A Unified Framework with Environmental and Interaction Uncertainty for Robust Multi-Agent Reinforcement Learning

Anonymous authors
Paper under double-blind review

Abstract

Multi-agent reinforcement learning (MARL) has achieved remarkable success across diverse domains, yet its robustness remains hindered by various inherent uncertainties arising from multi-agent systems. Although previous studies have explored robustness in MARL, most of them focus on a single type of uncertainty, without a unified framework to handle multiple sources simultaneously. As a result, their methods often fail to remain robust when exposed to diverse and interacting disturbances. To address this limitation, we propose a unified framework that explicitly models two complementary sources of uncertainty: environmental uncertainty, caused by stochastic dynamics, and interaction uncertainty, arising from the unpredictable behaviors of other agents. We capture these factors using hierarchical entropy-based uncertainty sets, which are then integrated into the robust Markov game formulation. This hierarchical design enables the framework to distinguish the distinct impacts of each uncertainty source while avoiding the excessive conservatism of treating them as a single unified set. On top of this formulation, we introduce the solution concept of an Aleatoric Robust Equilibrium (ARE), where each agent optimizes its policy against worst-case scenarios derived from the hierarchical sets. To compute the ARE, we develop specialized actor–critic algorithms with theoretical convergence guarantees. Extensive experiments in both the multi-agent particle environment (MPE) and the multi-agent MuJoCo benchmark show that our approach achieves consistently superior robustness and performance across a wide range of uncertainty settings.

1 Introduction

MARL is essential for tackling complex decision-making tasks in dynamic and uncertain environments, with diverse applications such as autonomous driving (Dinneweth et al., 2022; Wu et al., 2022), robotics (Perrusquía et al., 2021; Zhong et al., 2025), strategic game-playing (Samvelyan et al., 2019), and resource allocation (Cui et al., 2020). However, real-world scenarios are inherently plagued by uncertainties stemming from stochastic dynamics, partial observability, and the non-stationarity of agent interactions (Kardeş et al., 2011; Li et al., 2014). Such uncertainties undermine the robustness and reliability of multi-agent systems, often resulting in degraded performance (Lockwood & Si, 2022). This motivates the development of robust MARL frameworks that can explicitly account for uncertainty and ensure reliable performance in complex environments.

Prior work on robust MARL has generally focused on a single source of uncertainty. On the one hand, methods involving environmental uncertainty focus on mitigating stochastic transitions or model noise to enhance policy stability under uncertain dynamics (He et al., 2023; Zhang et al., 2020b). On the other hand, methods involving interaction uncertainty focus on handling adversarial or stochastic perturbations in other agents' policies (Li et al., 2019; Muchen Sun et al., 2025). However, methods that consider only one source of uncertainty often fail in realistic multi-agent systems when both environmental and interactive perturbations occur simultaneously. This limitation underscores the need for a unified framework that jointly accounts for both types of uncertainty. Thus, we propose a unified framework that jointly models both sources of uncertainty. Fig. 1a depicts environmental uncertainty where obstacles emerge at random positions (transparent cones), requiring adaptive navigation. Fig. 1b highlights interaction uncertainty,

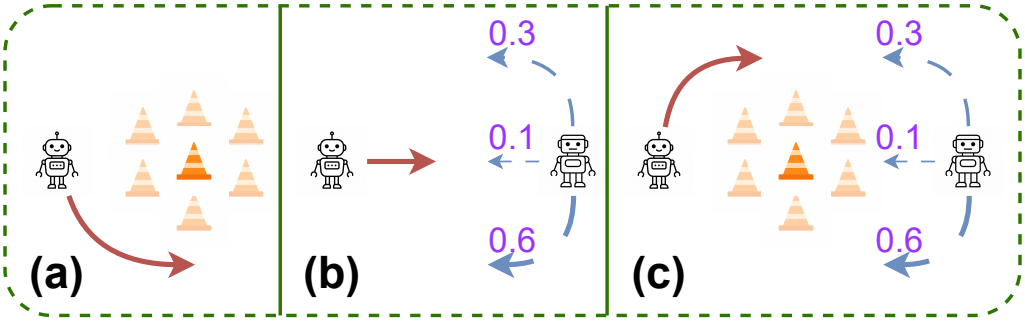

Figure 1: Illustration of two distinct sources of uncertainty in multi-agent systems. (a) Environmental uncertainty, arising from stochastic or uncertain obstacles in the environment; (b) Interaction uncertainty, induced by the stochastic and adaptive behaviors of other agents; (c) The proposed perspective considers both sources jointly, yielding a more realistic uncertainty representation for robust decision-making.

where the right-side opponent chooses its motion probabilistically (e.g., preferring the middle path because it has a lower collision probability), thereby introducing behavioral unpredictability into the interaction. Fig. 1c combines both cases, showing that in practice agents must cope with uncertain environments and adaptive opponents. This joint perspective yields a more faithful representation of multi-agent uncertainty and enables more robust solutions.

In the robust MARL, uncertainty sets are commonly used to characterize worst-case realizations under which policies must remain safe. A straightforward approach is to combine multiple sources using a "max-of-both" construction. However, this formulation is problematic: the maximum environmental disturbance and the maximum interaction deviation rarely occur at the same time, yet the construction implicitly assumes their co-occurrence. As a result, the worst-case region is exaggerated, often leading to overly conservative policies (Xu et al., 2023; Huang et al., 2022). To address this, we introduce hierarchical entropy-based uncertainty sets, where interaction uncertainty is modeled at a higher level and environmental uncertainty at a lower level. This hierarchical design preserves the causal dependency between interaction and environmental uncertainty by conditioning environmental perturbations on the worst-case perturbed policy, rather than allowing independent maximization over both sources. Consequently, it maintains robustness while mitigating the excessive conservatism inherent in the conventional "max-of-both" formulation. We summarize the main contributions of this paper as follows:

- We propose a novel framework that incorporates *environmental* and *interaction* uncertainty components, enabling more targeted and effective mitigation in multi-agent reinforcement learning.
- We introduce a hierarchical structure for constructing uncertainty sets, which preserves consistency between different sources of uncertainty and leads to less conservative strategies compared to flat unions.
- We define our solution concept as attaining aleatoric robust equilibrium, and develop tailored actor-critic algorithms with theoretical convergence guarantees. We further validate the framework through extensive experiments in diverse benchmark environments, demonstrating consistent improvements in robustness and more accurate handling of uncertainty.

2 Related Work

2.1 Aleatoric Uncertainty

Aleatoric uncertainty captures the inherent randomness in decision-making environments, such as stochastic state transitions, sensor noise, or unpredictable outcomes of actions. Unlike epistemic uncertainty, which can

be reduced with more data, aleatoric uncertainty is irreducible and thus poses a fundamental challenge for reliable reinforcement learning. Understanding and mitigating aleatoric uncertainty is particularly important in safety-critical and multi-agent systems, where such randomness can significantly undermine robustness. Recent research on robust RL under aleatoric uncertainty has mainly aimed to improve robustness and exploration efficiency (Machado et al., 2018; Moerland et al., 2017; Charpentier et al., 2022; Clements et al., 2019; Mavor-Parker et al., 2022). A variety of approaches have been proposed to handle this type of uncertainty in both exploration and exploitation. For example, Mai et al. (2022) proposes the IV-RL framework, which incorporates inverse-variance weighting to quantify aleatoric uncertainty, resulting in significant improvements in sample efficiency. Furthermore, Clements et al. (2019) develops a framework that disentangles aleatoric and epistemic uncertainties in Q-value predictions, enabling safer and more robust decision-making in risk-sensitive applications. In multi-agent systems, aleatoric uncertainty is more complex, as it stems from both the stochastic environment and the interdependent interactions among agents (Li et al., 2019; Tang et al., 2021; He et al., 2023; Amato & Oliehoek, 2013; Hao et al., 2023; Shi et al., 2024). This complexity motivates research on decomposing uncertainty into different sources, yet such work remains limited. For instance, Zhao et al. (2024) separates the uncertainty into contributions from the current agent and from other agents. To the best of our knowledge, however, existing studies do not yet provide a systematic treatment of aleatoric uncertainty that jointly accounts for environmental factors and agent interactions, leaving the development of principled decomposition methods an open challenge.

2.2 Robust Markov Games

Robust Markov game frameworks have been introduced to handle uncertainty and improve robustness (Aghassi & Bertsimas, 2006; Sharma & Gopal, 2007; Tan et al., 2019; Zhang et al., 2020a). Bukharin et al. (2024) proposes the ERNIE framework, which incorporates adversarial regularization to improve the robustness of policies in MARL. Xu et al. (2023) explores hierarchical latent variable models within the context of robust MARL to address over-conservatism in traditional robust methods. McMahan et al. (2024) investigates the computational complexity in robust Markov games with s-rectangular uncertainty and establishes equivalences between robust Nash equilibria and regularized Markov games. Ma et al. (2023) examines robustly correlated equilibria in Markov games with model uncertainty and introduces a robust decentralized V-learning algorithm that achieves polynomial sample complexity. In our work, we introduce a robust Markov game formulation that explicitly accounts for both sources of aleatoric uncertainty through hierarchical entropy-based sets.

3 Methodology

In this section, we present our robust MARL framework. We first **formulate the problem as a robust Markov game**, where uncertainties from both the environment and agent interactions are explicitly modeled. To capture these uncertainties effectively, we introduce **hierarchical entropy-based uncertainty sets**, which naturally lead to the solution concept of the ARE, which is defined as a state where each agent maximizes its return under worst-case scenarios given by the hierarchy. Building on this formulation, we then **develop tailored policy gradient and actor–critic algorithms** that can effectively compute the ARE in practice, together with the implementation details on constructing entropy-based uncertainty sets and applying them during learning.

3.1 Problem Formulation

3.1.1 Markov Games

A standard Markov game Littman (1994) is defined as a tuple:

$$\mathcal{G} = \langle \mathcal{S}, N, \{\mathcal{A}_i\}_{i=1}^N, \{R_i\}_{i=1}^N, \mathcal{P}, \gamma \rangle. \tag{1}$$

where $\mathcal{S}$ is the state space, and N is the number of agents. For each agent i, $\mathcal{A}_i$ is the action space, and the joint action space is $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$. $R_i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ represents the reward function for agent i.

$\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the state transition function, where $\Delta(\mathcal{S})$ denotes the probability distribution over the state space. And $\gamma \in (0, 1]$ is the discounting factor.

At each timestep, each agent i selects an action $a_i \in \mathcal{A}_i$ according to its policy $\pi_i \in \Pi_i$, forming a joint action $a = (a_1, \ldots, a_N) \in \mathcal{A}$ and a joint policy $\pi = (\pi_1, \ldots, \pi_N)$, where $\pi : \mathcal{S} \to \mathcal{A}$. The joint action leads to the next state s' according to the transition function $\mathcal{P}(s' \mid s, a)$. Each agent i receives an individual reward $r_i = R_i(s, a)$ and learns to maximize its expected return:

$$J_i(s, \pi) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_i(s^t, a^t) \mid s_0 = s, a^t \sim \pi, \mathcal{P}\Big], \tag{2}$$

where s_0 is the initial state.

Similarly, the value function for each agent i can be defined as:

$$\begin{aligned} V_i(s) =& \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_i(s^t, a^t) \mid s_0 = s, a^t \sim \pi, \mathcal{P}\Big], \\ =& \mathbb{E}\Big[R_i(s, a) + \gamma V_i(s')\Big]. \end{aligned} \tag{3}$$

3.1.2 Robust Markov Games

Inspired by the works (Satia & Lave Jr, 1973; Aghassi & Bertsimas, 2006; Nilim & Ghaoui, 2003), we extend standard Markov games to robust Markov games, which consider two different types of uncertainty resources. The entropy-based robust Markov Games are defined as a tuple:

$$\mathcal{G}_R = \langle \mathcal{S}, N, \{\mathcal{A}_i\}_{i=1}^N, \{\mathcal{R}_i\}_{i=1}^N, \tilde{\mathcal{P}}_s, \{\Delta_{\pi_i}\}_{i=1}^N, \gamma \rangle, \tag{4}$$

where $\mathcal{S}$, N, $\mathcal{R}_i$, and γ represent the state space, the number of agents, the reward function for agent i, and the discount factor, respectively. $\mathcal{A}_i$ denotes the action space for the agent i, where actions are sampled from the uncertainty-perturbed policy $\bar{\pi}_i$. The policy $\bar{\pi}_i$ is defined as $\bar{\pi}_i = \pi_i + \delta_i$, where $\delta_i \in \Delta_{\pi_i}$ represents the perturbation of the policy of the set of interaction uncertainties $\{\Delta_{\pi_i}\}_{i=1}^N$. In addition, $\bar{\mathcal{P}}_s$ represents the environmental uncertainty sets for the transition probabilities.

In robust Markov games, adversarial learning has been widely explored to enhance robustness, following prior works (Subramanian et al., 2023; Guo et al., 2022; Aghassi & Bertsimas, 2006). A common approach is to introduce **nature players** that actively search for worst-case realizations within the predefined uncertainty sets. By identifying these adversarial scenarios that minimize cumulative rewards, the normal agents are then trained to maximize their strategies against them, leading to a principled min–max optimization formulation.

Definition 3.1. In particular, the goal for robust Markov Games is to learn a value function $\bar{V}_i^*(s)$ by updating the Bellman equation recursively:

$$\begin{aligned} \bar{V}_i^*(s) =& \max_{\pi_i} \min_{\substack{\delta_i \in \Delta_{\pi_i} \\ \bar{\mathcal{P}}_s \in \bar{\mathcal{P}}_s}} \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_i(s^t, a^t) \mid s_0 = s, \pi, \delta_i, \bar{\mathcal{P}}_s\Big] \\ =& \max_{\pi_i} \min_{\substack{\delta_i \in \Delta_{\pi_i} \\ \bar{\mathcal{P}}_s \in \bar{\mathcal{P}}_s}} \sum_{a_i \in \mathcal{A}_i} \pi(a_i \mid s) \prod_{j \neq i} (\pi_j(a_j \mid s) + \delta_j) \Big(R_i(s, a) + \gamma \sum_{s' \in \mathcal{S}} \bar{P}(s' \mid s, a) \bar{V}_i^*(s') \Big). \end{aligned} \tag{5}$$

where $\bar{\mathcal{P}}_s$ denotes the possible transition probability that lies in the uncertainty set for the transition dynamics $\tilde{P}_s$.

Theorem 3.2 (Existence and Uniqueness of the Robust Value Function)**.** *Assume that the state space $\mathcal{S}$ and action space $\mathcal{A}$ are bounded, the uncertainty sets Δ_{π_i} and $\bar{\mathcal{P}}_s$ are compact and non-empty. And the robust value function $\bar{V}_i(s)$ satisfies the Bellman operator T:*

$$T[\bar{V}_i](s) = \max_{\pi_i} \min_{\substack{\delta_i \in \Delta_{\pi_i} \\ \bar{\mathcal{P}}_s \in \tilde{\mathcal{P}}_s}} \sum_{a_i \in \mathcal{A}_i} \pi(a_i \mid s) \prod_{j \neq i} (\pi_j(a_j \mid s) + \delta_j) \Big(R_i(s,a) + \gamma \sum_{s' \in \mathcal{S}} \tilde{\mathcal{P}}(s' \mid s,a) \bar{V}_i(s') \Big). \tag{6}$$

Under these conditions, the robust value function $\bar{V}_i^*(s)$ exists and is unique for all $i \in \mathcal{N}$ and $s \in \mathcal{S}$.

The proof of these properties is provided in Appendix A.1.

3.2 Hierarchical Entropy-based Uncertainty Sets

In the max-min optimization framework, selecting perturbations from the uncertainty sets is critical for ensuring robustness. In this section, we describe the construction of hierarchical entropy-based uncertainty sets. Specifically, these uncertainties are naturally decomposed using entropy:

$$H(s' \mid s) = \underbrace{\mathbb{E}_{a \sim \pi(a|s)}[H(s' \mid s,a)]}_{\substack{\text{Environmental} \\ \text{Uncertainty}}} + \underbrace{H(a \mid s)}_{\substack{\text{Interaction} \\ \text{Uncertainty}}}. \tag{7}$$

Eq. 7 decomposes the total conditional entropy of the next state into two complementary sources which together provide a complete characterization of aleatoric uncertainty in multi-agent systems. The first term $\mathbb{E}_{a \sim \pi(a|s)}[H(s' \mid s,a)]$, corresponds to *environmental uncertainty*, which reflects the randomness of state transitions when the current state and action are fixed. The second term, $H(a \mid s)$, corresponds to *interaction uncertainty*, which captures the stochasticity of agents' action selections given the same state.

Building on the entropy formulation in Eq. 7, we observe a natural hierarchy: the randomness in action selection directly affects the distribution of future states. This motivates us to *analyze* the uncertainty in two stages.

At the higher level, we define the **interaction uncertainty set** that models perturbations in agents' strategies. Specifically, given the nominal policy π_i, the perturbed policy $\bar{\pi}_i = \pi_i + \delta_i$ belongs to

$$\Delta_{\pi_i} = \big\{ \delta_i \;\big|\; H(a \mid s) - H_{\bar{\pi}_i}(a \mid s) \leq \rho_{\text{int}} \big\}, \tag{8}$$

where $H_{\bar{\pi}_i}(a \mid s)$ denotes the entropy of the perturbed policy, and ρ_{int} bounds the admissible deviation in action entropy.

Conditioned on the worst-case perturbed policy $\bar{\pi}$ drawn from Δ_{π_i}, we define the lower-level **environmental uncertainty set** as

$$\tilde{\mathcal{P}}_s(\bar{\pi}) = \big\{ \bar{P}(s' \mid s, \bar{a}) \;\big|\; H(s' \mid s, \bar{a}) - H_{\bar{P}}(s' \mid s, \bar{a}) \leq \rho_{\text{env}} \big\}, \tag{9}$$

where $\bar{a} \sim \bar{\pi}$, and ρ_{env} controls the permissible variation in transition entropy. The specific quantification of these entropy terms and their practical estimation methods will be detailed later in Section 3.4.

Together, these two nested sets form the overall **hierarchical uncertainty set**:

$$\mathcal{U}_H = \big\{ (\bar{P}, \delta) \;\big|\; \delta \in \Delta_{\pi_i},\ \bar{P} \in \tilde{\mathcal{P}}_s(\bar{\pi} = \pi + \delta) \big\}, \tag{10}$$

which captures the causal dependence between interaction and environmental uncertainties.

Lemma 3.3 (Subset Property)**.** *The hierarchical uncertainty set $\mathcal{U}_H$ is a subset of the general product set $\mathcal{U}_G$ where $\mathcal{U}_G$ allows nature to select worst-case interaction and environmental perturbations independently.*

$$\mathcal{U}_H \subseteq \mathcal{U}_G. \tag{11}$$

The proof can be found in Appendix A.2.

Theorem 3.4 (Performance Bound Improvement)**.** *For any policy π and state s, the worst-case value under $\mathcal{U}_H$ is no worse than that under $\mathcal{U}_G$:*

$$\inf_{\mathcal{U}_H} V_{\tilde{\mathcal{P}},\delta}(s) \;\geq\; \inf_{\mathcal{U}_G} V_{\tilde{\mathcal{P}},\delta}(s). \tag{12}$$

The proof is listed in Appendix A.3.

Together, Lemma 3.3 and Theorem 3.4 show that the hierarchical sets exclude unrealistic joint disturbances and provide stronger robustness guarantees.

With the hierarchical uncertainty sets in place, we now need to specify the corresponding solution concept. To this end, we define the **ARE**, which characterizes the equilibrium outcome of agents optimizing against the worst-case disturbances derived from these sets.

Definition 3.5. Consider the optimal policy of the nature player as $\rho^* = (\rho_1^*, \rho_2^*, \ldots, \rho_N^*)$, where ρ represents the coupling policies of $\mathcal{P}$ and δ. A joint policy $\pi^* = (\pi_1^*, \pi_2^*, \ldots, \pi_N^*)$ and ρ^* together form the ARE. If for any $s \in \mathcal{S}$ and all $i \in \mathcal{N}$, there exists a vector of value functions $V_* = (\bar{V}_1^*, \ldots, \bar{V}_N^*)$, with each $\bar{V}_i^* : \mathcal{S} \to \mathbb{R}$, satisfying:

$$(\pi_i^*(\cdot \mid s), \rho_i^*(\cdot \mid s)) \in \arg\max_{\pi_i(\cdot|s)} \min_{\rho_i(\cdot|s)} \sum_{a_i \in \mathcal{A}_i} \pi(a_i \mid s) \prod_{j \neq i} \bar{\pi}_j(a_j \mid s, \rho_j) \Big(R_i(s,a) + \gamma \sum_{s' \in \mathcal{S}} \bar{\mathcal{P}}(s' \mid s, a, \rho_i) \bar{V}_i^*(s') \Big). \tag{13}$$

Theorem 3.6 (Existence of ARE)**.** *Assume that $\mathcal{S}$ and $\mathcal{A}$ are finite sets, Δ_{π_i} and $\tilde{\mathcal{P}}_s$ are compact and convex, and the reward function $R_i(s,a)$ and transition probability $\tilde{\mathcal{P}}(s' \mid s, a)$ are bounded. Then the ARE exists.*

The proof can be found in Appendix A.4. Unlike a standard Nash equilibrium, the ARE is a robust Markov perfect Nash equilibrium, where the nature player acts as a state-wise adversary constrained by hierarchical entropy-based uncertainty sets, rather than as a strategic long-horizon opponent.

3.3 Actor-Critic Update for ARE

To achieve the ARE, it is essential to adopt effective update methods capable of finding this equilibrium. In this part, we introduce the actor-critic approach as a means of achieving ARE. Specifically, we first define a **parametric goal function** that captures the optimization of each agent under the hierarchical uncertainty sets. We then introduce **parameterized policy update rules** for both the agent and its corresponding nature players, enabling gradient-based learning in the min–max setting. Finally, we derive and implement the **actor–critic update rule**, which jointly learns the actor and the robust critic through temporal-difference learning.

3.3.1 Parametric Goal Function

Each agent $i \in \mathcal{N}$ has a policy π_i parameterized by π_{θ_i}. The corresponding adversarial strategy, which governs the perturbation of the policy, is denoted as δ_i and parameterized by w_i, i.e., δ_{w_i}. Additionally, the transition dynamics for agent i are governed by $\mathcal{P}_i$, parameterized by ξ_i, denoted as $\mathcal{P}_{\xi_i}$.

The overall parametric joint strategy, including both agents and nature players, is represented as:

$$\tilde{\pi}_\theta = (\pi_{\theta_1}, \delta_{w_1}, \mathcal{P}_{\xi_1}, \pi_{\theta_2}, \delta_{w_2}, \mathcal{P}_{\xi_2}, \ldots, \pi_{\theta_N}, \delta_{w_N}, \mathcal{P}_{\xi_N}). \tag{14}$$

We define the objective of each agent i under the joint strategy $\tilde{\pi}_\theta$ as the expected cumulative reward: $J_i(\theta) := \bar{V}^i_{\tilde{\pi}_\theta}(s_0)$. And the parametric robust value function satisfies the following recursive relationship:

$$\bar{V}^i_{\tilde{\pi}_\theta}(s) = \sum_{a_i \in \mathcal{A}_i} \pi(a_i \mid s) \prod_{j \neq i} \big(\pi_{\theta_j}(a_j \mid s) + \delta_{w_j}(a_j \mid s)\big) \Big(R_i(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{\xi_j}(s' \mid s, a) \bar{V}^i_{\tilde{\pi}_\theta}(s') \Big). \tag{15}$$

Similarly, we define the parametric Q-value function with parameters ψ_i, which satisfies the fixed-point equation:

$$Q_{\psi_i}(s,a) = R_i(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{\xi_i}(s' \mid s, a) \sum_{a' \in \mathcal{A}} \prod_{j \neq i} \big(\pi_{\theta_j}(a'_j \mid s') + \delta_{w_j}(a'_j \mid s')\big) Q_{\psi_i}(s', a'). \tag{16}$$

3.3.2 Parametric Policy Update Rules

Given the objective defined above, we now derive the corresponding policy gradient updates that enable each agent to optimize its policy. For each agent i, the policy gradient of the objective $J_i(\theta)$ with respect to the parameter θ_i is given by:

$$\nabla_{\theta_i} J_i(\theta) = \mathbb{E}_{s,a\sim\zeta_{\tilde{\pi}_\theta}}\left[\frac{\pi_{\theta_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) + \delta_{w_i}(a_i \mid s)} \nabla_{\theta_i} \log \pi_{\theta_i}(a_i \mid s) \cdot Q_{\psi_i}(s,a)\right], \tag{17}$$

where $\zeta_{\tilde{\pi}_\theta}$ is the discounted state visitation measure under the joint strategy $\tilde{\pi}_\theta$, defined as: $\zeta_{\tilde{\pi}}^{s_0}(s) = \sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s \mid s_0, \tilde{\pi})$.

For the adversarial nature player corresponding to agent i, the policy gradient with respect to the parameters ω_i and ξ_i are given by:

$$\nabla_{\omega_i} J_i(\omega) = \mathbb{E}_{s,a\sim\zeta_{\tilde{\pi}_\theta}}\left[\frac{\nabla_{\omega_i}\delta_{\omega_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) + \delta_{\omega_i}(a_i \mid s)} \cdot Q_{\psi_i}(s,a)\right]. \tag{18}$$

$$\nabla_{\xi_i} J_i(\xi) \approx \mathbb{E}_{\tau\sim(P_{\xi_i},\,\bar{\pi}_\theta)}\left[\Big(\sum_{t=0}^{T-1} \nabla_{\xi_i} \log P_{\xi_i}\big(s^{t+1} \mid s^t, a^t\big)\Big) J_i(\pi,\tau)\right]. \tag{19}$$

Here, J_i is the return for agent i, and τ represents the sampled trajectories. The detailed derivation is listed in A.5.

3.3.3 Actor-Critic Algorithm

In this setting, the critic is implemented as a parameterized Q-network that estimates the robust action-value function for each agent. Specifically, for agent i, the robust Q-function is represented as $Q_{\psi_i} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, where $\psi_i \in \mathbb{R}^d$ denotes the critic parameters. The critic is updated through temporal-difference learning (Sutton, 1988; Konda & Tsitsiklis, 1999), which provides an online estimate of the robust return under the joint policy. While the policy is updated based on the estimated Q value from the critic:

Critic:

$$\epsilon_i^t = R_i(s^t, a^t) + \gamma Q_{\psi_i^t}(s^{t+1}, a^{t+1}) - Q_{\psi_i^t}(s^t, a^t), \tag{20}$$

$$\psi_i^{t+1} = \psi_i^t + \alpha \cdot \epsilon_i^t \cdot \nabla_{\psi_i} Q_{\psi_i^t}(s^t, a^t), \tag{21}$$

where ϵ_i^t is the TD error for agent i, and $\alpha > 0$ is the learning rate for the critic.

Actor:

$$\theta_i^{t+1} = \theta_i^t + \beta \cdot \frac{\pi_{\theta_i^t}(a_i^t \mid s^t)}{\pi_{\theta_i^t}(a_i^t \mid s^t) + \delta_{w_i^t}(a_i^t \mid s^t)} \cdot \nabla_{\theta_i} \log \pi_{\theta_i^t}(a_i^t \mid s^t) \bar{Q}_{\psi_i^t}(s^t, a^t), \tag{22}$$

where $\beta > 0$ is the learning rate for the actor.

3.4 Implementation Details

In this section, we provide the implementation details for our proposed framework. Computing these entropy terms exactly in Eq. 7 is infeasible in practice due to the high-dimensional state and joint-action spaces in MARL. We therefore introduce practical estimation methods for environmental uncertainty and interaction uncertainty, which serve as the basis for constructing the corresponding uncertainty sets used during training.

To compute the environmental uncertainty $H(s' \mid s, a)$ denoted in Eq. 7, we calculate the entropy as:

$$H(s' \mid s, a) = \frac{1}{2} \log\big((2\pi e)^d \cdot |\sigma^2|\big), \tag{23}$$

where d is the dimensionality of the state space, and σ^2 is the predicted variance of the next-state distribution, to estimate the transition variance σ, we utilize a probabilistic neural network Specht (1990); Mohebali et al. (2020) to model the environmental dynamics $P(s' \mid s, a)$. This network predicts both the mean and variance of the transition dynamics, enabling us to model the stochasticity in the environment effectively. The probabilistic dynamics model is trained by minimizing the negative log-likelihood loss that captures the error in both the predicted mean and variance of the transitions:

$$\mathcal{L}_{\text{dynamics}} = \frac{1}{N} \sum_{k=1}^{N} \left[\frac{\|s'^k - \mu(s^k, a^k)\|^2}{2\sigma^2(s^k, a^k)} + \log \sigma(s^k, a^k) \right], \tag{24}$$

where $\mu(s^k, a^k)$ is the predicted mean, $\sigma(s^k, a^k)$ is the predicted variance, and N is the number of samples in the batch.

Interaction uncertainty is quantified by the entropy of the joint action distribution $H(a \mid s)$, defined as

$$H(a \mid s) = -\sum_{a \in \mathcal{A}} P(a \mid s) \log P(a \mid s), \tag{25}$$

where $P(a \mid s)$ denotes the probability of executing joint action a in state s. Since the joint action space grows exponentially with the number of agents, directly computing $P(a \mid s)$ is intractable. Following the decentralized Markov decision process framework (Oliehoek et al., 2016), we factorize the joint policy as

$$P(a \mid s) = \prod_{i=1}^{N} P(a_i \mid o_i), \tag{26}$$

where o_i is the local observation of agent i. This factorization enables decentralized computation of the joint distribution from individual agent policies. To further reduce complexity, we approximate the entropy via Monte Carlo sampling: given K sampled joint actions $\{a^k\}_{k=1}^{K}$ drawn from the factorized distribution, we estimate

$$H(a \mid s) \approx -\frac{1}{K} \sum_{k=1}^{K} \log \prod_{i=1}^{N} P(a_i^k \mid o_i). \tag{27}$$

This sampling-based estimation efficiently measures interaction uncertainty without enumerating the entire joint action space.

Further implementation details are listed at Algorithm 1.

4 Experiments

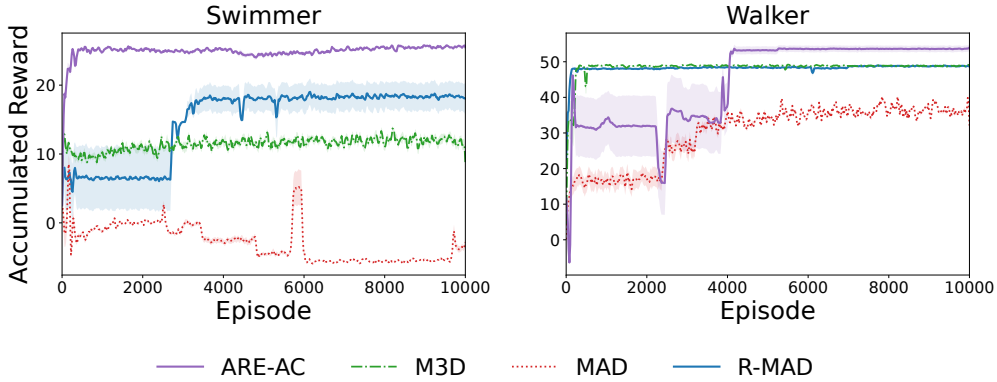

Figure 2: Accumulated reward under mid-level noise in multi-agent MuJoCo tasks.

In this section, we conduct empirical evaluations to demonstrate the validity and effectiveness of the proposed method. Specifically, we aim to answer the following questions:

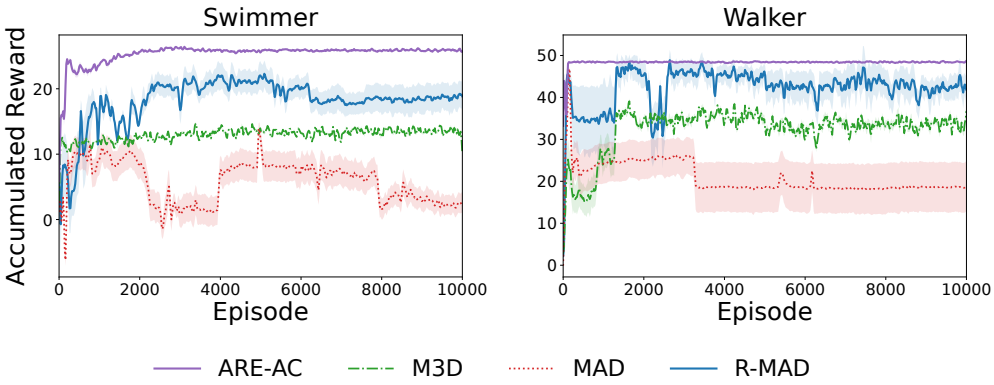

Figure 3: Accumulated reward under high-level noise in multi-agent MuJoCo tasks.

- **Performance Evaluation:** Can the proposed ARE framework ensure robust performance under different uncertainty scenarios?
- **Ablation Study:** How do different types of uncertainty and varying noise magnitudes affect policy performance? In particular, can the hierarchical uncertainty sets lead to less conservative performance?
- **Uncertainty Decomposition Analysis:** Can the ARE framework accurately tackle different sources of uncertainty?
- **Impact of Noise Levels and Settings:** How do varying noise levels and different noise settings influence policy robustness?

4.1 Environment Settings and Baselines

We evaluate our approach across benchmark environments from both the MPE environments (Lowe et al., 2017): cooperative navigation, keep-away, physical deception, predator-prey, and the multi-agent MuJoCo tasks (Peng et al., 2021): Cheetah 2×3, Cheetah 6×1, Swimmer, Walker. Detailed environment configurations are provided in Appendix B.1. To simulate different levels of uncertainty, we adjust ρ_{int} and ρ_{env} in Eq. 8 and Eq. 9, which control interaction and environmental uncertainty, respectively. These parameters do not directly set a fixed additive-noise scale; instead, they specify the radii of entropy-bounded uncertainty sets, so larger ρ_{int} or ρ_{env} permits more adversarial deviations and therefore induces stronger interaction/environmental noise. The corresponding noise levels, reward structures, and evaluation metrics are summarized in Appendix B.2.

Building on the actor–critic framework, we propose the **ARE-AC** algorithm, designed to achieve robust performance under both types of aleatoric uncertainty. We compare against three baselines. **M3DDPG (M3D)** extends MADDPG with a max-min formulation but only accounts for perturbations in agent policies (Li et al., 2019). **R-MADDPG (R-MAD)** introduces robustness against perturbations in the reward and transition models, but does not explicitly capture interaction uncertainty (Zhang et al., 2020b). Finally, **MADDPG (MAD)** serves as the standard multi-agent actor–critic baseline without robustness mechanisms (Lowe et al., 2017). For all baselines, we retain their original module designs as in the respective papers. However, to ensure fair comparison, we evaluate them under the same setting as ARE-AC, where both interaction and environmental noise are simultaneously present. In this way, although some baselines are inherently restricted to handling only one type of perturbation, our experiments highlight the advantage of ARE-AC in addressing multiple sources of aleatoric uncertainty within a unified framework.

4.2 Performance Evaluation

We first evaluate all methods in the **Cooperative Navigation** environment under mid- and high-noise conditions (Fig. 4). **ARE-AC** consistently outperforms all baselines, exhibiting faster convergence, lower

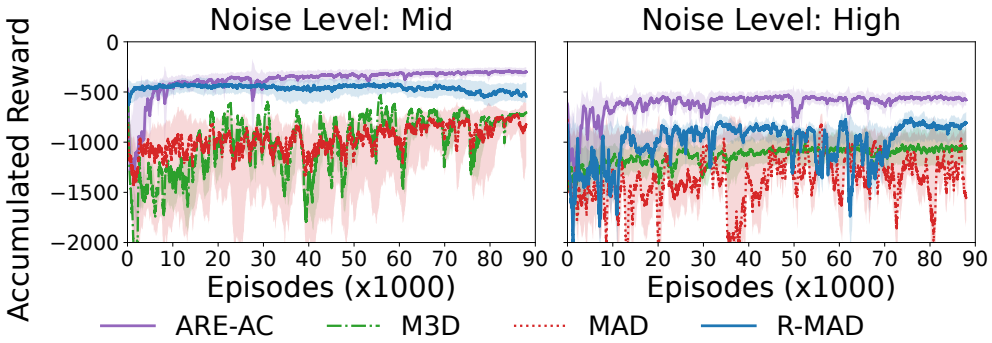

Figure 4: Cumulative reward in Cooperative Navigation under mid and high-level noise.

Algorithm	**Keep-Away**		**Physical Deception**		**Predator-Prey**		**Multiwalker**	
	Mid	High	Mid	High	Mid	High	Mid	High
ARE-AC	**2.32 ± 0.79**	**1.68 ± 0.98**	**1.34 ± 0.22**	**0.77 ± 0.11**	**-0.74 ± 0.18**	**-0.86 ± 0.21**	**0.12 ± 0.38**	**-1.65 ± 1.12**
R-MAD	1.67 ± 0.88	1.07 ± 0.63	1.17 ± 0.30	0.52 ± 0.34	-0.92 ± 0.26	-1.13 ± 0.44	-0.39 ± 0.24	-2.97 ± 0.34
M3D	1.43 ± 0.72	0.84 ± 0.45	0.83 ± 0.17	0.27 ± 0.09	-0.96 ± 0.30	-1.69 ± 0.47	-0.62 ± 0.29	-3.31 ± 0.58
MAD	1.19 ± 0.62	0.39 ± 0.07	0.67 ± 0.14	0.17 ± 0.05	-0.83 ± 0.21	-1.96 ± 0.32	-1.09 ± 0.47	-5.28 ± 0.61

Table 1: Average return (± std) under **mid** and **high** noise levels across four environments.

variance, and higher cumulative rewards. This superior performance arises from its ability to handle both *environmental* and *interaction* uncertainties simultaneously. In contrast, **M3D**, which focuses solely on policy-level perturbations, struggles to adapt when environmental transitions become stochastic. Conversely, **R-MAD** accounts only for environmental uncertainty and thus suffers from instabilities caused by interaction randomness among agents. **MAD** lacks any explicit robustness mechanism and performs the worst across all noise levels, highlighting the need for structured uncertainty modeling. We omit detailed comparisons under low noise conditions, as the perturbations are too small to significantly impact policy accuracy, resulting in comparable performance across all methods.

Beyond Cooperative Navigation, we further assess performance in the **Keep-Away**, **Physical Deception**, and **Predator–Prey** scenarios, as well as in the multi-agent MuJoCo benchmarks (Table 1, Fig. 2, and Fig. 3). Across these diverse domains, **ARE-AC** consistently achieves the best performance under both mid- and high-level noise, maintaining stable returns even when other methods degrade sharply. These results confirm that **ARE-AC** effectively unifies robustness against both environmental and interaction uncertainties, making it the most reliable framework across varying noise intensities.

4.3 Ablation Study

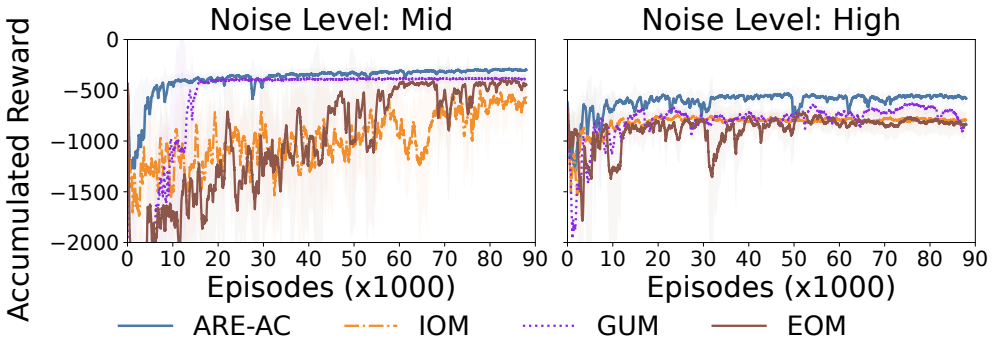

Figure 5: Ablation study of average return in Cooperative Navigation under mid and high-level noise.

Fig. 5 presents an ablation study comparing **ARE-AC** with three ablated variants under mid- and high-level noise. In these experiments, we explicitly control the construction of uncertainty sets to isolate the contributions of different components. The **General Uncertainty Model (GUM)** constructs a universal

uncertainty set by jointly optimizing over environmental and interaction disturbances, but lacks the proposed hierarchical decomposition. The **Environmental-Only Model (EOM)** retains only the environmental uncertainty set and ignores interaction-level disturbances, while the **Interaction-Only Model (IOM)** preserves only the interaction uncertainty set and discards environmental disturbances. It is noted that ARE-AC consistently outperforms all variants under both mid- and high-level noise conditions. Compared to GUM, ARE-AC's superior performance validates the benefit of the proposed hierarchical optimization, which avoids the overly conservative policies that emerge when both uncertainties are maximized simultaneously. EOM and IOM, while still formulated within our unified ARE framework, exhibit significant degradation once either type of uncertainty is excluded. This result highlights that both environmental and interaction disturbances constitute critical sources of performance degradation, and omitting either one leads to policies that fail to maintain robustness under realistic noisy conditions.

4.4 Uncertainty Decomposition Analysis

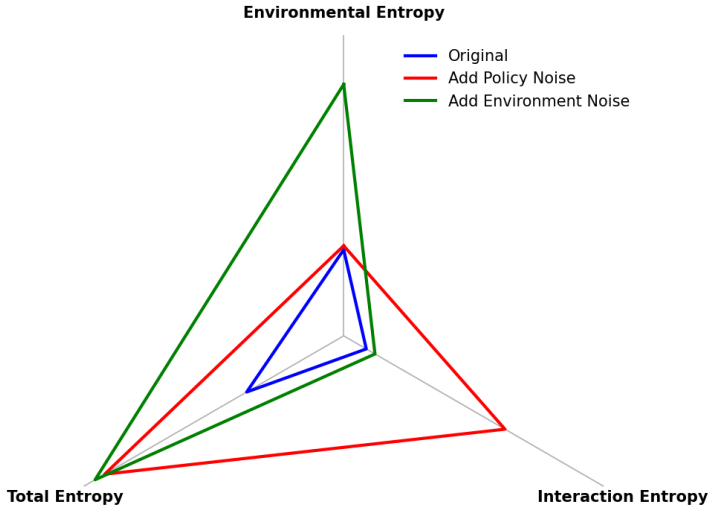

Figure 6: Radar plot of uncertainty decomposition.

To further verify whether our framework can accurately distinguish and capture the effects of environmental and interaction uncertainty introduced in Sec. 3.4, we conduct controlled experiments that selectively inject noise into different sources of uncertainty.

Specifically, we independently introduce high-level noise to either the environment (environmental noise) or the policy (interaction noise) and observe the resulting changes in entropy. As shown in Fig. 6, adding policy noise results in a significant increase in interaction entropy (Eq. 27) compared with original entropy (blue line), while leaving environmental entropy (Eq. 23) largely unchanged. Conversely, when environmental noise is applied, environmental entropy rises sharply, whereas interaction entropy exhibits minimal variation. This clear separation demonstrates that each entropy measure is primarily influenced by its corresponding noise source, confirming the accuracy of our decomposition. We further examine sensitivity by gradually increasing the noise magnitude for both components. Fig. 7 illustrates the resulting trends of interaction and environmental entropy. Interaction entropy consistently increases with interaction noise while remaining stable under environmental noise, indicating that the entropy estimator in Eq. 27 is selectively responsive to policy-level perturbations. In contrast, environmental entropy increases smoothly with environmental noise while remaining insensitive to policy perturbations, demonstrating that the estimator defined in Eq. 23 accurately captures transition-level stochasticity. Together, these results validate the effectiveness of our entropy-based quantification in disentangling and measuring the two sources of aleatoric uncertainty.

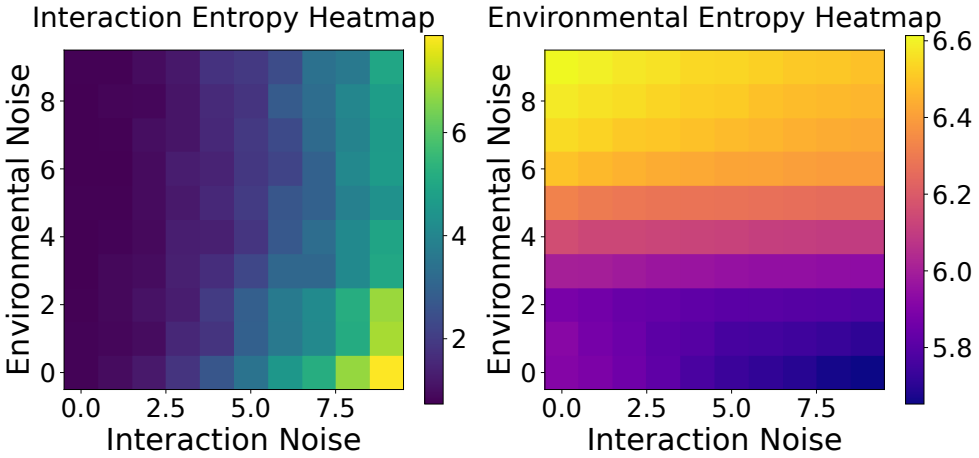

Figure 7: Heatmap visualization of uncertainty decomposition.

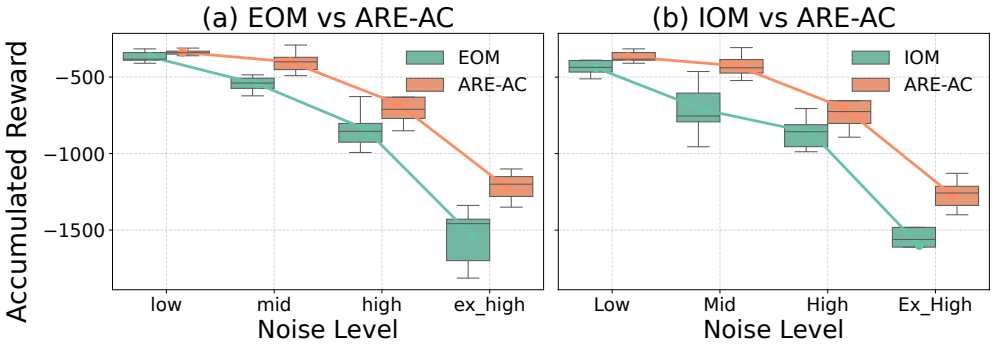

Figure 8: Performance under increasing environmental and interaction noise.

4.5 Analysis of Uncertainty Types and Noise Scaling Effects

Fig. 8 compare the performance of our full uncertainty decomposition method (ARE-AC) against two partial modeling baselines: EOM (environment-only) and IOM (interaction-only). In Fig. 8a, we gradually increase the environmental noise while keeping the interaction noise fixed. ARE-AC maintains stable performance even under extremely high disturbance levels, whereas EOM suffers a sharp degradation once the environmental variance exceeds the level captured by its single uncertainty set. Conversely, Fig. 8b varies the interaction noise, where IOM fails to compensate for the adaptive behaviors of other agents, while ARE-AC continues to yield higher and more stable returns. These results demonstrate that ARE-AC can adaptively emphasize the dominant uncertainty source in each scenario and preserve robustness.

5 Conclusion

In this work, we proposed a robust Markov game framework that explicitly models two fundamental sources of uncertainty: *environmental* and *interaction* uncertainty. By introducing hierarchical entropy-based uncertainty sets, our framework captures these uncertainties in a unified yet structured manner, allowing agents to optimize against worst-case scenarios without incurring excessive conservatism. To attain the ARE, we developed tailored actor-critic algorithms and provided theoretical guarantees of their convergence and feasibility. Comprehensive experiments across both MPE and MuJoCo multi-agent benchmarks demonstrate that our method consistently outperforms existing baselines under various noise conditions.

References

Michele Aghassi and Dimitris Bertsimas. Robust game theory. *Mathematical programming*, 107(1):231–273, 2006.

Christopher Amato and Frans A Oliehoek. Bayesian reinforcement learning for multiagent systems with state uncertainty. In *Workshop on Multi-Agent Sequential Decision Making in Uncertain Domains*, pp. 76–83. Citeseer, 2013.

Alexander Bukharin, Yan Li, Yue Yu, Qingru Zhang, Zhehui Chen, Simiao Zuo, Chao Zhang, Songan Zhang, and Tuo Zhao. Robust multi-agent reinforcement learning via adversarial regularization: Theoretical foundation and stable algorithms. *Advances in Neural Information Processing Systems*, 36, 2024.

Bertrand Charpentier, Ransalu Senanayake, Mykel Kochenderfer, and Stephan Günnemann. Disentangling epistemic and aleatoric uncertainty in reinforcement learning. *arXiv preprint arXiv:2206.01558*, 2022.

William R Clements, Bastien Van Delft, Benoît-Marie Robaglia, Reda Bahi Slaoui, and Sébastien Toth. Estimating risk and uncertainty in deep reinforcement learning. *arXiv preprint arXiv:1905.09638*, 2019.

Jingjing Cui, Yuanwei Liu, and Arumugam Nallanathan. Multi-agent reinforcement learning-based resource allocation for uav networks. *IEEE Transactions on Wireless Communications*, 19(2):729–743, 2020. doi: 10.1109/TWC.2019.2935201.

Joris Dinneweth, Abderrahmane Boubezoul, René Mandiau, and Stéphane Espié. Multi-agent reinforcement learning for autonomous vehicles: A survey. *Autonomous Intelligent Systems*, 2(1):27, 2022.

Jun Guo, Yonghong Chen, Yihang Hao, Zixin Yin, Yin Yu, and Simin Li. Towards comprehensive testing on the robustness of cooperative multi-agent reinforcement learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 115–122, 2022.

Jianye Hao, Tianpei Yang, Hongyao Tang, Chenjia Bai, Jinyi Liu, Zhaopeng Meng, Peng Liu, and Zhen Wang. Exploration in deep reinforcement learning: From single-agent to multiagent domain. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

Sihong He, Songyang Han, Sanbao Su, Shuo Han, Shaofeng Zou, and Fei Miao. Robust multi-agent reinforcement learning with state uncertainty. *Transactions on Machine Learning Research*, 2023.

Peide Huang, Mengdi Xu, Fei Fang, and Ding Zhao. Robust reinforcement learning as a stackelberg game via adaptively-regularized adversarial training. *arXiv preprint arXiv:2202.09514*, 2022.

Erim Kardeş, Fernando Ordóñez, and Randolph W Hall. Discounted robust stochastic games and an application to queueing control. *Operations research*, 59(2):365–382, 2011.

Vijay Konda and John Tsitsiklis. Actor-critic algorithms. *Advances in neural information processing systems*, 12, 1999.

Shihui Li, Yi Wu, Xinyue Cui, Honghua Dong, Fei Fang, and Stuart Russell. Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 4213–4220, 2019.

Zhongkui Li, Zhisheng Duan, and Frank L Lewis. Distributed robust consensus control of multi-agent systems with heterogeneous matching uncertainties. *Automatica*, 50(3):883–889, 2014.

Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, pp. 157–163. Elsevier, 1994.

Owen Lockwood and Mei Si. A review of uncertainty for deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 18, pp. 155–162, 2022.

Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30, 2017.

Shaocong Ma, Ziyi Chen, Shaofeng Zou, and Yi Zhou. Decentralized robust v-learning for solving markov games with model uncertainty. *Journal of Machine Learning Research*, 24(371):1–40, 2023.

Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018.

Vincent Mai, Kaustubh Mani, and Liam Paull. Sample efficient deep reinforcement learning via uncertainty estimation. *arXiv preprint arXiv:2201.01666*, 2022.

Augustine Mavor-Parker, Kimberly Young, Caswell Barry, and Lewis Griffin. How to stay curious while avoiding noisy tvs using aleatoric uncertainty estimation. In *International Conference on Machine Learning*, pp. 15220–15240. PMLR, 2022.

Jeremy McMahan, Giovanni Artiglio, and Qiaomin Xie. Roping in uncertainty: Robustness and regularization in markov games. *arXiv preprint arXiv:2406.08847*, 2024.

Thomas M Moerland, Joost Broekens, and Catholijn M Jonker. Efficient exploration with double uncertain value networks. *arXiv preprint arXiv:1711.10789*, 2017.

Behshad Mohebali, Amirhessam Tahmassebi, Anke Meyer-Baese, and Amir H Gandomi. Probabilistic neural networks: a brief overview of theory, implementation, and application. *Handbook of probabilistic models*, pp. 347–367, 2020.

Max Muchen Sun, Francesca Baldini, Katie Hughes, Peter Trautman, and Todd Murphey. Mixed strategy nash equilibrium for crowd navigation. *The International Journal of Robotics Research*, 44(7):1156–1185, 2025.

Arnab Nilim and Laurent Ghaoui. Robustness in markov decision problems with uncertain transition matrices. *Advances in neural information processing systems*, 16, 2003.

Frans A Oliehoek, Christopher Amato, et al. *A concise introduction to decentralized POMDPs*, volume 1. Springer, 2016.

Bei Peng, Tabish Rashid, Christian Schroeder de Witt, Pierre-Alexandre Kamienny, Philip Torr, Wendelin Böhmer, and Shimon Whiteson. Facmac: Factored multi-agent centralised policy gradients. *Advances in Neural Information Processing Systems*, 34:12208–12221, 2021.

Adolfo Perrusquía, Wen Yu, and Xiaoou Li. Multi-agent reinforcement learning for redundant robot control in task-space. *International Journal of Machine Learning and Cybernetics*, 12:231–241, 2021.

Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philiph H. S. Torr, Jakob Foerster, and Shimon Whiteson. The StarCraft Multi-Agent Challenge. *CoRR*, abs/1902.04043, 2019.

Jay K Satia and Roy E Lave Jr. Markovian decision processes with uncertain transition probabilities. *Operations Research*, 21(3):728–740, 1973.

Rajneesh Sharma and Madan Gopal. A robust markov game controller for nonlinear systems. *Applied Soft Computing*, 7(3):818–827, 2007.

Laixi Shi, Eric Mazumdar, Yuejie Chi, and Adam Wierman. Sample-efficient robust multi-agent reinforcement learning in the face of environmental uncertainty. *arXiv preprint arXiv:2404.18909*, 2024.

Donald F Specht. Probabilistic neural networks. *Neural networks*, 3(1):109–118, 1990.

Jayakumar Subramanian, Amit Sinha, and Aditya Mahajan. Robustness and sample complexity of model-based marl for general-sum markov games. *Dynamic Games and Applications*, 13(1):56–88, 2023.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3:9–44, 1988.

Jing-lei Tan, Cheng Lei, Hong-qi Zhang, and Yu-qiao Cheng. Optimal strategy selection approach to moving target defense based on markov robust game. *computers & security*, 85:63–76, 2019.

Bohan Tang, Yiqi Zhong, Ulrich Neumann, Gang Wang, Siheng Chen, and Ya Zhang. Collaborative uncertainty in multi-agent trajectory forecasting. *Advances in Neural Information Processing Systems*, 34: 6328–6340, 2021.

Jingda Wu, Zhiyu Huang, and Chen Lv. Uncertainty-aware model-based reinforcement learning: Methodology and application in autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 8(1):194–203, 2022.

Mengdi Xu, Peide Huang, Yaru Niu, Visak Kumar, Jielin Qiu, Chao Fang, Kuan-Hui Lee, Xuewei Qi, Henry Lam, Bo Li, et al. Group distributionally robust reinforcement learning with hierarchical latent variables. In *International Conference on Artificial Intelligence and Statistics*, pp. 2677–2703. PMLR, 2023.

Kaiqing Zhang, Sham Kakade, Tamer Basar, and Lin Yang. Model-based multi-agent rl in zero-sum markov games with near-optimal sample complexity. *Advances in Neural Information Processing Systems*, 33: 1166–1178, 2020a.

Kaiqing Zhang, Tao Sun, Yunzhe Tao, Sahika Genc, Sunil Mallya, and Tamer Basar. Robust multi-agent reinforcement learning with model uncertainty. *Advances in neural information processing systems*, 33: 10571–10583, 2020b.

Xinyu Zhao, Jianxiang Liu, Faguo Wu, Xiao Zhang, and Guojian Wang. Uncertainty modified policy for multi-agent reinforcement learning. *Applied Intelligence*, 54(22):12020–12034, 2024.

Shan Zhong, Gang Wang, Jingkui Zhang, Xiaoyang Wang, Kah Chan Teh, He Diao, Ping Zhang, Jiacheng He, Zhi Zeng, Tee Hiang Cheng, et al. Mad3pg: A framework for multi-agent deep denoising diffusion policy gradient optimization. *Information Fusion*, pp. 104026, 2025.

A Mathematical Proof

A.1 Proof of Theorem 3.2

Proof. To prove the existence and uniqueness of $\bar{V}_i^*(s)$, we rely on the Banach fixed-point theorem (Kardeş et al., 2011). The max-min Bellman operator T is a contraction mapping, as established in (He et al., 2023). Specifically, for any two value functions $\bar{V}_1, \bar{V}_2 \in \mathcal{V}$, the operator T satisfies:

$$\|T[\bar{V}_1] - T[\bar{V}_2]\|_\infty \leq \gamma \|\bar{V}_1 - \bar{V}_2\|_\infty.$$

where $\gamma \in [0, 1)$ is the discount factor. By the Banach fixed-point theorem, T has a unique fixed point $\bar{V}^* \in \mathcal{V}$, which satisfies $T[\bar{V}_*](s) = \bar{V}^*(s)$ for all $s \in \mathcal{S}$.

Thus, the robust value function $\bar{V}_i^*(s)$ exists and is unique. □

A.2 Proof of Lemma 3.3

Proof. Let $\mathcal{U}_G$ denote the general uncertainty set:

$$\mathcal{U}_G = \{(\mathcal{P}, \delta) : \mathcal{P} \in \tilde{\mathcal{P}}_s, \delta \in \Delta_\pi\}.$$

The hierarchical entropy-based uncertainty set $\mathcal{U}_H$ is defined as:

$$\mathcal{U}_H = \{(\mathcal{P}, \delta) : \mathcal{P} \in \tilde{\mathcal{P}}_s(\delta), \delta \in \Delta_\pi\}.$$

Both $\mathcal{U}_G$ and $\mathcal{U}_H$ share the same set for δ, i.e., $\delta \in \Delta_\pi$. Thus, the δ component of $\mathcal{U}_H$ is a subset of that in $\mathcal{U}_G$.

In $\mathcal{U}_H$, $\mathcal{P}$ is conditionally dependent on δ, and $\mathcal{P} \in \tilde{\mathcal{P}}_s(\delta)$. Since $\tilde{\mathcal{P}}_s(\delta) \subseteq \tilde{\mathcal{P}}_s$, we have:

$$\mathcal{P} \in \tilde{\mathcal{P}}_s(\delta) \implies \mathcal{P} \in \tilde{\mathcal{P}}_s.$$

Since both the components in U_H are subsets of the U_G, U_H is a subset of the U_G. □

A.3 Proof of Theorem 3.4

Proof. To prove that the performance lower bound under the hierarchical entropy-based uncertainty set is no worse than that under the general uncertainty set, we proceed as follows:

$$\begin{aligned}
\inf_{\mathcal{U}_H} \bar{V}_{\bar{\mathcal{P}},\delta}(s) &= \inf_{\substack{\bar{\mathcal{P}} \in \tilde{\mathcal{P}}_s(\delta), \\ \delta \in \Delta_\pi}} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \mid s_0 = s\right] \\
&\geq \inf_{\substack{\bar{\mathcal{P}} \in \tilde{\mathcal{P}}_s, \\ \delta \in \Delta_\pi}} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \mid s_0 = s\right] \\
&= \inf_{\mathcal{U}_G} \bar{V}_{\bar{\mathcal{P}},\delta}(s).
\end{aligned}$$

The inequality holds because $\tilde{\mathcal{P}}_s(\delta) \subseteq \tilde{\mathcal{P}}_s$ is proved by Lemma 3.3 and the works (Kardeş et al., 2011; Xu et al., 2023) demonstrated, which means that the hierarchical entropy-based uncertainty set imposes additional constraints on $\mathcal{P}$ compared to the general uncertainty set. Thus, we conclude:

$$\inf_{\mathcal{U}_H} \bar{V}_{\mathcal{P},\delta}(s) \geq \inf_{\mathcal{U}_G} \bar{V}_{\mathcal{P},\delta}(s).$$

□

A.4 Proof of Theorem 3.6

Proof. Following the proof (Zhang et al., 2020b; Kardeş et al., 2011), we establish the existence of an ARE by applying Kakutani's Fixed Point Theorem.

Define the correspondence $\Phi : \Pi \to 2^{\Pi}$, where Π denotes the space of all joint policies, as follows:

$$\Phi(\pi) = \Bigg\{ \tilde{\pi} : \ \tilde{\pi}_i(\cdot \mid s) \in \arg\max_{\pi_i(\cdot|s)} \min_{\substack{\delta_i \in \Delta_{\pi_i} \\ \tilde{\mathcal{P}}(\cdot|s,\cdot) \in \tilde{\mathcal{P}}_s}} \sum_{a_i \in \mathcal{A}_i} \pi(a_i \mid s)$$

$$\prod_{j \neq i} \big(\pi_j(a_j \mid s) + \delta_j\big) \Big(R_i(s,a) + \gamma \sum_{s' \in \mathcal{S}} \tilde{\mathcal{P}}(s' \mid s, a)\, \bar{V}_i(s') \Big) \Bigg\}.$$

where $\bar{V}_i(s)$ has been proved to converge to the unique $\bar{V}_i^*(s)$. To apply Kakutani's Fixed Point Theorem, we verify the following conditions:

- **Non-emptiness:** For any joint policy $\pi \in \Pi$, the maximization over π_i and minimization over δ_i and $\tilde{\mathcal{P}}$ are well-defined because the feasible sets Δ_{π_i} and $\tilde{\mathcal{P}}_s$ are compact and convex. Thus, the solution set of the optimization problem, and hence $\Phi(\pi)$, is non-empty.
- **Compactness and Convexity:** The set $\Phi(\pi)$ is compact and convex. Compactness arises because the feasible sets Δ_{π_i} and $\tilde{\mathcal{P}}_s$ are compact, and the objective function is continuous in π_i. To show that $\Phi(\pi)$ is convex, let $\pi_1, \pi_2 \in \Phi(\pi)$. For any $\lambda \in [0,1]$, consider the convex combination $\pi' = \lambda\pi_1 + (1-\lambda)\pi_2$.

  Given the structure of the policy function π, we have:

  $$\pi(a \mid s) = \sum_{a \in \mathcal{A}} \prod_{j \neq i} \pi_j(a_j \mid s) \cdot \pi_i(a_i \mid s),$$

  which is linear with respect to $\pi_j(a_j \mid s)$. Since $\pi_j(a_j \mid s)$ is linear in π_j, any convex combination of π_1 and π_2 satisfies:

  $$\pi_j(a_j \mid s; \pi') = \lambda \pi_j(a_j \mid s; \pi_1) + (1-\lambda)\pi_j(a_j \mid s; \pi_2).$$

  Furthermore, the feasible sets Δ_{π_i} and $\tilde{\mathcal{P}}_s$ are convex by assumption, leveraging the fact that Shannon entropy constraints define convex level sets. Thus, the max-min optimization problem defining $\Phi(\pi)$ preserves convexity:

  $$\pi' \in \Phi(\pi).$$

  Therefore, $\Phi(\pi)$ is a convex correspondence.
- **Compactness of the Policy Space Π:** The policy space Π is compact because $\mathcal{S}$ and $\mathcal{A}$ are finite, and the joint policies π are probability distributions over $\mathcal{A}$. This ensures that Π is a subset of a finite-dimensional simplex, which is compact.

By Kakutani's Fixed Point Theorem, the correspondence Φ admits at least one fixed point $\pi_* \in \Phi(\pi_*)$. This fixed point corresponds to a joint policy π_* satisfying the equilibrium condition.

Thus, the joint policy $\tilde{\pi_*}$ and the value functions $\bar{V}_i^*$ constitute an ARE. □

A.5 Proof of Policy Gradient Updating Rules

Proof. We derive the policy gradient update for each agent $i \in \mathcal{N}$ under the defined joint strategy $\tilde{\pi}_\theta = (\pi_{\theta_1}, \delta_{w_1}, \mathcal{P}_{\xi_1}, \ldots, \pi_{\theta_N}, \delta_{w_N}, \mathcal{P}_{\xi_N})$.

Taking the gradient of $J_i(\theta)$ with respect to θ_i, we start by expanding:

$$\nabla_{\theta_i} J_i(\theta) = \nabla_{\theta_i} \bar{V}^i_{\tilde{\pi}_\theta}(s_0).$$

Using the recursive definition of

$$J_i(\theta) = \bar{V}^i_{\tilde{\pi}_\theta}(s_0) \;=\; \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \zeta_{\tilde{\pi}_\theta}(s,a)\, Q_{\psi_i}(s,a)$$

$$\begin{aligned}
\nabla_{\theta_i} J_i(\theta) &= \nabla_{\theta_i} \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\, Q_{\psi_i}(s,a) \\
&= \sum_{s,a} \nabla_{\theta_i}\Big[\zeta_{\tilde{\pi}_\theta}(s,a)\Big]\, Q_{\psi_i}(s,a) \\
&= \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; \nabla_{\theta_i} \log\Big[\zeta_{\tilde{\pi}_\theta}(s,a)\Big]\; Q_{\psi_i}(s,a) \\
&= \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; \nabla_{\theta_i} \log\Big[\tilde{\pi}_\theta(a \mid s)\Big]\; Q_{\psi_i}(s,a), \\
&= \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; \nabla_{\theta_i} \log\Big[\pi_{\theta_i}(a_i \mid s) + \delta_{w_i}(a_i \mid s)\Big]\; Q_{\psi_i}(s,a) \\
&= \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; \frac{\nabla_{\theta_i}\, \pi_{\theta_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) + \delta_{w_i}(a_i \mid s)}\; Q_{\psi_i}(s,a) \\
&= \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; \frac{\pi_{\theta_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) + \delta_{w_i}(a_i \mid s)} \nabla_{\theta_i}\; \log\, \pi_{\theta_i}(a_i \mid s)\; Q_{\psi_i}(s,a) \\
&= \mathbb{E}_{s,a\sim\zeta_{\tilde{\pi}_\theta}}\left[\frac{\pi_{\theta_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) + \delta_{w_i}(a_i \mid s)}\; \nabla_{\theta_i}\; \log\, \pi_{\theta_i}(a_i \mid s)\; Q_{\psi_i}(s,a)\right].
\end{aligned}$$

Similarly, we derive the gradient update of the ω:

$$\begin{aligned}
\nabla_{\omega_i}\, J_i(\omega) =&\; \nabla_{\omega_i} \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; Q_{\psi_i}(s,a) \\
=&\; \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\, \nabla_{\omega_i}\; \log\Big[\tilde{\pi}_\theta(a \mid s)\Big]\, Q_{\psi_i}(s,a) \\
=&\; \sum_{s,a} \zeta_{\tilde{\pi}_\theta}(s,a)\; \frac{\nabla_{\omega_i}\, \delta_{\omega_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) +\; \delta_{\omega_i}(a_i \mid s)}\; Q_{\psi_i}(s,a) \\
=&\; \mathbb{E}_{s,a\sim\zeta_{\tilde{\pi}_\theta}}\left[\frac{\nabla_{\omega_i}\, \delta_{\omega_i}(a_i \mid s)}{\pi_{\theta_i}(a_i \mid s) + \delta_{\omega_i}(a_i \mid s)}\; Q_{\psi_i}(s,a)\right].
\end{aligned}$$

Additionally, we use sampling-based method to update ξ:

$$J_i(\xi) = \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \zeta_{\tilde{\pi}_\theta}(s,a)\, Q_{\psi_i}(s,a),$$

$$\tau = (s_0, a_0, s_1, a_1, \ldots, s_{T-1}, a_{T-1}, s_T),$$

$$\nabla_{\xi_i} \log \Pr_{\xi_i}(\tau) = \sum_{t=0}^{T-1} \nabla_{\xi_i} \log P_{\xi_i}\big(s^{t+1} \mid s^t, a^t\big),$$

$$\nabla_{\xi_i} J_i(\xi) \approx \mathbb{E}_{\tau\sim(P_{\xi_i},\,\tilde{\pi}_\theta)}\left[\Big(\sum_{t=0}^{T-1} \nabla_{\xi_i} \log P_{\xi_i}\big(s^{t+1} \mid s^t, a^t\big)\Big)\cdot J_i(s,\tau)\right].$$

□

B Environment Settings

B.1 Environment Details

B.1.1 Cooperative Navigation

This scenario involves **3 agents** and **3 landmarks**. The agents receive rewards based on their distance to the landmarks, with the goal of minimizing the distance between themselves and any landmark. However, collisions between agents result in penalties. Therefore, the agents must learn to efficiently allocate themselves across the landmarks while avoiding collisions.

B.1.2 Predator-Prey

In the predator-prey environment, there are **3 faster "good agents"** and **1 adversary** who is slower. The goal of the good agents is to evade capture, while the adversaries aim to catch them. The environment also includes large, immovable obstacles, which create barriers and add complexity to the movement strategies of the agents.

B.1.3 Keep-Away

This scenario includes **1 agent**, **1 adversary**, and **1 landmark**. The agent earns rewards by staying close to the landmark, while the adversary gains rewards by either approaching the landmark or forcing the agent away from it. As a result, the adversary learns strategies to block or disrupt the agent's attempts to maintain proximity to the landmark.

B.1.4 Physical Deception

In this environment, there are **2 good agents**, **1 adversary**, and **2 landmarks**. Among the landmarks, one is designated as the "target landmark." Good agents are rewarded for being close to the target landmark but are penalized if the adversary approaches it. Meanwhile, the adversary earns rewards for getting closer to the target landmark, despite not knowing which one it is. To succeed, the good agents must learn to coordinate and strategically split up, effectively misleading the adversary into targeting the wrong landmark.

B.1.5 Multi-Agent MuJoCo Tasks

To evaluate robustness and scalability in continuous control settings, we further employ a set of **multi-agent MuJoCo benchmarks**. These tasks include **Cheetah** 2×3, **Cheetah** 6×1, **Swimmer**, and **Walker**. In the Cheetah environments, multiple limbs (or agents) must coordinate to generate efficient locomotion while maintaining stability under external perturbations. The **Cheetah** 2×3 setup partitions the body into two groups of three actuators each, promoting inter-group coordination, whereas the **Cheetah** 6×1 task treats all six actuators as independent agents that must achieve synchronized gait control. The **Swimmer** and **Walker** tasks evaluate robustness in low- and high-dimensional continuous action spaces, respectively, where agents must maintain smooth propulsion and balance under stochastic disturbances. These environments collectively provide diverse testing conditions for analyzing the robustness and coordination capabilities of our framework across both discrete and continuous domains.

B.2 Metrics

We evaluate the robustness and effectiveness of the proposed framework under three uncertainty levels: **low**, **mid**, and **high**. The bounds for interaction and environmental uncertainties (ρ_{int} and ρ_{env}) are set to 0.1, 0.5, and 1.5 for these three levels, respectively. Each experiment is repeated with five fixed random seeds (`111-120`) to ensure reproducibility, and results are averaged across these runs.

Cumulative Reward. The primary performance metric is the cumulative reward (or return), defined as the sum of all rewards obtained within one episode:

$$R_{\text{total}} = \sum_{t=0}^{T} R^t,$$

which measures the overall task success and long-term coordination ability of the agents.

Reward Structure in MPE. In the Multi-Agent Particle Environment (MPE) tasks, the reward for each agent is defined as the negative Euclidean distance to its target (e.g., landmark or goal position), encouraging agents to minimize their distance while avoiding penalties such as collisions. For example, in *Cooperative Navigation*, each agent receives $r_i = -\|p_i - p_{\text{landmark}}\|_2$ plus a small penalty when colliding with others. This design encourages both spatial efficiency and cooperative distribution across goals.

Reward Structure in MuJoCo. In the continuous-control MuJoCo tasks, the total reward at each step combines several components: a forward-progress term, a control-effort penalty, and a survival bonus. The instantaneous reward is given by

$$r_t = r_{\text{forward}} - c_{\text{control}} + r_{\text{survive}},$$

where r_{forward} measures the agent's velocity in the target direction, c_{control} penalizes large actuator torques or joint accelerations, and r_{survive} provides a small constant bonus for maintaining a stable posture. This formulation encourages energy-efficient and stable locomotion under uncertainty.

B.3 Uncertainty Injection and Baselines' Settings

In this subsection, we describe how environmental and interaction uncertainties are instantiated during training and evaluation, and how these uncertainty sources are applied consistently across all compared methods. Rather than directly prescribing the magnitude of injected noise, we quantify uncertainty by the sizes of entropy-bounded uncertainty sets, parameterized by ρ_{env} and ρ_{int}.

Environmental Uncertainty Environmental uncertainty models stochasticity in the transition dynamics. For each environment, we inject environmental noise by perturbing the nominal transition function $P(s' \mid s, a)$ with a stochastic disturbance ρ_{env} in Eq. 9 that controls the noise magnitude. We define three noise regimes: We consider three environmental noise regimes, corresponding to $\rho_{\text{env}} = 0.1$ (low), $\rho_{\text{env}} = 0.5$ (mid), and $\rho_{\text{env}} = 1.5$ (high), respectively. For MuJoCo environments, the noise is applied to joint accelerations and velocities; for MPE tasks, the noise is used to agent positions and velocities after each transition.

Interaction Uncertainty Interaction uncertainty captures stochasticity and unpredictability in agents' action selections. At each timestep, the executed action $\tilde{a}_i$ for agent i is sampled from a perturbed policy that ρ_{int} in Eq. 8 controls the strength of interaction noise. We consider: $\rho_{\text{int}} = 0.1$ (low), $\rho_{\text{int}} = 0.5$ (mid), and $\rho_{\text{int}} = 1.5$ (high).

Combined Uncertainty Setting Unless otherwise stated, both environmental and interaction uncertainties are simultaneously present during evaluation. That is, at each timestep, agents execute perturbed actions $\tilde{a}_i$ and transitions evolve according to the noisy dynamics described above.

Baseline-Specific Handling of Uncertainty We now clarify how each baseline interacts with the injected uncertainties:

- **MADDPG (MAD)**: Does not explicitly model uncertainty. Both environmental and interaction noise are treated as exogenous disturbances.
- **M3DDPG (M3D)**: Models adversarial perturbations at the policy level and is therefore explicitly robust to interaction uncertainty. Environmental noise is injected externally and not optimized adversarially.

Parameter	Description	Value
Episode Length	Length of each episode in training	50
Memory Length	Size of the replay buffer	1×10^5
Tau	Soft update coefficient for target networks	0.001
Gamma	Discount factor for future rewards	0.95
Seed	Seed for random number generators	111 to 115
Actor Learning Rate	Learning rate for actor-network	0.001
Critic Learning Rate	Learning rate for critic network	0.0001
Batch Size	Number of samples per batch update	64
Epsilon Decay	Rate of decay for exploration epsilon	10000

Table 2: Detailed hyperparameters used in the experiment.

Network Type	Input	Layers and Neurons	Activation Functions
Actor	Observation (`dim_observation`)	FC1: 500, FC2: 128, FC3: `dim_action`	ReLU, ReLU, Tanh
Critic	Observation and Action (`n_agent` × `dim_observation` + `n_agent` × `dim_action`)	FC1: 1024, FC2: 512, FC3: 300, FC4: 1	ReLU, ReLU, ReLU, None
Adv_Actor_Type_1	Observation (`dim_observation`)	FC1: 500, FC2: 128, FC3: 1	ReLU, ReLU, Tanh
Adv_Actor_Type_2	Extended Observation (`dim_observation` + 1)	FC1: 500, FC2: 128, FC3: 1	ReLU, ReLU, Tanh
Adv_Critic	Observation and Action (`n_agent` × `dim_observation` + `n_agent` × `dim_action`)	FC1: 1024, FC2: 512, FC3: 300, FC4: 1	ReLU, ReLU, ReLU, None
Transition Model	State and Action (`state_dim` + `action_dim`)	FC1: `hidden_dim`, FC2: `hidden_dim`, Mean and Logvar Heads: `state_dim`	ReLU, ReLU, None, None

Table 3: Detailed Architecture of Neural Networks

- **R-MADDPG (R-MAD)**: Models adversarial perturbations in transition and reward dynamics, and is therefore robust to environmental uncertainty. Interaction noise is injected externally and not optimized adversarially.
- **ARE-AC (Ours)**: Explicitly models both uncertainty sources using hierarchical entropy-based uncertainty sets. Interaction perturbations are optimized at the upper level, and environmental perturbations are optimized conditionally at the lower level.

C Hyperparameters

In this section, we introduce the detailed setting of our algorithm and experiments.

First of all, the hyperparameters are defined as:

Besides, our detailed neural network structure is listed as follows:

D Algorithm Details

Algorithm 1 Actor-Critic for Robust Multi-Agent RL with Uncertainty Decomposition (ARE-AC)

Require: Initialize critic network parameters $\{\psi_i\}_{i\in\mathcal{N}}$, adversarial parameters $\{\omega_i\}_{i\in\mathcal{N}}$, $\{\xi_i\}_{i\in\mathcal{N}}$, policy parameters $\{\theta_i\}_{i\in\mathcal{N}}$, probabilistic dynamics model ϕ and replay buffer $\mathcal{D}$.

for episode $= 1$ to M **do**
  Receive an initial state s_0.
  for $t = 1, \ldots, T$ **do**
    for $i \in \mathcal{N}$ **do**
      Sample a set of candidate actions $\{a^k\}_{k=1}^{K}$.
      Compute entropy based on sampled actions, current states, and policies to determine uncertainty sets by Eq. 23 and 27.
      Obtain strategy perturbation $\delta_{\omega_i}(\cdot \mid s^t)$ based on current state.
      Obtain transition perturbation $\mathcal{P}_{\xi_i}(\cdot \mid s^t, \delta_{\omega_i})$ based on current states and policy perturbation.
      Sample action $a_i \sim \pi_{\theta_i}(\cdot \mid s^t) + \delta_{\omega_i}(a_i \mid s^t)$.
    end for
    Execute action a, observe new state s^{t+1} (perturbed by $\mathcal{P}_{\xi_i}$), and receive rewards R^t.
    Store $(s^t, a, \delta_{\omega_i}, \mathcal{P}_{\xi_i}, R^t, s^{t+1})$ in replay buffer $\mathcal{D}$.
    Update $s^t \leftarrow s^{t+1}$.
  end for
  for $i \in \mathcal{N}$ **do**
    Sample mini-batch $(s^t, a^t, \delta_{\omega_i}, \mathcal{P}_{\xi_i}, R^t, s^{t+1})_{t=1}^{S}$ from $\mathcal{D}$.
    Update critic by the Eq. 20.
    Update actor using the Eq. 22.
    Update adversarial strategy perturbation network ω_i by Eq. 18.
    Update adversarial transition perturbation network ξ_i by Eq. 19.
    Update the probabilistic dynamics model by Eq. 24.
  end for
end for