

Hybrid Action Based Reinforcement Learning for Multi-Objective Compatible Autonomous Driving

Guizhe Jin, Zhuoren Li, Bo Leng, Wei Han, Lu Xiong, and Chen Sun

Abstract—Reinforcement Learning (RL) has shown excellent performance in solving decision-making and control problems of autonomous driving, which is increasingly applied in diverse driving scenarios. However, driving is a multi-attribute problem, leading to challenges in achieving multi-objective compatibility for current RL methods, especially in both policy updating and policy execution. On the one hand, a single value evaluation network limits the policy updating in complex scenarios with coupled driving objectives. On the other hand, the common single-type action space structure limits driving flexibility or results in large behavior fluctuations during policy execution. To this end, we propose a Multi-objective Ensemble-Critic reinforcement learning method with Hybrid Parametrized Action for multi-objective compatible autonomous driving. Specifically, an advanced MORL architecture is constructed, in which the ensemble-critic focuses on different objectives through independent reward functions. The architecture integrates a hybrid parameterized action space structure, and the generated driving actions contain both abstract guidance that matches the hybrid road modality and concrete control commands. Additionally, an uncertainty-based exploration mechanism that supports hybrid actions is developed to learn multi-objective compatible policies more quickly. Experimental results demonstrate that, in both simulator-based and HighD dataset-based multi-lane highway scenarios, our method efficiently learns multi-objective compatible autonomous driving with respect to efficiency, action consistency, and safety.

Index Terms—Reinforcement learning, autonomous driving, motion planning, hybrid action.

I. INTRODUCTION

Reinforcement learning (RL) has good potential in solving temporal decision-making problems [1], which can learn viable and near-optimal policies for complex tasks [2]. The RL agent explores policies through interactions with the environment, enabling self-improvement [3], [4]. Therefore, RL is considered as an effective way to solve decision-making and control problems for autonomous driving (AD) [5]. It has led to widespread application in driving scenarios [6] and has outperformed human drivers in certain tasks [7].

However, current RL methods still face several limitations in achieving compatibility with key driving objectives such as safety, efficiency, and action consistency [8], [9]. In particular,

Guizhe Jin, Zhuoren Li, Bo Leng, Wei Han and Lu Xiong are with the School of Automotive Studies, Tongji University, Shanghai 201804, China. (Email: jgz13573016892@163.com, 1911055@tongji.edu.cn, lengbo@tongji.edu.cn, tianwei@foxmail.com, xiong_lu@tongji.edu.cn).

Chen Sun is with the Department of Data and Systems Engineering, University of Hong Kong, Hong Kong. (Email: c87sun@hku.hk).

This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

when addressing multi-attribute driving tasks, mainstream RL-based AD approaches exhibit shortcomings in both policy update and execution: (i) For policy updates, most rely on a single critic (value network) to evaluate and guide learning, making it difficult to efficiently explore multi-objective-compatible policies within a large and complex traffic state space; (ii) For policy execution, most approaches employ a single-type action space structure to handle hybrid road modality, which limits the policy’s ability to fully represent real driving behaviors and forces a trade-off among certain objectives.

In terms of policy update, employing a single critic (i.e., a single reward function) to evaluate policy performance fails to capture the strong coupling and potential conflicts among driving objectives. When multiple attributes of an AD task are combined into a single reward function, the agent may disproportionately focus on certain attributes during training [8]. As a result, some objectives may be neglected in specific states, leading to inaccurate value estimation and suboptimal policy performance. This can cause the agent to behave in ways that are misaligned with multi-objective expectations, such as becoming overly aggressive to maximize speed or excessively conservative to ensure safety. In contrast, multi-objective reinforcement learning (MORL) addresses this issue more effectively by constructing reward function vectors [10], enabling better compatibility among multiple objectives. Furthermore, complex traffic state spaces demand efficient exploration during policy updates. Most existing RL-based AD methods rely on random exploration, which prevents the agent from actively seeking out unknown regions and discovering potentially effective policies [11]. This random exploration often results in the collection of redundant experiences that contribute little to policy improvement, leading to inefficient convergence or even entrapment in local optima.

For policy execution, using a single-type action space to generate either abstract or concrete driving behaviors constrains RL agents to discrete actions that lack flexibility or continuous actions that lack consistency. A common approach is to have the agent produce discrete, long-term driving goals, such as semantic decisions [12] or target points for path planning [13]. However, because the agent does not directly control the vehicle’s motion, its ability to adapt driving behavior flexibly is limited. While this long-term planning enhances action consistency, it reduces responsiveness to dynamic changes. Conversely, directly outputting short-term control commands [14] allows for greater flexibility, but often results in less consistent behavior, with frequent fluctuations and abrupt reactions to environmental changes.

To alleviate the limitations of policy updating and execution in typical multi-objective AD tasks, this paper proposes a **Multi-objective Ensemble-Critic** reinforcement learning method with **Hybrid Parametrized Action space (HPA-MoEC)** for multi-objective compatibility. The HPA-MoEC adopts a MORL architecture focused on AD tasks. By defining multiple reward functions to decouple different driving attributes, each reward function guides an ensemble-critic to focus on specific driving objectives, thereby assisting the actor (policy network) in learning multi-objective compatible driving behaviors. The architecture further integrates a hybrid parameterized action space structure containing a discrete action set and its corresponding continuous parameters, which together generate driving actions that combine abstract guidance and concrete control commands. Additionally, uncertainty estimates from the ensemble-critic guide the agent to explore promising driving policies, facilitating more efficient exploration in unknown environments. Evaluation results on both simulation and HighD dataset-based multi-lane highway scenarios demonstrate that HPA-MoEC efficiently learns multi-objective compatible driving behaviors, significantly improving driving efficiency, action consistency, and safety. The main contributions are summarized as follows:

- 1) A MORL architecture compatible with multiple AD objectives is proposed, in which ensemble-critic focuses on a distinct objective using separate reward functions. Considering the safety-critical nature of AD, two driving objectives are defined and evaluated with two ensemble critics: one targeting overall performance, including interactivity, and the other dedicated to safety. By isolating the safety objective, the effectiveness of our architecture is demonstrated through improved safety performance in experimental results.
- 2) A hybrid parameterized action space structure is designed to combine finer-grained guidance and control commands to adapt to hybrid road modality. This hybrid action space consists of discrete actions and their corresponding continuous parameters, which together generate both abstract guidance and concrete control outputs. Our design achieves greater driving flexibility and reduced behavioral fluctuations, ensuring compatibility between driving efficiency and action consistency.
- 3) An epistemic uncertainty-based exploration mechanism is developed to enhance learning efficiency and complement the hybrid action space structure. By dynamically adjusting the direction and magnitude of exploration according to uncertainty and its trends, the agent is encouraged to more rapidly explore high-uncertainty regions for potentially effective policies. This exploration mechanism significantly improves the learning efficiency of multi-objective compatible policies.

The remainder of this paper is organized as follows: Section II reviews related work, while Section III outlines the methodology. Specific implementation details are presented in Section IV. Section V discusses the experimental results, and the conclusions are provided in Section VI.

II. RELATED WORKS

The AD task involves making complex sequential decisions in a dynamic environment and can therefore be modeled as Markov Decision Processes (MDPs) [15]. The MDP is commonly represented as a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \gamma \rangle$, where \mathcal{S} is the state space, \mathcal{A} is the action space, \mathcal{R} is the reward function, \mathcal{T} is the transition function, and γ is the discount factor. At time t , the RL agent selects action $a_t \in \mathcal{A}$ based on state $s_t \in \mathcal{S}$, then receives reward $r_t \in \mathcal{R}$ from the environment and transitions to state s_{t+1} according to \mathcal{T} . The goal of the agent is to find an optimal policy through trial-and-error to maximize the expected reward.

A. Multi-Objective Policy Evaluation

For AD problems, multiple attitudes should be considered, requiring the policy to be multi-objective compatible. The objectives are sometimes conflicting, like safety and driving efficiency [6]. The most common design is to linearly combine all attributes into a single, additive reward function for policy evaluation [16], typically based on mainstream RL algorithms such as Deep Q-Network (DQN) [17] and Soft Actor-Critic (SAC) [18]. Specifically, the weights of this linearly expressed reward function are typically determined through manual design after multiple trial-and-error iterations [19], or by applying Inverse-RL to human demonstrations [20]. However, policy evaluation under this linear assumption may be inaccurate because the highest rewarding action may not be the one that enables multi-objective compatible driving [10], [21], leading to reduced policy performance [22]. Additionally, a single critic representing multiple attitude rewards forces the learning of value coherence, which may not accurately reflect the true critic and degrade policy quality [23].

The MORL has recently attracted significant attention for its ability to address complex decision-making problems involving multiple, often conflicting, objectives [21]. A typical MORL approach employs an architecture with multiple critics to enable multi-objective compatible policy updates [10], [24]–[26]. Specifically, key attributes are separated from a single reward function by defining multiple reward functions, with each attribute treated as an independent evaluation objective [10]. Several AD-related studies have demonstrated the advantages of MORL in driving tasks by incorporating objectives such as safety [24], efficiency [27], and comfort [25]. Additionally, [26] introduces a pre-trained safe-critic to guide the policy towards safer actions. Building on these ideas, we propose HPA-MoEC, an advanced MORL architecture that integrates a novel hybrid action space and introduces epistemic uncertainty via ensemble-critic to enhance policy exploration and learning capability. Compared to previous MORL methods, HPA-MoEC achieves better and faster multi-objective learning in hybrid road modality.

B. Action Space Structure

Many current RL-based AD methods use a single action type to control vehicle, which fail to be compatible with high driving flexibility and small behavior fluctuations. On one

hand, some studies use a discrete action space to generate abstract behavior decisions, offering long-term targets that indirectly guide vehicle control. Specifically, [28], [29] use DQN and its improved versions to generate semantic lateral actions, such as left or right lane changes. Additionally, [30], [31] introduces longitudinal discrete acceleration and deceleration actions. To provide clearer guidance, some studies select from a discrete set of trajectories [32], [33] or directly generate the positions and desired speeds of target points [34]. However, these methods often reduce the alignment between agent outputs and driving behavior, as they rely on integration with a basic controller for vehicle control, which limits flexibility. On the other hand, some studies [35], [36] directly generate steering angles laterally and accelerations longitudinally from a continuous action space, aiming to enhance flexibility. However, fluctuations in the network's output can cause frequent changes in steering angle and acceleration commands [37]. In scenarios with dedicated lanes, lateral fluctuations caused by steering angle variations will result in unpredictable paths. The experimental results in [14] provide further evidence for the existence of driving behavior fluctuations. In contrast, fluctuations in longitudinal acceleration are manageable and enable more flexible speed trajectories [38].

For compatibility of flexibility and small behavior fluctuations, several studies design hybrid actions by discretizing parts of a continuous action space [39], [40], or using a parameterized action space [14], [38], [41], which generates both lateral discrete abstract targets and longitudinal continuous concrete acceleration commands. Additionally, [42] designs a dual-layer decision-making control model that combines parallel DQN and Deep Deterministic Policy Gradient (DDPG) for hybrid output. [43] trains skill-agents for various driving objectives to output acceleration, from which DQN can flexibly select. However, the aforementioned studies fail to sufficiently integrate discrete and continuous actions, nor do they adequately account for the hybrid nature of road structures in complex driving environments. In contrast, our novel hybrid action space is specifically tailored to driving, providing abstract guidance compatible with hybrid road modality alongside continuous concrete control commands.

C. Policy Exploration Mechanism

Policy exploration helps agents discover potentially multi-objective compatible policies. A proper exploration mechanism can accelerate the learning process to converge to a viable policy faster [44]. However, the most common exploration strategy in RL is random exploration, such as ϵ -greedy in DQN [45], random noise in TD3 [46], and the maximum entropy mechanism in SAC [18]. This randomized mechanism makes policy exploration lack of orientation and leads to repeated collection of experience samples, which reduces the training efficiency [47]. Although some studies attempt to introduce reward novel state [48] or error of reward [49] to modify the exploration level, this does not change the nature of random exploration. This inefficient exploration mechanism limits the potential performance of RL policies, particularly when pursuing multi-objective compatibility in complex traffic

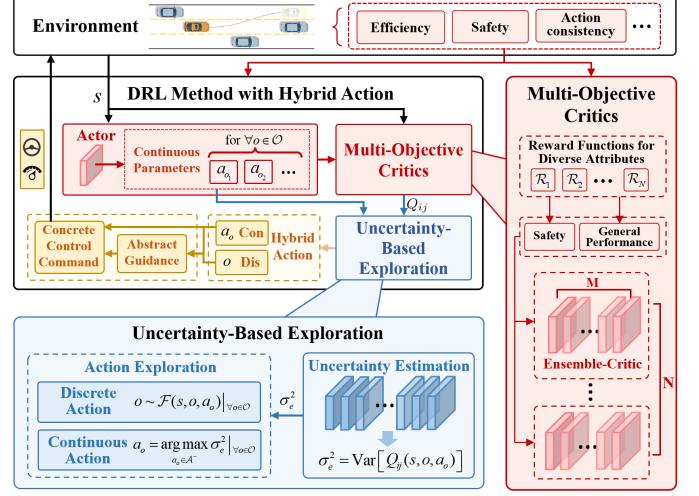


Fig. 1. The overall framework of proposed HPA-MoEC. The actor of RL Method firstly generates the continuous action parameters a_o based on states s , which are then input into the Multi-Objective Critics module along with s for evaluating the value function. This module consists of N ensemble-critics corresponding to the different attributes, and each of them is an ensemble of M critics. The Exploration Strategy module then captures epistemic uncertainty from the ensemble-critics and selects the final hybrid action (o, a_o) that enhances training efficiency.

scenarios. Some other studies [50] attempt to use reward shaping to encourage exploration, but the manually imposed rewards heavily depend on the designer's experience.

Some studies [51], [52] use model ensemble technique to capture epistemic uncertainty and select actions that encourage the agent to explore high-uncertainty areas, thus accelerating policy training. Few studies leverage epistemic uncertainty in AD tasks to improve driving policy training efficiency [53]. Therefore, this paper develops an epistemic uncertainty-based exploration mechanism with multiple ensemble-critics for hybrid actions, enabling faster learning of multi-objective policy.

III. METHODOLOGY

In this section, we will present the overall framework and specific formulation details related to the HPA-MoEC methodology.

A. Overall Framework

The method proposed in this paper is based on a hybrid parameterized action space for policy evaluation and improvement, considering multiple objectives to achieve multi-objective compatibility. Thus, the MDP can be rewritten as a new tuple $\langle \mathcal{S}, \mathcal{H}, [\mathcal{R}_1, \dots, \mathcal{R}_N], \mathcal{T}, \gamma \rangle$, where:

- \mathcal{H} represents the hybrid parameterized action space, where $\mathcal{H} = \{(o, a_o) | a_o \in \mathcal{A}_{\mathcal{O}}, \text{for } \forall o \in \mathcal{O}\}$. The o is the discrete action option selected from the discrete action option set \mathcal{O} . The a_o can be seen as the continuous action parameter corresponding to o , drawn from the continuous interval $\mathcal{A}_{\mathcal{O}}$ corresponding to \mathcal{O} .
- $[\mathcal{R}_1, \dots, \mathcal{R}_N]$ represents a set of N reward functions, where \mathcal{R}_i denotes the i -th reward function for $i \in [1, \dots, N]$.

To construct a fine-grained abstract guidance suitable for hybrid road modality, the designed hybrid action space enables the agent to simultaneously output discrete actions o and continuous action parameters a_o , ensuring optimality in both. These outputs are then used to generate both abstract guidance and concrete control commands. Specifically, lateral concrete control commands are generated by combining abstract guidance with prior knowledge, while longitudinal commands are directly derived from a_o .

In addition, the agent should consider multiple attributes of the AD task during policy evaluation and efficiently explore multi-objective compatible viable policies. Therefore, we design the Multi-objective Ensemble-Critic framework, which takes N attributes as evaluation objectives and helps agent explore in high-certainty regions. Specifically, the framework consists of N ensemble-critics, which work together for policy evaluation based on the reward functions $[\mathcal{R}_1, \dots, \mathcal{R}_N]$, each focusing on different attributes. Meanwhile, each ensemble-critic consists of M critics. The epistemic uncertainty σ_e and its change trend can be captured through ensemble-critic, which helps to orient exploration. The overall framework of the proposed HPA-MoEC method is shown in Fig. 1.

B. Policy and Value Function Representation

Under the hybrid parameterized action space, the state-action value function of the optimal policy can be described by the Bellman optimal equation as follows:

$$Q(s_t, o_t, a_{o,t}) = \mathbb{E} \left[r_t + \gamma \max_{o \in \mathcal{O}} \left\{ \sup_{a_o \in \mathcal{A}_{\mathcal{O}}} Q'(s_{t+1}, o, a_o) \right\} \right]. \quad (1)$$

HPA-MoEC consists of N ensemble-critics, each composed of M critics, resulting in a total of $N \times M$ critics for value function evaluation. Specifically, each critic can estimate the value of the action (o, a_o) in state s based on its focused attributes. Let Q_{ij} represents the optimal value function evaluated by the i -th critic corresponding to \mathcal{R}_i :

$$Q_{ij}(s_t, o_t, a_{o,t}) = \mathbb{E} \left[r_{i,t} + \gamma \max_{o \in \mathcal{O}} \left\{ \sup_{a_o \in \mathcal{A}_{\mathcal{O}}} Q_{ij}(s_{t+1}, o, a_o) \right\} \right] \quad (2)$$

where $j \in [1, \dots, M]$, $r_{i,t} = \mathcal{R}_i(s, o, a_o)$. However, finding the optimal continuous action a_o is challenging in a hybrid parameterized action space. To overcome this, we assume that the value function is fixed, meaning that for any state s and discrete action o , the a_o depends on state s . At this stage, the problem of optimizing in the continuous space becomes determining the mapping from state s to action a_o : $\mathcal{S} \rightarrow \mathcal{A}_{\mathcal{O}}$. By using a deterministic policy network $\mu(s; \theta^{\mu})$ to approximate this mapping, the continuous action a_o can be obtained, with network parameters θ_{μ} . This policy network is known as the actor. Meanwhile, a value network is employed to approximate the value function Q_{ij} , with parameters θ_{ij}^Q . Given the assumption that the value function is fixed, the MDP in the parameterized action space can be viewed as the process of exploring θ^{μ} for a given θ_{ij}^Q :

$$Q_{ij}(s, o, \mu(s; \theta^{\mu}); \theta_{ij}^Q) \approx \sup_{a_o \in \mathcal{A}_{\mathcal{O}}} Q_{ij}(s, o, a_o; \theta_{ij}^Q) \mid_{\forall o \in \mathcal{O}}. \quad (3)$$

Specifically, this process can be approximated using a two-timescale update rule [54], where the training update step size for θ_{ij}^Q is much larger than that for θ_{ij}^{μ} . Therefore, Q_{ij} can be expressed as:

$$\begin{aligned} Q_{ij}(s_t, o_t, a_{o,t}; \theta_{ij}^Q) &= \\ \mathbb{E} \left[r_{i,t} + \gamma \max_{o \in \mathcal{O}} Q_{ij}(s_{t+1}, o, \mu(s_{t+1}; \theta^{\mu}); \theta_{ij}^Q) \right]. \end{aligned} \quad (4)$$

To pursue higher returns, referring to the value network update target in DQN [17], the update target for a single critic is:

$$y_{ij,t} = r_{i,t} + \gamma \max_{o \in \mathcal{O}} Q'_{ij}(s_{t+1}, o, \mu'(s_{t+1}; \theta^{\mu'}); \theta_{ij}^{Q'}), \quad (5)$$

where, Q_{ij}' and μ' are the target networks used to assist in updating the critic and actor, with parameters $\theta_{ij}^{Q'}$ and $\theta_{ij}^{\mu'}$, respectively.

In our architecture, each critic is not updated independently. For each ensemble-critic, every critic within shares the same driving attribute. Then, all ensemble-critics collaborate to guide the actor in learning a multi-objective compatible driving policy. To ensure consistency among all critics in evaluating driving behavior, the evaluation results—both for specific attributes and overall performance—should be fed back to each critic, for updating networks. Therefore, it is necessary to construct the critic's update target from both the ensemble-critic perspective and the multi-objective compatible overall perspective. For the i -th ensemble-critic, its overall evaluation of the policy's performance under a given attribute is the expectation of the value provided by the M critics:

$$\begin{aligned} \bar{Q}_i(s_t, o, a_{o,t}) &= \mathbb{E}_{j \in [1, \dots, M]} \left[Q_{ij}(s_t, o, \mu(s_{t+1}; \theta^{\mu}); \theta_{ij}^Q) \right] \\ &= \frac{1}{M} \sum_{j=1}^M Q_{ij}(s_t, o, \mu(s_{t+1}; \theta^{\mu}); \theta_{ij}^Q) \mid_{\forall o \in \mathcal{O}}. \end{aligned} \quad (6)$$

Correspondingly, the overall target of this ensemble-critic during training can be expressed as:

$$\bar{y}_{i,t} = r_{i,t} + \gamma \max_{o \in \mathcal{O}} \bar{Q}'_i(s_{t+1}, o, a_{o,t+1}), \quad (7)$$

where \bar{Q}'_i is the expectation of all Q'_{ij} for $j \in [1, \dots, M]$, similar to Eq.(6).

In addition, the actor's outputs assign different attention to the N ensemble-critics, according to the weights ω_i . Thus, the value function for evaluating the policy's multi-objective compatibility at the overall level can be represented as follows:

$$Q_{all}(s_t, o, a_{o,t}) = \sum_{i=1}^N \omega_i \bar{Q}_i(s_t, o, a_{o,t}) \mid_{\forall o \in \mathcal{O}}, \quad (8)$$

where $\sum_i^N \omega_i = 1$. Building on this, the overall target for all critics in the HPA-MoEC can be written as:

$$y_{all,t} = r_t^{all} + \gamma \max_{o \in \mathcal{O}} Q'_{all}(s_{t+1}, o, a_{o,t+1}), \quad (9)$$

where r^{all} combines the attribute rewards based on the attention level of each ensemble-critic, i.e., $r^{all} = \sum_i^N \omega_i r_i$. The Q'_{all} is denoted by the weighted sum of \bar{Q}'_i , similar to Eq.(8).

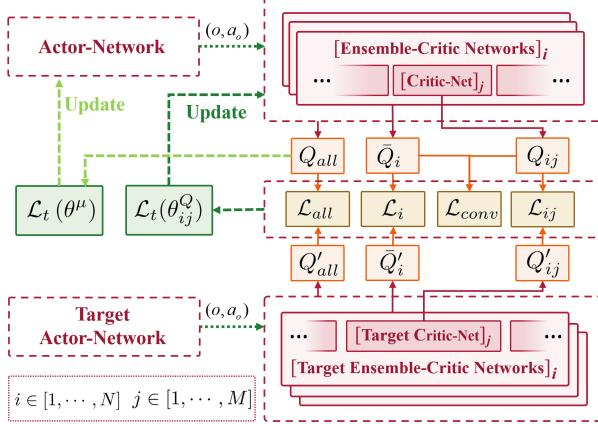


Fig. 2. The network parameter update process for actor and any critic. The target networks are soft-updated.

Thus, the update of the θ_{ij}^Q considers not only the critic's own TD error but also the average TD error of all critics in the ensemble for a given attribute, and the overall TD error of all critics. For these three aspects, corresponding loss functions are defined as follows:

$$\mathcal{L}_{ij,t}(\theta_{ij}^Q) = \frac{1}{2} [y_{ij,t} - Q_{ij}(s_t, o_t, a_{o,t}; \theta_{ij}^Q)]^2, \quad (10)$$

$$\mathcal{L}_{i,t}(\theta_{ij}^Q) = \frac{1}{2} [y_{i,t} - \bar{Q}_i(s_t, o_t, a_{o,t})]^2, \quad (11)$$

$$\mathcal{L}_{all,t}(\theta_{ij}^Q) = \frac{1}{2} [y_{all,t} - Q_{all}(s_t, o_t, a_{o,t})]^2. \quad (12)$$

To prevent any critic from significantly deviating due to random factors and disrupting policy convergence, we have added a guiding term to the loss function of the parameter θ_{ij}^Q . This helps ensure that all critics in the ensemble-critic are updated in a similar direction:

$$\mathcal{L}_{conv,t}(\theta_{ij}^Q) = \frac{1}{2} [Q_{ij}(s_t, o_t, a_{o,t}; \theta_{ij}^Q) - \bar{Q}_i(s_t, o_t, a_{o,t})]^2. \quad (13)$$

In summary, when updating the parameters θ_{ij}^Q , the final loss function account for the four aspects discussed earlier:

$$\mathcal{L}_t(\theta_{ij}^Q) = \lambda_t \cdot \mathbf{L}_t^T. \quad (14)$$

where $\mathbf{L}_t = [\mathcal{L}_{ij,t}, \mathcal{L}_{i,t}, \mathcal{L}_{all,t}, \mathcal{L}_{conv,t}]$ represents the vector of loss function and $\lambda_t = [\lambda_1, \lambda_2, \lambda_3, \lambda_4]$ is the corresponding weight vector. By backpropagating the loss defined in Eq. 14, the value network Q_{ij} can be updated iteratively.

The target for updating the actor is more straightforward, i.e., finding a multi-objective compatible optimal policy by maximizing the overall value function:

$$\mathcal{L}_t(\theta^\mu) = -\frac{1}{M} \sum_{i=1}^N \omega_i \sum_{j=1}^M \sum_{o \in \mathcal{O}} Q_{ij}(s_t, o, \mu(s_{t+1}; \theta^\mu); \theta_{ij}^Q). \quad (15)$$

Overall, the updating process of the actor's parameter θ^μ and any critic's parameter θ_{ij}^Q is shown in Fig. 2.

C. Uncertainty Estimation and Exploration Strategy

Epistemic uncertainty reflects the agent's lack of knowledge due to incomplete learning and can be captured by ensemble-critic [55]. In the i -th ensemble-critic, a larger discrepancy between the evaluation results of the critics indicates higher epistemic uncertainty about the corresponding attribute. Such discrepancies can be quantified by the variance, so the epistemic uncertainty $\sigma_{e,i}^2$ of the i -th attribute is:

$$\sigma_{e,i}^2(s, o, a_o) = \text{Var}_{j \in [1, \dots, M]} [Q_{ij}(s, o, a_o)] |_{\forall o \in \mathcal{O}}. \quad (16)$$

Considering that different attention levels are assigned to each ensemble-critic to achieve multi-objective compatibility, the weights ω_i are also used to compute the agent's total epistemic uncertainty:

$$\sigma_e^2(s, o, a_o) = \sum_{i=1}^N \omega_i \sigma_{e,i}^2(s, o, a_o) |_{\forall o \in \mathcal{O}}. \quad (17)$$

In the parameterized action space, a_o is treated as a parameter of o . Thus, the change in epistemic uncertainty for any action pair (o, a_o) can be captured by the gradient:

$$\mathcal{G} = \nabla_{a_o} \sigma_e^2(s, o, a_o) |_{\forall a_o \sim \mu(s)}. \quad (18)$$

Additionally, it is necessary to clarify that $\sigma_e^2(s, o, a_o)$ represents the epistemic uncertainty of the state-action pair for $\forall o \in \mathcal{O}$, while the overall uncertainty of the environment at state s is denoted as $\sigma_e^2(s)$. Specifically, the two are related as follows:

$$\sigma_e^2(s) = \mathbb{E} [\sigma_e^2(s, o, a_o)] |_{\forall o \in \mathcal{O}} = \frac{1}{|\mathcal{O}|} \sum_{o \in \mathcal{O}} \sigma_e^2(s, o, a_o). \quad (19)$$

Oriented by the captured epistemic uncertainty, the agent employs two different exploration strategies for the discrete action o and its corresponding continuous action a_o while exploring potentially viable policies. For continuous action, the agent's final executed a_o is determined by both the actor's output and the chosen o . Thus, the ideal continuous action exploration strategy is to solve a nonlinear continuous optimization problem: $\arg \max_{a_o \in \mathcal{A}_O} \sigma_e^2(s, o, a_o) |_{\forall o \in \mathcal{O}}$, to maximize exploration across all discrete actions o . However, solving this problem is computationally expensive and impractical for efficient policy training. Therefore, we choose a cheaper alternative by constructing a finite set of actions \mathcal{A}^- , where $\mathcal{A}^- \subset \mathcal{A}_O$, based on the actor's origin output and epistemic uncertainty gradient. This discretizes the problem of selecting high-uncertainty actions in the continuous domain:

$$\mathcal{A}^- = \left\{ a_o \middle| a_o = \text{sat}_{\mathcal{A}_O} \left[\mu(s) + \frac{k \cdot \varsigma}{K} \mathcal{G} \right], k \sim \mathcal{U}(1, K) \right\}, \quad (20)$$

$$a_o = \arg \max_{a_o \in \mathcal{A}^-} \sigma_e^2(s, o, a_o) |_{\forall o \in \mathcal{O}}, \quad (21)$$

where $\mathcal{U}(1, K)$ denotes a uniform distribution over integers from 1 to K . The ς is a coefficient that decreases with training steps, where $\varsigma \in (0, 1)$, reflecting the agent's focus on exploring actions. This simplified approach enhances continuous action exploration with low computational cost.

Similarly, the most exploratory discrete action is the one that maximizes epistemic uncertainty: $\arg \max_{o \in \mathcal{O}} \sigma_e^2(s, o, a_o)$. However, when the epistemic uncertainty of all discrete actions in the set \mathcal{O} is low, relying on epistemic uncertainty to choose actions contributes little to strategy exploration, since the agent is already confident about all actions. Thus, we define an uncertainty threshold $\sigma_{e,th}^2$ to ensure the agent adopts a greedy strategy and maximizes reward when its uncertainty is low. Additionally, since the parameters of the critic-networks are randomly initialized and their outputs may fluctuate, the estimation of epistemic uncertainty has fluctuations. We use a probabilistic approach rather than directly selecting the action with maximum uncertainty. Specifically, similar to the *Softmax* function, the probability of selecting a discrete action is based on its uncertainty value, with the total probability across all actions summing to 1. Therefore, the selection of discrete actions follows the function \mathcal{F} , where $o \sim \mathcal{F}(s, o, a_o) |_{\forall o \in \mathcal{O}}$:

$$\mathcal{F} = \begin{cases} o \sim \varepsilon(s, o, a_o) |_{\forall o \in \mathcal{O}} & \text{if } \varsigma \sigma_e^2(s) > \sigma_{e,th}^2 \\ \arg \max_{o \in \mathcal{O}} Q_{all}(s, o, a_o) & \text{else} \end{cases}, \quad (22)$$

$$\varepsilon(s, o, a_o) = \frac{e^{\sigma_e^2(s, o, a_o)}}{\sum_{o \in \mathcal{O}} e^{\sigma_e^2(s, o, a_o)}} |_{\forall o \in \mathcal{O}}, \quad (23)$$

where ε indicates the probability of choosing each action.

Based on the methods discussed above, we provide the complete algorithmic training process for our HPA-MoEC in Algorithm 1.

IV. IMPLEMENTATION

Multi-lane highway scenarios are both common and challenging, requiring driving policies that satisfy objectives such as efficiency, action consistency, and safety. This section presents the implementation details of HPA-MoEC in these scenarios, including the MDP formulation, training setup, and baseline models.

A. MDP Formulation

1) *State Space*: An appropriate state space representation is essential for effective policy learning. Specifically, the state space includes feature information about the Ego Vehicle (EV) and six surrounding vehicles (SVs) in its current and adjacent lanes:

$$\mathcal{S} \triangleq \left\{ [ID_{lane}, x, y, \varphi, v_x, v_y]^{EV}, [p_n, \Delta x_n, \Delta y_n, \varphi_n, \Delta v_{x,n}, \Delta v_{y,n}]_{n \in [1..6]}^{SVs} \right\}, \quad (24)$$

where the state of the EV in the road coordinate system consists of six variables: lane ID, longitudinal and lateral position, heading angle, and longitudinal and lateral velocity. For the n -th SV, the relevant information includes: a presence flag, longitudinal and lateral position relative to the EV, heading angle, and longitudinal and lateral velocity relative to the EV. Notably, the EV only monitors SVs within the longitudinal observation range $\Delta x \in [-80m, 160m]$.

Algorithm 1 Training process of proposed HPA-MoEC

Require: Step sizes $\{\alpha, \beta\}$, total training steps T , soft-update parameter τ , number of critics in ensemble-critic M , attribute weight ω , loss function weight λ .

1: **Initialize:** networks $\{Q_{ij}\}, \mu, \{Q'_{ij}\}, \mu'\}$ with random parameters $\{\theta_{ij}^Q, \theta^{\mu}, \theta_{ij}^{Q'}, \theta^{\mu'}\}$ for $i \in [1, \dots, N]$ and $j \in [1, \dots, M]$, replay buffer size D , exploration parameter ς .

2: **for** $t = 0$ to T **do**

3: Get state s_t from environment.

4: Capture σ_e^2 and its gradient according to Eq. (16) (18).

5: Select $a_{o,t}$ for $\forall o \in \mathcal{O}$, according to Eq. (21).

6: Select o_t according to Eq. (22).

7: Generate abstract guidance by o_t and $a_{o,t}$.

8: Generate concrete control commands for EV.

9: Get s_{t+1} and $r_{i,t}$ from environment, for $i \in [1, \dots, N]$.

10: Store $\{s_t, (o_t, a_{o,t}), [r_{1,t}, \dots, r_{N,t}], s_{t+1}\}$ into D .

11: Sample transitions randomly from D .

12: Calculate $\mathcal{L}_t(\theta_{ij}^Q)$ for each critic, according to Eq.(14).

13: Update every critic network via gradient descent:

14: $\theta_{ij,t+1}^Q \leftarrow \theta_{ij,t}^Q - \alpha_t \nabla \mathcal{L}_t(\theta_{ij}^Q)$.

15: Calculate $\mathcal{L}_t(\theta^{\mu})$ for actor, according to Eq.(15).

16: Update actor network via gradient descent:

17: $\theta_{ij,t+1}^{\mu} \leftarrow \theta_{ij,t}^{\mu} - \beta_t \nabla \mathcal{L}_t(\theta^{\mu})$.

18: Soft-update every target critic network:

19: $\theta_{ij,t+1}^{Q'} \leftarrow \tau \theta_{ij,t}^Q + (1 - \tau) \theta_{ij,t}^{Q'}$.

20: Soft-update actor:

21: $\theta_{t+1}^{\mu'} \leftarrow \tau \theta_t^{\mu} + (1 - \tau) \theta_t^{\mu'}$.

22: update $\varsigma, s_t \leftarrow s_{t+1}$.

23: **if** s_t is terminal **then**

24: Reset environment.

25: **end if**

26: **end for**

27: **return**

2) *Hybrid Parameterized Action Space*: For multi-lane scenarios with hybrid road modalities, we design explicit hybrid parameterized actions as follows: i) discrete semantic decision action b , ii) continuous parameter l for constructing a guiding path, and iii) continuous acceleration command acc . The concrete correspondence is: $b \leftarrow o, (l, acc) \leftarrow a_o$. Specifically, b is selected from a discrete set $\{LLC : -w_r, RLC : w_r, LK : 0\}$, where w_r represents the road width, with LLC and RLC representing left and right lane-change, respectively, and LK indicating lane-keeping. Considering the vehicle kinematic model [16], the value range for l is defined as follows:

$$l \in \left[\min \left(\sqrt{4R_0 w_r - w_r^2}, \frac{v_x^2}{2acc_{\max}^-} \right), e^{|v_x| + w_r} \right], \quad (25)$$

where R_0 and acc_{\max}^- represent the minimum turning radius and maximum braking acceleration of the EV. In addition, the range of the acceleration command acc is $[-3m/s^2, 3m/s^2]$.

At each time step t , with (b_t, l_t, acc_t) output by the agent, the positions of the guiding path points can be generated using

a polynomial curve-based formula:

$$y_{0,t+h} = \sum_{m=0}^5 \gamma_m x_{0,t+h}^m, \text{ where } h \in [1, \dots, H_p], \quad (26)$$

where (x_{t+h}, y_{t+h}) represents the position of the point at time step $t + h$, and H_p is the planning horizon. The coefficients γ_m of the polynomial curve can be obtained by solving a system of linear equations. Specifically, the EV's position and heading at the starting point are known, while the heading at the end point can be obtained from the road information [6]. Actually, the guiding path is determined by the selection of its endpoint position (x_{t+H_p}, y_{t+H_p}) , which is derived from the RL agent's output, where $x_{t+H_p} = l_t$ and $y_{t+H_p} = b_t$. As the guiding path generated, the EV's steering angle command δ is output using prior knowledge, specifically the Stanley algorithm in this paper. Finally, both the steering angle δ and the acceleration acc are used together for EV driving control.

3) *Reward Function for Multi-Objective*: Since safety is the fundamental requirement for driving, safety attribute is treated as a distinct objective and a corresponding safety reward function is designed for one ensemble-critic. Other attributes are combined into a single general performance reward function for another ensemble-critic. This enhances the RL agent's compatibility with safety and general driving performance.

The safety reward function, \mathcal{R}_{safe} , focuses on safety in two aspects:

$$\mathcal{R}_{safe} = -10f_{unsafe} + 0.5\text{sat}_{[0,1]}\left[\frac{\Delta t}{t_{max}}\right], \quad (27)$$

where, f_{unsafe} is set to 1 when the EV goes off the road or collides with SVs, and 0 otherwise. To further identify potential safety risks, the safety reward function also includes the TTC (Time to Collision) metric, where Δt is the estimated time to collision between the EV and the vehicle ahead, and t_{max} is the maximum time for TTC evaluation. The values 10 and 0.5 are the weights assigned to the two aspects mentioned above, respectively.

The general performance reward function, \mathcal{R}_{gen} , incorporates considerations of efficiency, comfort, and interaction:

$$\begin{aligned} \mathcal{R}_{gen} &= \mathcal{R}_{eff} + \mathcal{R}_{comf} + \mathcal{R}_{int} \\ \mathcal{R}_{eff} &= \frac{|v - v_t|}{v_t} - \max(0, \frac{v_p - v}{v_p}) \\ \mathcal{R}_{comf} &= -0.5 \frac{|\delta|}{|\delta_{max}|} - 0.5 \frac{|acc|}{|acc_{max}|} \\ \mathcal{R}_{int} &= -0.1 \sum_{n=1}^6 \frac{|acc_n^{SV}|}{|acc_{max}|}, \end{aligned} \quad (28)$$

where \mathcal{R}_{eff} is efficiency reward, encouraging the EV to maintain a speed close to the target v_t . Meanwhile, a low-speed penalty is applied to minimize the impact of the vehicle's deceleration on overall traffic flow, with the threshold set at v_p . The \mathcal{R}_{comf} is comfort reward, related to the action consistency of steering angle and acceleration, where δ_{max} and acc_{max} denote the maximum values of the two control commands. Moreover, \mathcal{R}_{int} represents the interaction reward, penalizing



Fig. 3. Schematic diagram of multi-lane highway environments in highway-env. Green vehicles represent EVs, while blue vehicles represent SVs.

EV's interference with SV's motion while interacting with environment. The acc_n^{SV} denotes the observed acceleration of the n -th SV. The number before each item is the weight of the attention given to it.

B. Training Setup

We developed a three-lane structured road environment using the AD simulation platform, highway-env [56], in which the EV attempts to accomplish a multi-objective compatible driving task. A schematic diagram of the study scenario is shown in Fig. 3. Specifically, all vehicles, including the EV, are randomly placed on the three-lane road with random initial speeds. The IDM and MOBIL models are applied to control the longitudinal and lateral movements of the SVs [16]. The SVs may change lanes at appropriate times to get closer to the target speed, potentially disrupting the EV. Additionally, we use the vehicle capacity (V/C) to represent traffic congestion, setting it to 0.5 to create moderate congestion. This ensures the EV has enough space to change lanes without oversimplifying the environment.

During training, the episode ends when $f_{unsafe} = 1$, after which the environment and all vehicles are reinitialized. Each episode is capped at 200 seconds to avoid the EV operating for long periods in low-variability scenarios. Details of the hyperparameter settings used in the algorithm training are provided in Table I.

TABLE I
HYPERPARAMETERS

Para.	Item	Value
M	Number of critics in an ensemble-critic	6
ω_1, ω_2	Weights of \mathcal{R}_{safe} and \mathcal{R}_{gen}	0.4, 0.6
γ	Discount factor	0.9
α	Training step size of critic	0.01
β	Training step size of actor	0.001
λ	Weights of loss functions for critic	[0.5, 0.2, 0.2, 0.1]
K	parameters for con-action exploration	10
τ	Soft-update parameter	0.005
T	Number of steps for training	200000
ς	Exploration weight parameter	1 → 0.001
—	Number of hidden layers in critic/actor	3
—	Hidden layer size	256
—	Activation function	Tanh
—	Replay buffer size	40000
—	Sample batch size	256
—	Training optimizer	Adam

Additionally, our method is tested on 200 episodes in both the training environment and the HighD [57] real-world dataset. For testing on the HighD dataset, the trained agent controls randomly selected vehicles, while the SVs follow their predefined trajectories.

C. Comparison Models

1) *Comparison Baseline*: To comprehensively evaluate the proposed HPA-MoEC, we compare it with several widely used RL methods for the AD task. All methods share the same training and testing environments, as well as the state space. The main difference is that, unlike HPA-MoEC, the other methods couple the attributes into a single reward function: $\mathcal{R}_{base} = \omega_1 \mathcal{R}_{safe} + \omega_2 \mathcal{R}_{gen}$. More importantly, the action spaces structure and policy exploration strategies in the following methods differ:

- Deep Q-Network (**DQN**) [17]: It only generates discrete semantic decisions and is paired with a PID controller to control the EV. The exploration strategy used is ϵ -greedy.
- SAC with Continuous actions (**SAC-C**) [18]: It only outputs continuous control commands, which are lateral steering angle and longitudinal acceleration. Its exploration is enhanced through maximum entropy and the addition of Gaussian noise to the actions.
- SAC with Hybrid actions (**SAC-H**) [18]: SAC-H discretizes part of the continuous action space in SAC, producing outputs similar to HPA-MoEC.
- PPO with Hybrid actions (**PPO-H**) [58]: An on-policy actor-critic algorithm with action space similar to SAC-H.

To ensure fair comparisons and reliable conclusions, all methods use the same network architecture, learning rate, and other key hyperparameters. Additionally, for method-specific parameters, we perform extensive tuning within reasonable ranges and select the optimal configuration for each method.

2) *Ablation Model*: To further validate the effectiveness of the three key techniques used in HPA-MoEC: i) Hybrid parametric action space structure; ii) Multi-critic policy evaluation architecture; and iii) Epistemic uncertainty-based policy exploration, we design the following ablation baselines:

- **HPA-MoEC**: The method proposed in this paper includes all three technical components.
- **HPA-Mo**: By removing component iii from HPA-MoEC, policy exploration is no longer oriented by uncertainty. Policy evaluation for each objective is performed by a single critic only, rather than by an ensemble-critic.
- **HPA**: By removing component ii from HPA-Mo, only one overall objective remains, with one corresponding critic for evaluating policies that considers multiple attributes. In fact, this baseline is similar to the algorithm in [16].
- **DA-Mo**: By removing component i from HPA-Mo, this baseline generates only coarse-grained discrete semantic decisions as abstract guidance, which are combined with the PID controller to output steering angles. In fact, this baseline is similar to part of the work in [21].

D. Evaluation Metrics

To evaluate the driving performance of the proposed method across multiple objectives, we used several metrics for each episode:

- Average Reward (**AR**): AR is the ratio of total reward to episode length, offering a comprehensive evaluation of the RL agent's performance.

- Collision Rate (**CR, %**): Collisions result from hazardous driving behavior and can be used to evaluate the safety of the agent's driving policy.
- Average Speed (**AS, m/s**): The EV's speed indicates the agent's ability to intelligently execute lane changes actions to enhance driving efficiency.
- Number of Lane-change (**NL**): NL partially reflects the EV's flexibility and can be analyzed alongside AS to explain the reasons for improved driving efficiency.
- Variance of Steering angle (**VS, rad²**) and Acceleration (**VA, m²/s⁴**): VS and VA respectively indicate the vehicle's fluctuations in lateral and longitudinal behavior, reflecting the consistency of the driving policy's actions.

V. RESULTS AND DISCUSSIONS

A. Training Performance

The learning curves for general performance and safety during training are shown in Fig. 4, with each algorithm trained six times using different seeds. The total reward curve and corresponding variance distribution in Fig. 4(a) show that the our HPA-MoEC achieves higher rewards with smaller policy fluctuations. This indicates that, regardless of seed variations, its policy consistently converges to the best general performance. By comparison, the similar rewards achieved by SAC-H and PPO-H indicate that both of them perform worse than HPA-MoEC. Furthermore, without finer-grained guiding paths, the reward during SAC-C convergence is much lower, indicating poorer driving performance when both longitudinal and lateral direct control commands are output together. Using only semantic decision actions, the DQN receives the lowest reward, indicating that discrete actions alone are insufficient for complex driving tasks.

Additionally, once the minimum sample size required for training is gathered in the experience replay pool, the policy improvement speed of HPA-MoEC is significantly faster than that of all the baselines. This increase in training efficiency is attributed to the introduction of an epistemic uncertainty-based exploration strategy, which enables a oriented and faster exploration of potentially viable policies. Notably, since SAC-C directly controls the EV by outputting steering angle commands, it often veers off the road and ends the episode early, causing the reward curve to differ significantly from other methods.

As shown in Fig. 4(b), the change in CR for each method during training is illustrated, with the zoomed-in view of the converged curves highlighting that HPA-MoEC ultimately maintains a low CR. Thanks to the decoupling of the safety objective from the general performance objective within the multi-objective policy evaluation architecture, the agent places greater emphasis on safety. In contrast, SAC-H and PPO-H have slightly higher CRs, whereas DQN has the highest. Notably, although SAC-C performs poorly in total reward, it prioritizes the safety of the EV by maintaining a very low CR. This results from its conservative following behavior, which will be discussed in detail in Section V-B1.

B. Testing Performance

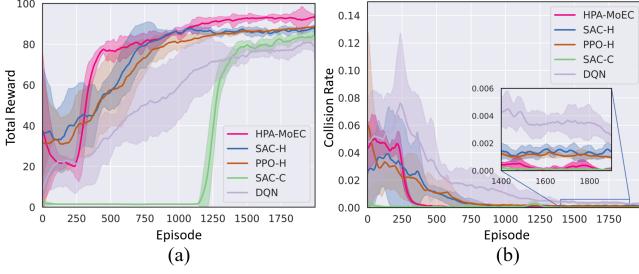


Fig. 4. The training process of our method with comparison methods quantified by: a) Total Reward and b) Collision Rate.

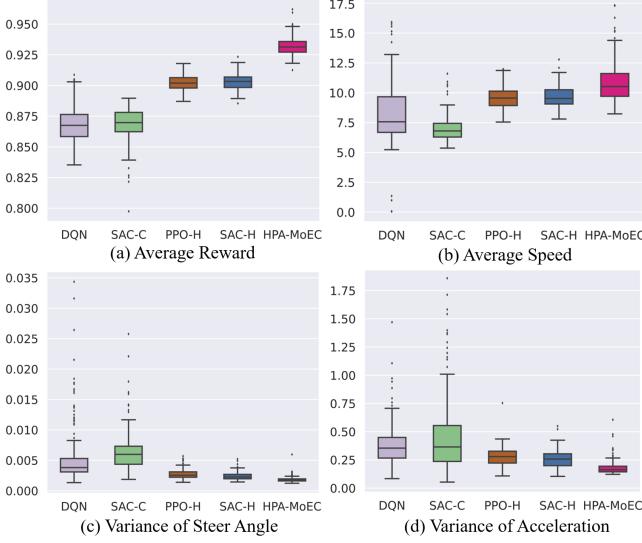


Fig. 5. Metrics distribution of testing with Rule-Based SVs: (a) average reward, (b) average speed, (c) variance of steering angle, (d) variance of acceleration.

TABLE II
TEST RESULTS WITH RULE-BASED SVS

Method	AR	AS	NL	VS	VA	CR
DQN	0.860	8.18	7.71	0.0055	0.381	0.38%
SAC-C	0.868	6.95	2.04	0.0063	0.452	0.01%
PPO-H	0.902	9.57	5.76	0.0027	0.279	0.13%
SAC-H	0.903	9.62	5.57	0.0024	0.256	0.12%
HPA-MoEC	0.932	10.87	7.14	0.0019	0.181	0.04%

1) *Testing with Rule-Based SVs:* The boxplots in Fig. 5 illustrate the distribution of four metrics in testing: average reward (Fig. 5(a)), average speed (Fig. 5(b)), and the variance of steering angle and acceleration (Fig. 5(c) and Fig. 5(d)). The quantitative statistics for all metrics are provided in Table II. Specifically, the driving policy of the proposed HPA-MoEC demonstrates advantages in driving efficiency, action consistency, and safety.

In general, HPA-MoEC receives the highest AR, which is consistent with the training results and indicates a more effective driving policy. SAC-H and PPO-H also perform well, with similar AR levels. In contrast, the driving policy of DQN and SAC-C perform poorly and exhibit considerable fluctuation, with lower and more dispersed AR values.

For driving efficiency, HPA-MoEC achieves the highest

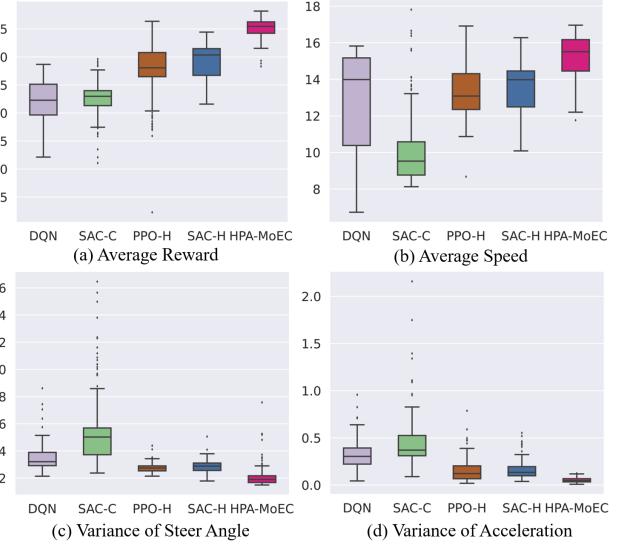


Fig. 6. Metrics distribution of testing in HighD dataset.

TABLE III
TEST RESULTS IN HIGHD DATASET

Method	AR	AS	NL	VS	VA	CR
DQN	0.909	12.74	8.84	0.0035	0.316	0.29%
SAC-C	0.913	10.12	1.25	0.0054	0.451	0.00%
PPO-H	0.938	13.32	5.67	0.0028	0.155	0.04%
SAC-H	0.945	13.58	5.69	0.0029	0.160	0.05%
HPA-MoEC	0.976	15.27	7.03	0.0021	0.051	0.01%

AS through more flexible lane changes. In comparison, SAC-H and PPO-H have lower ASs due to reduced lane-changing flexibility, leading to suboptimal efficiency. Specifically, compared to SAC-H, HPA-MoEC improves AS by 13% and increases NL by 28%. Compared to the above methods with hybrid actions, SAC-C's direct control of the EV results in the lowest AS and the fewest NL, indicating its inability to effectively leverage lane-changing opportunities to increase speed. Relying on discrete actions, DQN achieves higher AS in some episodes by frequent lane changes, but its overall AS ranks second to last. Overall, the hybrid actions provide greater flexibility and thus improve driving efficiency, especially by using a parameterized action space to generate outputs rather than discretizing part of the continuous actions.

For action consistency, HPA-MoEC exhibits the smallest VS and VA, implying a significant reduction in lateral and longitudinal driving behavior fluctuations. In comparison, although PPO-H and SAC-H also generate hybrid actions, their VS increases by 26% and 42%, respectively, while their VA increases by 41% and 54%, respectively. This indicates that the HPA-MoEC generates smoother guiding paths and acceleration commands through its parameterized action space. Notably, both SAC-C and DQN exhibit large VS and VA, indicating large behavior fluctuations. For DQN, the discrete decision set hampers smooth steering adjustments during lane changes and restricts acceleration flexibility. For SAC-C, the coupling between steering angle and acceleration commands makes it extremely challenging to produce smooth and regular

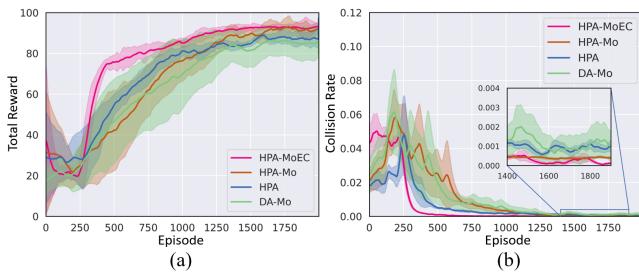


Fig. 7. The training process of our framework with ablation baselines quantified by: a) Total Reward and b) Collision Rate.

outputs when both exhibit fluctuations.

For safety performance, HPA-MoEC demonstrates the second-lowest CR, trailing only to SAC-C, highlighting its strong focus on safety. This is facilitated by a policy evaluation design with safety attribute as a separate objective, achieving a CR reduction of 67% and 69% for HPA compared to SAC-H and PPO-H, respectively. Notably, since SAC-C directly outputs control commands through a single continuous action space, it learns an over-conservative driving policy. Although this extreme concern for short-term safety significantly reduces CR, it greatly sacrifices efficiency and action consistency. In contrast, the slight compromise in safety offered by HPA-MoEC brings significant improvements in efficiency and action consistency, which is more aligned with the multi-objective requirements of AD. Additionally, DQN has a CR of 0.38%, much higher than other methods. With an average of 7.73 NL per episode, this indicates that its more aggressive driving policy increases the risk of putting the EV in danger.

2) *Testing in HighD-Dataset*: The testing results on the HighD dataset, including the distribution of evaluation metrics and quantitative statistics, are shown in Fig. 6 and Table III, respectively. Compared to the constructed simulation scenario, the traffic density in HighD is sparser, and all methods demonstrate better driving performance. Clearly, HPA-MoEC still achieves the highest AR, showing the good adaptability of its driving policy. It also maintains excellent control over acceleration and flexible lane-changing abilities, resulting in the highest AS and the most NL, except for DQN. Additionally, the guiding path still plays a role in the reduction of vehicle behavior fluctuations, keeping the VS and VA low. In terms of safety, the emphasis on safety attributes in HPA-MoEC reduces the CR to just 0.01%. Overall, HPA-MoEC outperforms all other baselines in terms of compatibility with the objectives of driving efficiency, action consistency, and safety, offering greater potential for real-world traffic applications.

C. Ablation study

1) *Training Performance*: The changes in total reward and collision rate for all ablation baselines during training are shown in Fig. 7. It is clear that as key components of HPA-MoEC are gradually removed, the performance decreases.

For HPA-Mo, policy convergence is greatly delayed. Compared to HPA-MoEC, HPA-Mo reaches similar final rewards and slightly higher CR. However, its convergence is slower,

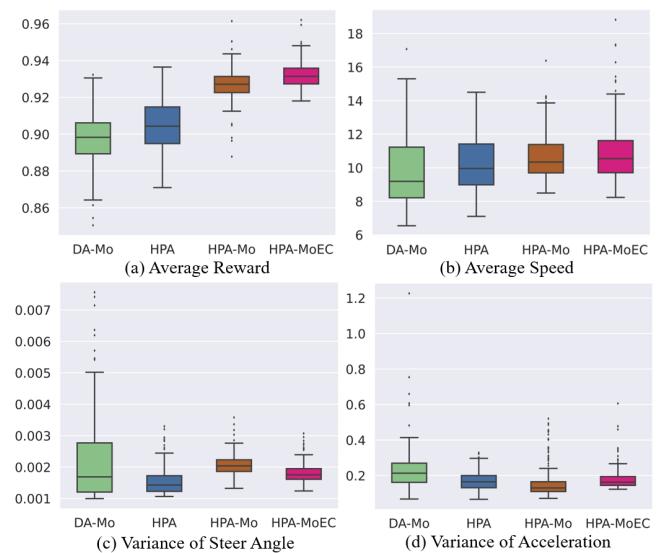


Fig. 8. Metrics distribution of ablation study with rule-based SVs.

TABLE IV
ABLATIVE STUDIES FOR HPA-MoEC WITH RULE-BASED SVS

Method	AR	AS	NL	VS	VA	CR
HPA-MoEC	0.932	10.87	7.14	0.0019	0.181	0.04%
HPA-Mo	0.927	10.63	6.90	0.0020	0.160	0.03%
HPA	0.905	10.36	6.11	0.0016	0.175	0.08%
DA-Mo	0.898	9.22	6.43	0.0025	0.185	0.06%

only reaching around the 1700th episode. In contrast, HPA-MoEC, despite involving more networks, converges around the 1400th episode, suggesting that epistemic uncertainty-based policy exploration improves training efficiency by about 18%.

For HPA, it shows lower rewards and higher CR at convergence compared to HPA-Mo. This suggests that the designed multi-objective compatible policy evaluation architecture is effective. Utilizing critics that specifically target general driving attributes and safety during policy evaluation can promote driving that is compatible with general performance and safety.

For DA-Mo, the reward it can obtain when converging is the lowest, and the CR is the highest. This shows that hybrid action space structure plays an important role in improving policy execution capabilities. A finer-grained guidance path enhances the correlation between agent output and driving behavior, further improving overall policy performance and safety.

2) *Testing with Rule-Based SVs*: The results of the ablation baseline tests, including data distributions and quantitative statistics, are shown in Fig. 8 and Table IV. The HPA-MoEC, with all technology components, demonstrates the best driving performance. As components are progressively removed, the driving performance of the ablation baselines declines accordingly.

HPA-Mo, although slow in policy convergence during training, shows driving performance close to HPA-MoEC in the final testing, with only a slight reduction in AR and AS.

HPA performs worse in both general driving performance and safety, with lower AR and higher CR. Specifically, remov-

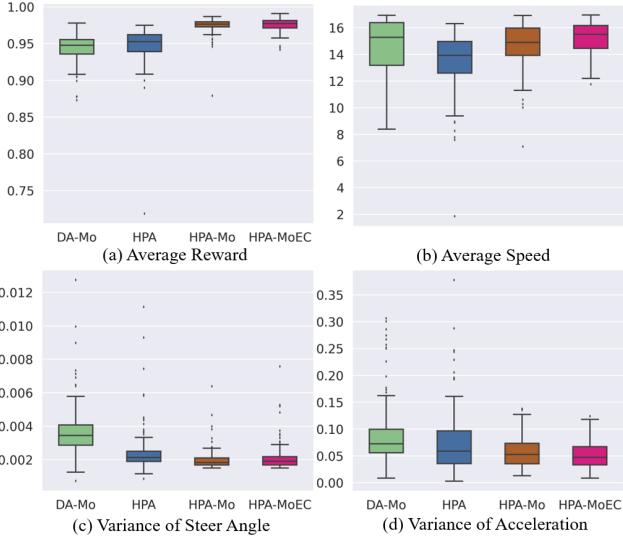


Fig. 9. Metrics distribution of ablation study in HighD dataset.

TABLE V
ABLATIVE STUDIES FOR HPA-MOEC IN HIGHD DATASET

Method	AR	AS	NL	VS	VA	CR
HPA-MoEC	0.976	15.27	7.03	0.0021	0.051	0.01%
HPA-Mo	0.975	14.71	6.88	0.0019	0.057	0.01%
HPA	0.948	13.49	5.95	0.0024	0.073	0.04%
DA-Mo	0.947	14.68	6.53	0.0036	0.076	0.04%

ing the multi-objective policy evaluation component leads to a significant decrease in AR and, more importantly, nearly a threefold increase in CR for HPA compared to HPA-Mo. This clearly demonstrates that our design maintains the compatibility of the policy with both general performance and safety during testing.

DA-Mo performs the worst across all metrics compared to the other ablation baselines. Notably, removing the hybrid action space results in approximately a 25% increase in VS compared to HPA, highlighting the larger fluctuations in lateral driving behavior. In addition, its AS decreases by 15%, with a wider distribution, while the CR increases by 100%, reflecting a decline in both driving efficiency and safety. Therefore, implementing a hybrid parameterized action space with finer-grained guidance paths helps the agent promote multi-objective driving, particularly in terms of reducing fluctuations in driving behavior.

3) *Testing in HighD-Dataset:* The testing results for all ablation baselines in the HighD dataset are shown in Fig. 9 and Table V. HPA-Mo falls slightly below HPA-MoEC in driving efficiency, but both have good driving performance. In contrast, HPA lags clearly behind both previous methods in AS and NL and has a higher CR. The DA-Mo is even worse, accompanying a notable increase in VS. This suggests that the multi-objective policy evaluation architecture and the hybrid parameterized action space with guiding paths still promote the compatibility of the objectives of driving efficiency, action consistency and safety in the HighD dataset.

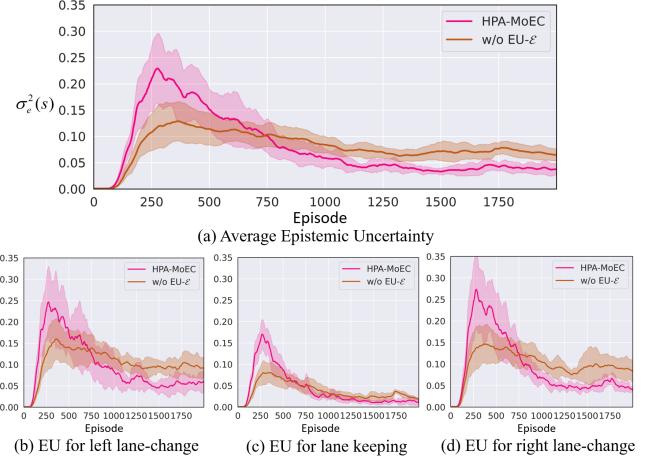


Fig. 10. Changes in epistemic uncertainty (EU) during training, including: a) average epistemic uncertainty, (b) EU for left lane-change, (c) EU for lane keeping, (d) EU for right lane-change.

D. Discussion

In summary, our HPA-MoEC method outperforms all the RL comparison baselines, where all three key technology components play a significant role in facilitating the learning of a multi-objective compatible policy. The hybrid parameterized action enhances the connection between agent actions and driving behavior by simultaneously outputting finer-grained guiding paths as well as direct acceleration commands. This action space structure promotes multi-objective compatibility, particularly enhancing action consistency by reducing driving behavior fluctuations while maintaining flexibility. The multi-objective policy evaluation architecture guides the agent in improving policy learning by treating general and safety attributes as distinct objectives and building the corresponding reward function and critic. This policy evaluation architecture improves both the driving general performance and safety, demonstrating its ability to achieve multi-objective compatible driving. In addition, the epistemic uncertainty-based policy exploration mechanism accelerates the convergence of multi-objective compatible viable policies, improving the training efficiency. It is also noteworthy that SVs in the HighD dataset exhibit human driving behaviors, which differ significantly from those in simulation traffic. Although HPA-MoEC is trained in simulated traffic, it still achieves strong driving performance when confronted with unfamiliar SVs. This demonstrates that HPA-MoEC possesses strong generalization capabilities and can effectively adapt to unfamiliar environments.

Additionally, to better observe the impact of our exploration mechanism on epistemic uncertainty, we denote 'w/o EU-E' as an attempt. In this attempt, ensemble-critics generate epistemic uncertainty but do not use it for exploration, instead performing random exploration. The curves in Fig. 10 show how epistemic uncertainty evolves throughout the policy improvement process. Our HPA-MoEC experiences higher average uncertainty in the early training phases, and then makes the uncertainty lower more rapidly during exploration. This suggests that HPA-MoEC explores more fully while converging the policy faster than randomized exploration. Further, the changes

in epistemic uncertainty for the three lane-change decisions follow a similar trend. Notably, changing lanes—whether to the left or right—results in higher uncertainty compared to lane keeping, suggesting that lane changes involve greater unknowns and risks.

VI. CONCLUSION AND FUTURE WORK

This paper proposes a Multi-objective Ensemble-Critic (HPA-MoEC) reinforcement learning method with Hybrid Parameterized Action space, capable of efficiently learning multi-objective compatible driving policies. HPA-MoEC adopts a more advanced MORL architecture, in which multiple reward functions guide ensemble-critics to focus on specific driving objectives. Meanwhile, the architecture integrates a hybrid parameterized action space structure, which can simultaneously generate abstract guidance and specific control commands that fit the hybrid road modality. In addition, an uncertainty-based exploration mechanism is developed to achieve faster learning of multi-objective compatible policies. We conduct the training and testing of the policy in both simulated traffic environments and the HighD dataset. The results show that HPA-MoEC effectively learns a multi-objective compatible autonomous driving policy in terms of efficiency, action consistency, and safety. The ablation study further demonstrated the role of technology components in HPA-MoEC in promoting multi-objective compatibility.

One limitation of our study is that the driving scenarios for training and testing are restricted to multi-lane highways. Although this typical structured road environment differs from other road types such as ramps and intersections, the driving objectives of EV in these various scenarios are generally similar: selecting appropriate behavioral goals and interacting with other vehicles. The key difference lies in how the state space is designed to enable the RL agent to comprehensively perceive the environment. Therefore, in future work, we aim to use higher-dimensional perception information (such as BEV images) as the state space to extend the application of HPA-MoEC to more complex traffic scenarios.

ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China under Grant No. 52325212 and No.52372394, in part by the National Key R&D Program of China under Grant No. 2022YFE0117100.

REFERENCES

- [1] A. Y. Majid, S. Saaybi, V. Francois-Lavet, R. V. Prasad, and C. Verhoeven, “Deep reinforcement learning versus evolution strategies: A comparative survey,” *IEEE Trans. Neural Netw. Learn. Sys.*, 2023.
- [2] Y. Zhang, B. Gao, L. Guo, H. Guo, and H. Chen, “Adaptive decision-making for automated vehicles under roundabout scenarios using optimization embedded reinforcement learning,” *IEEE Trans. Neural Netw. Learn. Sys.*, vol. 32, no. 12, pp. 5526–5538, 2021.
- [3] J. Hao, T. Yang, H. Tang, C. Bai, J. Liu, Z. Meng, P. Liu, and Z. Wang, “Exploration in deep reinforcement learning: From single-agent to multiagent domain,” *IEEE Trans. Neural Netw. Learn. Sys.*, vol. 35, no. 7, pp. 8762–8782, 2024.
- [4] Z. He, L. Dong, C. Song, and C. Sun, “Multiagent soft actor-critic based hybrid motion planner for mobile robots,” *IEEE Trans. Neural Netw. Learn. Sys.*, vol. 34, no. 12, pp. 10980–10992, 2022.
- [5] J. Xing, D. Wei, S. Zhou, T. Wang, Y. Huang, and H. Chen, “A comprehensive study on self-learning methods and implications to autonomous driving,” *IEEE Trans. Neural Netw. Learn. Sys.*, pp. 1–20, 2024.
- [6] Z. Li, G. Jin, R. Yu, B. Leng, and L. Xiong, “Interaction-aware deep reinforcement learning approach based on hybrid parameterized action space for autonomous driving,” in *Proc. SAE Intell. Connected Veh. Symposium (SAE ICVS)*, 2024.
- [7] P. R. Wurman, S. Barrett, K. Kawamoto, J. MacGlashan, K. Subramanian, T. J. Walsh, R. Capobianco, A. Devlic, F. Eckert, F. Fuchs, *et al.*, “Outracing champion gran turismo drivers with deep reinforcement learning,” *Nature*, vol. 602, no. 7896, pp. 223–228, 2022.
- [8] W. B. Knox, A. Allievi, H. Banzhaf, F. Schmitt, and P. Stone, “Reward (mis) design for autonomous driving,” *Artif. Intell.*, vol. 316, p. 103829, 2023.
- [9] Z. Zhu and H. Zhao, “A survey of deep rl and il for autonomous driving policy learning,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14043–14065, 2022.
- [10] X.-Q. Cai, P. Zhang, L. Zhao, J. Bian, M. Sugiyama, and A. Llorens, “Distributional pareto-optimal multi-objective reinforcement learning,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 15593–15613, 2023.
- [11] P. Ladosz, L. Weng, M. Kim, and H. Oh, “Exploration in deep reinforcement learning: A survey,” *Inf. Fusion*, vol. 85, pp. 1–22, 2022.
- [12] G. Li, Y. Qiu, Y. Yang, Z. Li, S. Li, W. Chu, P. Green, and S. E. Li, “Lane change strategies for autonomous vehicles: A deep reinforcement learning approach based on transformer,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2197–2211, 2023.
- [13] X. Lu, F. X. Fan, and T. Wang, “Action and trajectory planning for urban autonomous driving with hierarchical reinforcement learning,” *arXiv preprint arXiv:2306.15968*, 2023.
- [14] L. Chen, Y. He, Q. Wang, W. Pan, and Z. Ming, “Joint optimization of sensing, decision-making and motion-controlling for autonomous vehicles: A deep reinforcement learning approach,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 4642–4654, 2022.
- [15] X. Wang, S. Wang, X. Liang, D. Zhao, J. Huang, X. Xu, B. Dai, and Q. Miao, “Deep reinforcement learning: A survey,” *IEEE Trans. Neural Netw. Learn. Sys.*, vol. 35, no. 4, pp. 5064–5078, 2024.
- [16] G. Jin, Z. Li, B. Leng, W. Han, and L. Xiong, “Stability enhanced hierarchical reinforcement learning for autonomous driving with parameterized trajectory action,” in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, 2024.
- [17] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, “Human-level control through deep reinforcement learning,” *nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [18] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *Int. Conf. Mach. Learn. (ICML)*, pp. 1861–1870, PMLR, 2018.
- [19] A. Abouelazm, J. Michel, and J. M. Zoellner, “A review of reward functions for reinforcement learning in the context of autonomous driving,” *arXiv preprint arXiv:2405.01440*, 2024.
- [20] X. Wen, S. Jian, and D. He, “Modeling the effects of autonomous vehicles on human driver car-following behaviors using inverse reinforcement learning,” *IEEE Trans. Intell. Transp. Syst.*, 2023.
- [21] X. He, J. Hao, X. Chen, J. Wang, X. Ji, and C. Lv, “Robust multiobjective reinforcement learning considering environmental uncertainties,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 4, pp. 6368–6382, 2024.
- [22] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, “Concrete problems in ai safety,” *arXiv preprint arXiv:1606.06565*, 2016.
- [23] S. Mysore, G. Cheng, Y. Zhao, K. Saenko, and M. Wu, “Multi-critic actor learning: Teaching rl policies to act with style,” in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2022.
- [24] Z. Wang, S. Zhang, X. Feng, and Y. Sui, “Autonomous underwater vehicle path planning based on actor-multi-critic reinforcement learning,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 235, no. 10, pp. 1787–1796, 2021.
- [25] X. He and C. Lv, “Toward personalized decision making for autonomous vehicles: a constrained multi-objective reinforcement learning technique,” *Transp. Res. Part C Emerg. Technol.*, vol. 156, p. 104352, 2023.
- [26] K. Srinivasan, B. Eysenbach, S. Ha, J. Tan, and C. Finn, “Learning to be safe: Deep rl with a safety critic,” *arXiv preprint arXiv:2010.14603*, 2020.

- [27] K. Yang, X. Tang, S. Qiu, S. Jin, Z. Wei, and H. Wang, "Towards robust decision-making for autonomous driving on highway," *IEEE Trans. Veh. Technol.*, vol. 72, no. 9, pp. 11251–11263, 2023.
- [28] S. Nageshrao, H. E. Tseng, and D. Filev, "Autonomous highway driving using deep reinforcement learning," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, pp. 2326–2331, 2019.
- [29] S. Li, C. Wei, and Y. Wang, "Combining decision making and trajectory planning for lane changing using deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16110–16136, 2022.
- [30] P. Wolf, K. Kurzer, T. Wingert, F. Kuhnt, and J. M. Zollner, "Adaptive behavior generation for autonomous driving using deep reinforcement learning with compact semantic states," in *Proc. IEEE Intell. Veh. Symposium (IV)*, pp. 993–1000, 2018.
- [31] G. Chen, Y. Zhang, and X. Li, "Attention-based highway safety planner for autonomous driving via deep reinforcement learning," *IEEE Trans. Veh. Technol.*, 2023.
- [32] Y. Yu, C. Lu, L. Yang, Z. Li, F. Hu, and J. Gong, "Hierarchical reinforcement learning combined with motion primitives for automated overtaking," in *Proc. IEEE Intell. Veh. Symposium (IV)*, pp. 1–6, 2020.
- [33] G. Jin, Z. Li, B. Leng, and M. Shao, "Deep reinforcement learning lane-change decision-making for autonomous vehicles based on motion primitives library in hierarchical action space," *Artificial Intelligence and Autonomous Systems*, vol. 2, no. 0009, 2024.
- [34] X. Lu, F. X. Fan, and T. Wang, "Action and trajectory planning for urban autonomous driving with hierarchical reinforcement learning," *arXiv:2306.15968*, 2023.
- [35] Z. Wang, H. Huang, J. Tang, and L. Hu, "A deep reinforcement learning-based approach for autonomous lane-changing velocity control in mixed flow of vehicle group level," *Expert Syst. Appl.*, vol. 238, p. 122158, 2024.
- [36] Z. Qi, T. Wang, J. Chen, D. Narang, Y. Wang, and H. Yang, "Learning-based path planning and predictive control for autonomous vehicles with low-cost positioning," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1093–1104, 2023.
- [37] Anonymous, "Flipnet: Fourier lipschitz smooth policy network for reinforcement learning," in *Submitted to The 30th Proc. Int. Conf. Learn. Representations (ICLR)*, 2024. under review.
- [38] Q. Guo, O. Angah, Z. Liu, and X. J. Ban, "Hybrid deep reinforcement learning based eco-driving for low-level connected and automated vehicles along signalized corridors," *Transp. Res. Part C Emerg. Technol.*, vol. 124, p. 102980, 2021.
- [39] Z. Wei, P. Hao, and M. J. Barth, "Developing an adaptive strategy for connected eco-driving under uncertain traffic condition," in *Proc. IEEE Intell. Veh. Symposium (IV)*, pp. 2066–2071, 2019.
- [40] H. Liu, Z. Huang, X. Mo, and C. Lv, "Augmenting reinforcement learning with transformer-based scene representation learning for decision-making of autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 9, no. 3, pp. 4405–4421, 2024.
- [41] Y. Lin, X. Liu, and Z. Zheng, "Discretionary lane-change decision and control via parameterized soft actor-critic for hybrid action space," *Machines*, vol. 12, no. 4, p. 213, 2024.
- [42] J. Peng, S. Zhang, Y. Zhou, and Z. Li, "An integrated model for autonomous speed and lane change decision-making based on deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 21848–21860, 2022.
- [43] Y. Gurses, K. Buyukdemirci, and Y. Yildiz, "Developing driving strategies efficiently: A skill-based hierarchical reinforcement learning approach," *IEEE Control Syst. Lett.*, 2024.
- [44] Q. Liu, Y. Li, S. Chen, K. Lin, X. Shi, and Y. Lou, "Distributional reinforcement learning with epistemic and aleatoric uncertainty estimation," *Inf. Sci.*, vol. 644, p. 119217, 2023.
- [45] M. Tokic, "Adaptive ϵ -greedy exploration in reinforcement learning based on value differences," in *KI 2010: Advances in Artif. Intell.*, pp. 203–210, Springer, 2010.
- [46] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International conference on machine learning*, pp. 1587–1596, PMLR, 2018.
- [47] R. Chai, H. Niu, J. Carrasco, F. Arvin, H. Yin, and B. Lennox, "Design and experimental validation of deep reinforcement learning-based fast trajectory planning and control for mobile robot in unknown environment," *IEEE Trans. Neural Netw. Learn. Sys.*, vol. 35, no. 4, pp. 5778–5792, 2022.
- [48] M. C. Machado, M. G. Bellemare, and M. Bowling, "Count-based exploration with the successor representation," vol. 34, no. 04, pp. 5125–5133, 2020.
- [49] M. Usama and D. E. Chang, "Learning-driven exploration for reinforcement learning," in *Int. Conf. Control, Autom. Syst. (ICCAS)*, pp. 1146–1151, IEEE, 2021.
- [50] J. Wu, Z. Huang, W. Huang, and C. Lv, "Prioritized experience-based reinforcement learning with human guidance for autonomous driving," *IEEE Trans. Neural Netw. Learn. Sys.*, vol. 35, no. 1, pp. 855–869, 2024.
- [51] J. Zhang, B. Cheung, C. Finn, S. Levine, and D. Jayaraman, "Cautious adaptation for reinforcement learning in safety-critical settings," in *Int. Conf. Mach. Learn. (ICML)*, pp. 11055–11065, PMLR, 2020.
- [52] D. Kim, J. Shin, P. Abbeel, and Y. Seo, "Accelerating reinforcement learning with value-conditional state entropy exploration," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 36, 2024.
- [53] Z. Zhang, Q. Liu, Y. Li, K. Lin, and L. Li, "Safe reinforcement learning in autonomous driving with epistemic uncertainty estimation," *IEEE Trans. Intell. Transp. Syst.*, 2024.
- [54] V. S. Borkar, "Stochastic approximation with two time scales," *Systems & Control Letters*, vol. 29, no. 5, pp. 291–294, 1997.
- [55] C.-J. Hoel, K. Wolff, and L. Laine, "Tactical decision-making in autonomous driving by reinforcement learning with uncertainty estimation," in *Proc. IEEE Intell. Veh. Symposium (IV)*, pp. 1563–1569, 2020.
- [56] E. Leurent, "An environment for autonomous driving decision-making." <https://github.com/eleurent/highway-env>, 2018.
- [57] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, pp. 2118–2125, 2018.
- [58] J. Schulman, "Trust region policy optimization," *arXiv preprint arXiv:1502.05477*, 2015.