL exceeded the safety threshold during this period, confirming that the four-vehicle parallel scenario surpassed the cognitive scope of the MeUAL policy. Consequently, the ego vehicle successfully switched to the conservative IDM+MOBIL policy to avoid collision. Further analysis of Fig. 13(c)-(d) indicates that the first viable lane-changing gap emerged during timesteps 190-240, coinciding with the reduction in epistemic uncertainty below the safety threshold observed in Fig. 14. The ego vehicle promptly

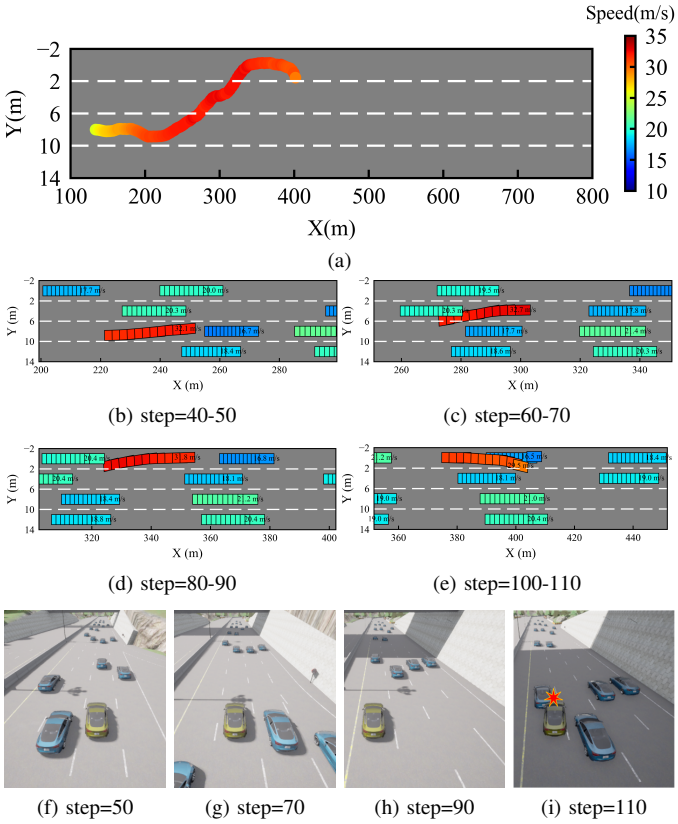


Fig. 12. The overtaking process of MeUAL in the challenging highway test scenario. (a) shows the driving trajectory of MeUAL. (b)-(e) are local enlarged views of specific step ranges. (f)-(i) are third-person views of the ego vehicle.

reverted to the MeUAL policy, executing lane-changing maneuvers from the first to third lane while accelerating from 17.9 m/s to 29.0 m/s. Finally, as depicted in Fig. 13(e), when an appropriate lane-changing gap appeared ahead in the first lane, the ego vehicle maintained high overtaking speed to complete lane transition. In summary, MeUAL-PSM is capable of seizing overtaking opportunities in a timely manner while ensuring safety through flexible policy switching, and exhibits interpretable overtaking behavior. Additionally, a supplemental video of this scene is provided[1].

[1]https://github.com/zsn2021/MeUAL

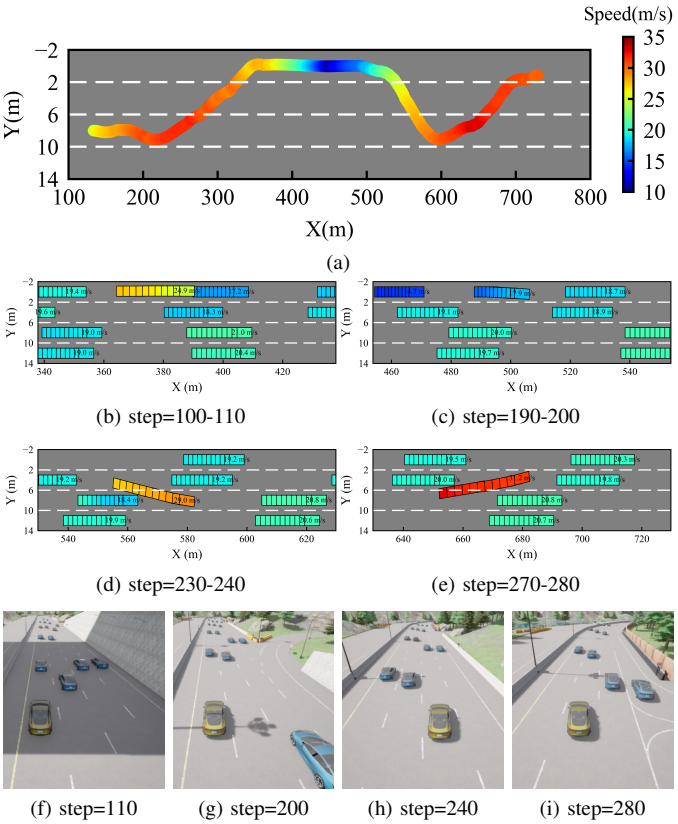


Fig. 13. The overtaking process of MeUAL-PSM in the challenging highway test scenario. (a) shows the driving trajectory of MeUAL-PSM. (b)-(e) are local enlarged views of specific step ranges. (f)-(i) are third-person views of the ego vehicle.

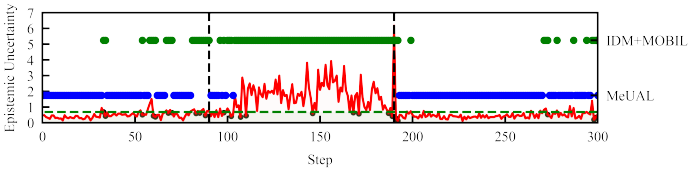


Fig. 14. The change of epistemic uncertainty and policy switching process of MeUAL-PSM in a specific test scenario. The green dashed line is the safety epistemic uncertainty threshold. The black scattered points are infeasible solutions.

Furthermore, the proposed method is deployed on the EA-B500 controller equipped with the NVIDIA Jetson Orin NX (as shown in Fig. 6). In terms of inference time, the average computation time of the UAL policy network in the decision layer is 0.56 ms. The average solution time of the MPC-based motion planning and control layer (23) is 8.3 ms. The computation time of both the safe RL policy network and the MPC solver remains stable and does not vary significantly across different highway simulation scenarios (e.g., varying sensor noise or traffic density). These findings validate the real-time applicability of the proposed method when deployed on edge computing hardware such as the Jetson Orin NX.

V. Conclusion

This study proposes a decision-making and control framework for automatic overtaking based on the MeUAL method, which aims to maximize vehicle driving efficiency while ensuring safety. The framework integrates the data-driven constraint optimization capability of Lagrangian-based safe RL, while preserving the advantages of traditional MPC methods in safety and constraint handling. Experimental results demonstrate that MeUAL outperforms existing benchmark algorithms in terms of the reward-cost trade-off, sample efficiency, and learning stability. To enhance the reliability of the MeUAL-driven automatic overtaking system in real-world deployment, this study proposes a policy switching mechanism triggered by a safety-aware uncertainty threshold. The robustness and interpretable overtaking behavior of MeUAL-PSM are verified in multiple challenging test scenarios.

Future work will first consider adopting a data-driven trajectory prediction model for surrounding vehicles to replace the current constant-speed assumption, thereby enhancing the safety and constraint satisfaction capabilities of the MPC-based motion planning and control module. Secondly, a relatively aggressive backup strategy will be designed to replace the current conservative IDM+MOBIL method, with the aim of further improving the driving efficiency of the hybrid approach. Finally, the proposed approach will be implemented and evaluated on a real-world autonomous vehicle platform.

References

[1] S. Teng, X. Hu, P. Deng, B. Li, Y. Li, Y. Ai, D. Yang, L. Li, Z. Xuanyuan, F. Zhu *et al.*, "Motion planning for autonomous driving: The state of the art and future perspectives," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 6, pp. 3692–3711, 2023.

[2] Y. Guan, Y. Ren, Q. Sun, S. E. Li, H. Ma, J. Duan, Y. Dai, and B. Cheng, "Integrated decision and control: Toward interpretable and computationally efficient driving intelligence," *IEEE transactions on cybernetics*, vol. 53, no. 2, pp. 859–873, 2022.

[3] K. Yuan, Y. Huang, S. Yang, M. Wu, D. Cao, Q. Chen, and H. Chen, "Evolutionary decision-making and planning for autonomous driving: A hybrid augmented intelligence framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 7, pp. 7339–7351, 2024.

[4] J. Wu, C. Huang, H. Huang, C. Lv, Y. Wang, and F.-Y. Wang, "Recent advances in reinforcement learning-based autonomous driving behavior planning: A survey," *Transportation Research Part C: Emerging Technologies*, vol. 164, p. 104654, 2024.

[5] P. S. Chib and P. Singh, "Recent advancements in end-to-end autonomous driving using deep learning: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 103–118, 2023.

[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] L. Le Mero, D. Yi, M. Dianati, and A. Mouzakitis, "A survey on imitation learning techniques for end-to-end autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 128–14 147, 2022.

[8] Y. Lu, J. Fu, G. Tucker, X. Pan, E. Bronstein, R. Roelofs, B. Sapp, B. White, A. Faust, S. Whiteson *et al.*, "Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7553–7560.

[9] S. E. Li, "Reinforcement learning for sequential decision and optimal control," 2023.

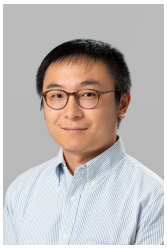
[10] D. Chen, M. R. Hajidavalloo, Z. Li, K. Chen, Y. Wang, L. Jiang, and Y. Wang, "Deep multi-agent reinforcement learning for highway on-ramp merging in mixed traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 11 623–11 638, 2023.

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

[12] S. Zhang, W. Zhuang, B. Li, K. Li, T. Xia, and B. Hu, "Integration of planning and deep reinforcement learning in speed and lane change decision-making for highway autonomous driving," *IEEE Transactions on Transportation Electrification*, 2024.
[13] W. Xiao, Y. Yang, X. Mu, Y. Xie, X. Tang, D. Cao, and T. Liu, "Decision-making for autonomous vehicles in random task scenarios at unsignalized intersection using deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 6, pp. 7812–7825, 2024.
[14] S. Paternain, L. Chamon, M. Calvo-Fullana, and A. Ribeiro, "Constrained reinforcement learning has zero duality gap," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
[15] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *International conference on machine learning*. PMLR, 2017, pp. 22–31.
[16] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International conference on machine learning*. PMLR, 2015, pp. 1889–1897.
[17] T.-Y. Yang, J. Rosca, K. Narasimhan, and P. J. Ramadge, "Projection-based constrained policy optimization," *arXiv preprint arXiv:2010.03152*, 2020.
[18] C. Wang, S. Zhou, L. Wang, Z. Lu, C. Wu, X. Wen, and G. Shou, "Autonomous driving via knowledge-enhanced safe reinforcement learning," *IEEE Transactions on Intelligent Vehicles*, 2024.
[19] Z. Wu, B. Tang, Q. Lin, C. Yu, S. Mao, Q. Xie, X. Wang, and D. Wang, "Off-policy primal-dual safe reinforcement learning," *arXiv preprint arXiv:2401.14758*, 2024.
[20] A. Stooke, J. Achiam, and P. Abbeel, "Responsive safety in reinforcement learning by pid lagrangian methods," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9133–9143.
[21] Z. Liu, Z. Cen, V. Isenbaev, W. Liu, S. Wu, B. Li, and D. Zhao, "Constrained variational policy optimization for safe reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2022, pp. 13 644–13 668.
[22] M. Tayal, A. Singh, S. Kolathaya, and S. Bansal, "A physics-informed machine learning framework for safe and optimal control of autonomous systems," *arXiv preprint arXiv:2502.11057*, 2025.
[23] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 3387–3395.
[24] M. Lu, J. S. Gosain, L. Sang, and M. Chen, "Safe learning in the real world via adaptive shielding with hamilton-jacobi reachability," *Proceedings of Machine Learning Research vol*, vol. 283, pp. 1–14, 2025.
[25] B. Brito, A. Agarwal, and J. Alonso-Mora, "Learning interaction-aware guidance for trajectory optimization in dense traffic scenarios," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18 808–18 821, 2022.
[26] 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.
[27] C.-J. Hoel, K. Wolff, and L. Laine, "Ensemble quantile networks: Uncertainty-aware reinforcement learning with applications in autonomous driving," *IEEE Transactions on intelligent transportation systems*, vol. 24, no. 6, pp. 6030–6041, 2023.
[28] K. Yang, X. Tang, S. Qiu, S. Jin, Z. Wei, and H. Wang, "Towards robust decision-making for autonomous driving on highway," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 9, pp. 11 251–11 263, 2023.
[29] S. Thrun and A. Schwartz, "Issues in using function approximation for reinforcement learning," in *Proceedings of the 1993 connectionist models summer school*. Psychology Press, 2014, pp. 255–263.
[30] 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.
[31] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.
[32] D. G. Luenberger, Y. Ye *et al.*, *Linear and nonlinear programming*. Springer, 1984, vol. 2.
[33] W. Schwarting, J. Alonso-Mora, L. Paull, S. Karaman, and D. Rus, "Safe nonlinear trajectory generation for parallel autonomy with a dynamic vehicle model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2994–3008, 2017.
[34] E. Leurent *et al.*, "An environment for autonomous driving decision-making," 2018.
[35] A. Ray, J. Achiam, and D. Amodei, "Benchmarking safe exploration in deep reinforcement learning," *arXiv preprint arXiv:1910.01708*, vol. 7, no. 1, p. 2, 2019.
[36] S. Ha, P. Xu, Z. Tan, S. Levine, and J. Tan, "Learning to walk in the real world with minimal human effort," *arXiv preprint arXiv:2002.08550*, 2020.

Sunan Zhang received the M.S. degree in automotive engineering from Chongqing University of Technology (CQUT), Chongqing, China, in 2023. He is currently working toward the Ph.D. degree at the School of Mechanical Engineering, Southeast University, Nanjing, Jiangsu Province, China. His research interests include reinforcement learning, model predictive control, and applications in decision-making and motion control for connected and autonomous vehicles.

Bingbing Li received the Ph.D. degree in vehicle engineering from the School of Mechanical Engineering, Southeast University, Nanjing, China, in 2024. From 2022 to 2023, he was a Visiting Research Scholar at the Department of Electronic and Electrical Engineering, University College London, U.K. He is currently a assistant research fellow with the School of Mechanical Engineering, Southeast University. His current research interests include connected and automated vehicles, energy-efficient driving, and flying car.



Boli Chen (Senior Member, IEEE) received his B.Eng. in Electrical and Electronic Engineering from Northumbria University, UK, in 2010. He earned his MSc and PhD in Control Systems from Imperial College London, UK, in 2011 and 2015, respectively. He is currently a Lecturer in the Department of Electronic and Electrical Engineering at University College London (UCL), UK. His research focuses on the control, optimization, and estimation of complex dynamical systems, with rich applications in smart cities, e.g., transportation, electric energy systems, and sensor networks. Dr Boli Chen is a member of the IEEE Control Systems Society Technical Committee on "Smart Cities". He serves as an Associate Editor for the IEEE Transactions on Intelligent Transportation Systems and the European Journal of Control. Additionally, he is a member of the EUCA Conference Editorial Board and the IEEE ITSC Editorial Board.

Bo Hu was born in Hefei, Anhui Province, China, in 1989. He received his B.S. degree in automotive engineering from Chongqing University of Technology (CQUT), Chongqing, China, in 2011, and M.S. and PhD degree in automotive engineering from University of Bath, Bath, UK, in 2012 and 2016, respectively. He is currently an Associate Professor with the Key Laboratory of Advanced Manufacturing Technology for Automobile Parts, Ministry of Education, CQUT. His current research interests include machine learning based control of intelligent and connected vehicles and modeling and control of advanced boosted engine systems.

Chen Sun is currently an Assistant Professor with the Department of Data and Systems Engineering, the University of Hong Kong. He received the Ph.D. degree in Mechanical & Mechatronics Engineering from University of Waterloo, ON, Canada in 2022, M.A.Sc degree in Electrical & Computer Engineering from University of Toronto, ON, Canada in 2017 and B.Eng. degree in automation from the University of Electronic Science and Technology of China, Chengdu, China, in 2014. His research interests include field robotics, safe and trustworthy autonomous driving and in general human-CPS autonomy.

Weichao Zhuang (Member, IEEE) received the B.S. and Ph.D. degrees in mechanical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2012 and 2017, respectively. From 2014 to 2015, he was a Visiting Student with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, USA. He is currently an Associate Professor with the School of Mechanical Engineering, Southeast University, Nanjing. His current research interests include, optimal control, clean energy vehicles, connected vehicles, and multiagent control.