Quality-Diversity for Multi-Agent Reinforcement Learning

Hao Chen*
The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, CASIA
School of Artificial Intelligence, UCAS
Beijing, China
chenhao2022@ia.ac.cn

Pengyi Li*
Tianjin University
Tianjin, China
lipengyi@tju.edu.cn

Bin Zhang†
Institute of Automation, CASIA
School of Artificial Intelligence, UCAS
Beijing, China
zhangbin2020@ia.ac.cn

Hu Fu
Huazhong University of Science and Technology
Wuhan, China
fuhu@hust.edu.cn

Zhiwei Xu
Shandong University
Shandong, China
zhiwei_xu@sdu.edu.cn

Ce Zhang
Institute of Automation, CASIA
School of Artificial Intelligence, UCAS
Beijing, China
zhangce2023@ia.ac.cn

Xinyue Lu
Institute of Automation, CASIA
School of Artificial Intelligence, UCAS
Beijing, China
luxinyue2023@ia.ac.cn

Guoliang Fan†
Institute of Automation, CASIA
School of Artificial Intelligence, UCAS
Beijing, China
guoliang.fan@ia.ac.cn

ABSTRACT

Quality–diversity optimization (QD) in multi-agent reinforcement learning (MARL) aims to evolve a population of team policies that are both high-performing and behaviorally diverse, enabling effective coordination in complex cooperative tasks. However, existing QD approaches often depend on random exploration to encourage diversity, resulting in unstable learning and limited coverage in high-dimensional environments. We propose MIQD, a mutual-information–enhanced QD framework that integrates fragment-based behavioral descriptors into the critic to capture short-term patterns and guide policy updates. Mutual information measures alignment between policy behavior and target descriptors; its step-level decomposition yields intrinsic rewards that promote alignment at each state–action pair. Experimental results show that our method consistently outperforms strong baselines across multiple metrics, demonstrating its effectiveness in jointly enhancing policy quality and diversity.

KEYWORDS

Multi-Agent Reinforcement Learning; Mutual Information; Quality-Diversity

*Equal contribution.
†Corresponding authors.

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/SLJK5791

ACM Reference Format:
Hao Chen, Pengyi Li, Bin Zhang, Hu Fu, Zhiwei Xu, Ce Zhang, Xinyue Lu, and Guoliang Fan. 2026. Quality-Diversity for Multi-Agent Reinforcement Learning. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/SLJK5791

1 INTRODUCTION

Multi-agent reinforcement learning (MARL) [29] has been widely applied to solving complex collaborative tasks, such as game AI [32], autonomous driving [12], and robotic control [2]. Among these, cooperative tasks are particularly prevalent: agents interact with the environment to obtain team rewards, and by maximizing these rewards, they develop more effective joint strategies. A commonly adopted paradigm in MARL is the centralized training with decentralized execution (CTDE) [10] framework. In this framework, agents have access to global information during training, while execution relies solely on each agent's local observations. This approach enables agents to learn how to cooperate efficiently.

In multi-agent reinforcement learning, the primary objective is to maximize team rewards and thereby obtain effective cooperative strategies. Most approaches are based on gradient updates of the value function: they maintain a multi-agent policy, apply temporal difference (TD) [20] methods to approximate the centralized value function, and then update each agent's policy through value decomposition [28] or actor–critic techniques [6].

Although these approaches can efficiently improve joint performance, they often overlook policy diversity. A lack of diversity can cause convergence to suboptimal equilibria and reduce robustness when facing novel or perturbed environments [3].

In contrast, evolutionary algorithms [4] optimize a population of policies rather than a single solution, using stochastic operators

such as crossover and mutation [19] to explore the policy space. This population-based mechanism naturally maintains diversity and allows for broader behavioral exploration. Building upon this idea, quality–diversity optimization [26] explicitly aims to evolve a set of policies that are both high-performing and behaviorally distinct. QD methods maintain an archive of policies organized by behavioral descriptors, which summarize how a policy behaves over an episode and provide a basis for quantifying diversity.

Classic single-agent QD algorithms, such as MAP-Elites [30], discretize the descriptor space into a grid-based archive and retain the best-performing policy within each cell. PGA-MAP-Elites [21] combines gradient-based policy improvement with evolutionary search by incorporating a critic to guide mutations toward higher performance. QD-PG [25] further augments the critic with an intrinsic diversity reward based on state novelty, encouraging exploration of underrepresented regions in the state space.

However, these methods primarily rely on stochasticity to induce behavioral diversity. Although behavioral descriptors (BDs) summarize how an agent or team behaves, they rarely participate directly in learning. Their critics typically account only for the environmental reward function or the added diversity reward, while neglecting the role of BDs in gradient updates. Some approaches have attempted to incorporate BDs directly into value function updates. For instance, DCG-MAP-ELITES [5] stores both states and their corresponding BDs during sampling, then uses the distance between the target BD and the sampled BD as an additional reward signal. This allows the value function to partially account for agent behavior during updates. Nevertheless, such a simple reward modeling scheme is insufficient to fully capture how BD-related behaviors influence different state–action pairs.

Existing QD methods compute BDs over entire trajectories, missing local behavioral variations. In long-horizon tasks (e.g., robot control), behavior changes across phases, while a global BD neglects these dynamics. To address this, we introduce MIQD, a novel mutual information enhanced quality-diversity method tailored for the multi-agent reinforcement learning framework. Our approach addresses key challenges in balancing policy performance and behavioral diversity. First, we propose the use of fragment behavioral descriptors to characterize agent behaviors during the sampling process. Traditional behavioral descriptor representations are typically computed over entire episodes, which prevents them from accurately capturing local behavioral variations. As a result, global BDs may fail to reflect short-term behavioral dynamics within long trajectories. By contrast, fragment BDs capture the behavioral tendencies of agents over shorter temporal segments, thereby providing a more accurate and localized description of their actions. Instead of modeling rewards solely based on distances between target and observed BDs, we leverage the mutual information of actions and fragment BDs to model rewards. This design enables a more precise and robust alignment between the learned policy and the targeted behavioral characteristics. Second, we address the mismatch between multi-step behavioral descriptor representations and the single-step nature of value updates. To resolve this, we introduce a mutual information factorization approach that decomposes the deviation between an agent's multi-step behavior and the target BD into step-wise contributions. This decomposition distributes behavioral feedback across individual state–action pairs, ensuring the value function consistently reflects the alignment between actions and the intended behavioral objectives. Finally, to fully leverage the agents generated during training, we integrate the Cross-Entropy Method [16] into the optimization pipeline, allowing high-performing offspring from the evolutionary process to guide policy improvement. Additionally, we employ a population-based approach with neighboring policies to achieve more accurate target value estimation during critic updates. Together, these contributions position MIQD as a flexible and effective framework for promoting both quality and diversity in MARL, offering a principled balance between performance optimization and behavioral exploration.

We demonstrate the effectiveness of MIQD in the Multi-Agent MuJoCo (MAMUJOCO) [23] environment. Experiments are conducted under different BD dimensional settings, and performance is evaluated across three metrics: maximum fitness, coverage, and QD score. Our method achieved consistently strong results on different tasks. Furthermore, to validate the contributions of individual components, we conducted ablation studies examining the effects of fragment BDs, the mutual-information-based reward, and the use of neighborhood policies.

2 BACKGROUND

2.1 Dec-POMDP

Decentralized partially observable Markov decision processes (Dec-POMDPs) [10] provide a general framework for modeling cooperative multi-agent reinforcement learning (MARL) tasks. A Dec-POMDP is defined as $G = \langle S, U, A, P, r, Z, O, \gamma \rangle$, where $a \in A := \{1, \ldots, N\}$ denotes an agent, and $s \in S$ represents the global state of the environment. At each timestep, every agent receives a local observation $z_a \in Z$ through the observation function $O(s, a) : S \times A \rightarrow Z$, and selects an action $u_a \in U_a$. The joint action of all agents is denoted as $\mathbf{u} \in \mathbf{U}$. The environment then transitions from state s to s' according to the transition function $P(s'|s, \mathbf{u}) : S \times \mathbf{U} \times S \rightarrow [0, 1]$. All agents share a common reward function $r(s, \mathbf{u}) : S \times \mathbf{U} \rightarrow \mathbb{R}$, and the objective is to maximize the expected discounted return: $R = \sum_{t=0}^{\infty} \gamma^t r_t$, where $\gamma \in [0, 1)$ is the discount factor. Each agent follows a stochastic policy $\pi(u|\tau) : T \times U \rightarrow [0, 1]$, where $\tau \in T$ denotes its local action–observation history.

2.2 Policy optimization

In the CTDE paradigm, a common learning architecture is the actor–critic framework. Each agent $a \in A$ maintains its own policy, parameterized by θ_a, denoted as $\pi_{\theta_a}(u_a|z_a)$, where z_a is the local observation available to the agent. The actor is responsible for generating decentralized actions during execution.

During training, however, critics are typically centralized. Each agent is associated with a centralized critic function $Q(s, \mathbf{u})$, which conditions on the global state s (or the joint observation) and the joint action $\mathbf{u}$. The critic evaluates the expected return of joint actions, thereby providing more informative gradients to guide policy updates. The actor parameters are updated by maximizing the expected Q-value under the policy distribution:

$$\nabla_{\theta_a} J(\pi_{\theta_a}) = \mathbb{E}\big[\nabla_{\theta_a} log \pi_{\theta_a}(u_a|z_a) Q(s, \mathbf{u})\big]. \tag{1}$$

This actor–critic formulation under CTDE enables each agent to learn a decentralized policy while exploiting centralized information during training. Variants such as MADDPG [18] and MATD3 [1] extend this framework by incorporating techniques like deterministic policy gradients, twin critics, and target policy smoothing to improve stability and sample efficiency. Such actor–critic architectures serve as the backbone of many modern MARL algorithms and provide the foundation for integrating diversity-aware optimization.

2.3 Single-Agent Quality-Diversity

The objective of quality-diversity methods is to simultaneously achieve high performance and behavioral diversity. Formally, let π_θ denote a policy parameterized by θ, and let $f(\pi_\theta)$ denote its fitness, typically defined as the undiscounted reward:

$$f(\pi_\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r(s_t, u_t)\right], \tag{2}$$

where τ is a trajectory generated by π_θ. A behavioral descriptor function $\phi : \tau \mapsto \mathbb{R}^d$ maps each trajectory to a d-dimensional descriptor vector that captures its behavioral characteristics.

MAP-Elites realizes this objective by discretizing descriptor space Φ into a grid-based archive $\mathcal{A}$. Each cell $\mathcal{A}[i]$ corresponds to a region of the descriptor space and stores the policy with the highest observed fitness in that region:

$$\mathcal{A}[i] = \arg\max_{\pi_\theta\in\mathcal{P}} f(\pi_\theta) \quad \text{s.t. } \phi(\pi_\theta) \in \Phi_i, \tag{3}$$

where Φ_i denotes the descriptor subspace associated with cell i.

During training, MAP-Elites samples parent policies π_θ from $\mathcal{A}$ and generates offspring $\pi_{\theta'}$ through variation operators such as mutation and crossover. After evaluating the offspring, the archive is updated as:

$$\mathcal{A}[i] \leftarrow \begin{cases} \pi_{\theta'} & \text{if } \phi(\pi_{\theta'}) \in \Phi_i \quad \textit{and} \quad f(\pi_{\theta'}) > f(\mathcal{A}[i]), \\ \mathcal{A}[i] & \text{otherwise.} \end{cases} \tag{4}$$

The goal of QD is not only to maximize $f(\pi_\theta)$, but also to discover a set of policies $\{\pi_i\}_{i=1}^{|\mathcal{A}|}$ that are both high-performing and well-distributed across the behavioral descriptor space $\mathcal{B}$. This objective can be expressed as: $\max_{\mathcal{A}} \sum_{\pi_i\in\mathcal{A}} f(\pi_i)$.

While MAP-Elites relies solely on evolutionary operators, PGA-MAP-Elites incorporates policy gradient optimization. A critic $Q_\psi(s, u)$ is trained to estimate state–action values, and policy is updated by the deterministic policy gradient:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}\left[\nabla_\theta \pi_\theta(s) \nabla_u Q_\psi(s, u)\big|u = \pi_\theta(s)\right]. \tag{5}$$

In this way, the mutation process is guided not only by stochastic variation but also by informed policy updates. Moreover, trajectories τ generated by offspring are added to a replay buffer $\mathcal{D}$, which is used to iteratively improve Q_ψ and further refine the archive. By combining gradient-based exploitation with descriptor-driven exploration, PGA-MAP-Elites achieves a more effective balance between performance maximization and diversity preservation.

3 RELATED WORK

Many QD methods rely on the stochasticity of crossover and mutation to promote policy diversity. However, such mechanisms often fall short of producing sufficiently diverse solutions. To overcome this limitation, QD-PG introduces an explicit diversity reward. It maintains a state archive and measures the novelty of sampled states with respect to this archive. A separate diversity critic is then trained using this reward, and policies drawn from the archive are updated under the guidance of this critic. By encouraging policies to explore novel regions of the state space, QD-PG achieves greater behavioral diversity than approaches that rely solely on random variation.

In addition to state novelty, other methods incorporate behavioral descriptors directly into the critic. For instance, DCG-MAP-Elites conditions the critic on BD values, where the reward is determined by the distance between the BD of the current episode and a target BD. This design allows policies to be updated with respect to different BD objectives, thereby guiding the search toward more diverse behavioral patterns. By conditioning the critic on BDs, DCG-MAP-Elites goes beyond simple novelty-based rewards and enables more direct control over diversity in the policy set.

Several notable variants build upon this integration. PBT-MAP-Elites [24] incorporates a population-based training procedure to jointly optimize the hyperparameters of diverse reinforcement learning agents, thereby enabling dynamic adaptation of training dynamics across the population. CMA-MEGA [7] further extends this idea by estimating descriptor gradients through evolution strategies and combining them with fitness gradients. This hybrid optimization process allows policies to improve performance while simultaneously achieving finer control over behavioral diversity.

Intrinsic rewards have been widely used to encourage exploration in multi-agent reinforcement learning. EITI [31] introduces influence-based intrinsic rewards to capture inter-agent dependencies and promote coordinated exploration. CERMIC [22] further calibrates intrinsic curiosity using learned models of inferred inter-agent intentions. NS-MERL [27] employs novelty-based fitness functions to encourage diverse exploratory behaviors for individual agents.

Recently, several methods have leveraged mutual information (MI) maximization to design intrinsic rewards that encourage greater behavioral diversity among agents. For instance, CDS [14] maximizes the MI [13] between agent trajectories and their identities, thereby promoting differentiation in both optimization and representation and enhancing inter-agent diversity. TKCA [8] adopts a mixture-of-experts (MoE) [33] architecture and maximizes the MI between the knowledge selection module and agent identities, resulting in more diverse knowledge representations. R3DM [9] maximizes the MI between agents' roles at a given timestep and their expected future trajectories, explicitly linking current roles with anticipated behaviors to facilitate complementary role emergence. Similarly, CIA [17] maximizes the MI between temporal credit assignments and agent identity representations to improve role differentiation through more distinguishable credit attributions. While these approaches effectively utilize MI to enhance diversity in multi-agent reinforcement learning, they focus primarily on role differentiation or identity-driven diversity within policy learning.

None of them incorporate MI into a quality–diversity framework that explicitly models and maintains diversity in the behavioral descriptor space. In contrast, our proposed MIQD method integrates MI with QD optimization, using mutual information to measure the alignment between state–action pairs and target behavioral descriptors. This integration enables an information-theoretic mechanism for promoting diversity not through random exploration or role labeling, but through the explicit modeling of behavioral descriptor consistency across the policy population.

4 METHOD

In this section, we introduce the proposed MIQD framework, a mutual-information-driven quality–diversity approach for multi-agent reinforcement learning. We begin by outlining the overall structure of MIQD and its learning process, highlighting how the method integrates behavioral descriptors into the value function to guide behavior-aware policy updates. We then analyze the mutual information factorization mechanism, which establishes a consistent link between multi-step behavioral descriptors and step-wise value optimization. Finally, we describe how population-based optimization and the cross-entropy method are incorporated to refine value estimation and exploit high-performing offspring throughout training. Together, these components form a cohesive framework that effectively enhances both performance and behavioral diversity in cooperative multi-agent settings, as shown in Figure 1.

4.1 Overall Framework

Behavioral descriptors are typically defined over entire trajectories, characterizing an agent's behavior across an episode. However, when the episode length is large, the global BD often deviates significantly from local behaviors within shorter segments of the trajectory. To reduce this discrepancy, we introduce fragment behavioral descriptors, which capture agent behaviors within localized temporal windows.

Formally, given an episode with state sequence $s_1, s_2, \ldots, s_T$, we first compute a smoothed local descriptor for each timestep t:

$$d_t = \frac{1}{2K+1} \sum_{t-K}^{t+K} \phi(s_t), \tag{6}$$

where the state descriptor function ϕ extracts a low-dimensional behavioral representation (e.g., joint positions, ground contact, velocities) from high-dimensional observations. For instance, in a two-legged robot locomotion task, the descriptor encodes which foot is in contact at each timestep; if only the first leg contacts the ground at time t, then $\phi(s_t) = [1, 0]$. Averaging descriptors over the trajectory yields a global behavioral descriptor summarizing the robot's overall gait, and K controls the window size. To further smooth variations across timesteps, we apply a second averaging step:

$$fd_t = \frac{1}{2K+1} \sum_{t-K}^{t+K} d_t. \tag{7}$$

This two-level averaging ensures that the fragment descriptors evolve smoothly over time and provide a stable characterization of local behaviors.

To encourage policies to update in the direction of specific behavioral descriptors rather than relying solely on stochasticity, we extend the value function to incorporate fragment behavioral descriptors. Specifically, the critic is defined as $Q(s, \mathbf{u}, fd)$, which estimates both the expected return of taking joint action $\mathbf{u}$ in state s and the degree to which this action aligns with the target fragment descriptor.

To enable the critic to capture this property, we introduce a behavioral relevance reward, which quantifies the correlation between a state–action pair and a target descriptor fd^*. The target descriptor is a behavioral descriptor assigned during each policy optimization round, guiding the policy to improve performance while matching the desired behavioral pattern. This reward is modeled using mutual information $I(s, \mathbf{u}; fd^*)$, thus encouraging the critic to learn how state–action pairs contribute to the realization of specific behavioral patterns. The critic is updated according to the following Bellman equation:

$$Q(s, \mathbf{u}, fd^*) = r(s, \mathbf{u}) + I(s, \mathbf{u}; fd^*) + \gamma \max_{\mathbf{u}'} Q(s', \mathbf{u}', fd^*), \tag{8}$$

where $r(s, \mathbf{u})$ is the extrinsic reward from the environment.

After obtaining the action-value function, we update policies in a manner analogous to PGA-MAP-Elites. We maintain an archive in which each cell corresponds to a region of the descriptor space and stores the best-performing policy for that region. During training, $\tilde{N}$ deterministic team policies $\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_{\tilde{N}}$ are sampled from the archive, and a target descriptor fd^* is randomly selected. For each sampled team policy $\boldsymbol{\pi}_i$, the individual policy of agent a, denoted as $\pi_{i,a}$ with parameters $\theta_{i,a}$, is updated to maximize the expected return under fd^*:

$$\nabla_{\theta_{i,a}} J(\pi_{i,a}) = \mathbb{E}\left[\nabla_{\theta_{i,a}} \pi_{i,a}(z_{i,a}) \nabla_{u_a} Q(s, \mathbf{u}, fd^*) \middle| u_a = \pi_{i,a}(z_{i,a})\right]. \tag{9}$$

4.2 Mutual Information Factorization

Although we introduce $I(s, \mathbf{u}; fd)$ to capture behavior relevance, its computation is not straightforward. This difficulty arises because the fragment behavioral descriptor is constructed from features aggregated across multiple timesteps. Consequently, it is challenging to directly determine, for a single state–action pair (s, u), how well the action aligns with a descriptor fd.

To address this issue, we decompose multi step-level mutual information into single step-level contributions. Specifically, consider a fragment episode consisting of $\xi_t = (s_{t-K}, \mathbf{u}_{t-K}, \ldots, s_{t+K}, \mathbf{u}_{t+K})$. Using Eqs. 6–7, we first compute its fragment descriptor fd. Since fd depends on multiple timesteps, the natural modeling choice is to estimate the mutual information $I(s, \mathbf{u}; fd|\xi)$.

For this purpose, we employ mutual information neural estimation MINE [11], which estimates mutual information as:

$$I(X; Y) = \sup_{\omega \in \Omega} \mathbb{E}_{p(x,y)}\left[T_\omega(x, y)\right] - \log \mathbb{E}_{p(x)p(y)}\left[e^{T_\omega(x,y)}\right], \tag{10}$$

where T_ω is a neural function parameterized by ω, and Ω is the set of all candidate functions. MINE provides a tractable and flexible way to approximate mutual information directly from samples, which is essential in high-dimensional MARL settings.

However, during policy gradient training, samples are drawn from the replay buffer at the step level. In this setting, the relevant

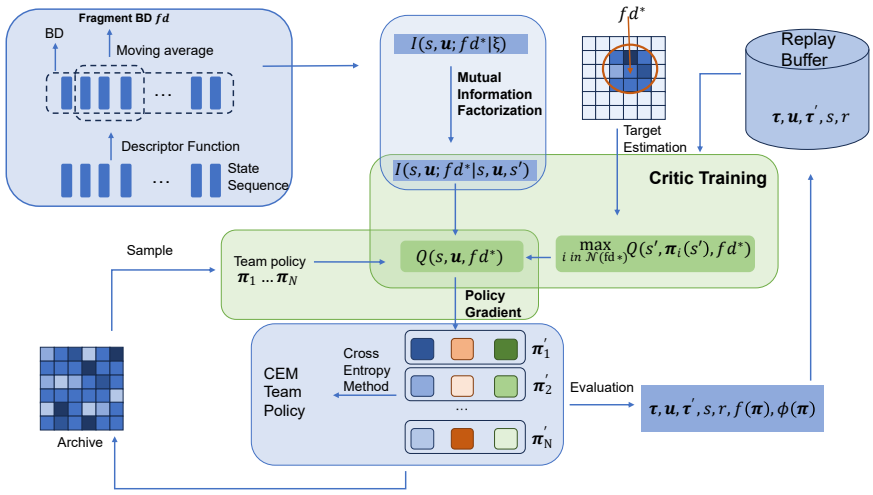


Figure 1: The overall framework of MIQD. Fragment behavioral descriptors are computed from state sequences, and mutual information is used to capture the relevance between actions and target behaviors. Multi-step MI is factorized into single-step terms for critic training with population-based target estimation. Policies sampled from the archive are updated via policy gradients, and CEM refines high-performing offspring.

form of mutual information becomes $I(s,\mathbf{u};fd|s,\mathbf{u},s')$. This discrepancy between fragment-level and step-level mutual information introduces a gap. To reconcile them, we adopt a mutual information decomposition, which shows that step-level MI is proportional to fragment-level MI. The derivation process of this decomposition is as follows:

$$
\begin{aligned}
&I_\omega(s,\mathbf{u};fd \mid \xi) = \\
&\mathbb{E}_{p(s,\mathbf{u},fd|\xi)}[T_\omega] - \log \mathbb{E}_{p(s,\mathbf{u}|\xi)\otimes p(fd|\xi)}[e^{T_\omega}] \\
&\propto \mathbb{E}_{p(s,\mathbf{u},fd|\xi)}[T_\omega] - \mathbb{E}_{p(s,\mathbf{u}|\xi)\otimes p(fd|\xi)}[e^{T_\omega}] \\
&= \mathbb{E}_{s,\mathbf{u},s'}\left[\mathbb{E}_{p(s,\mathbf{u},fd|s,\mathbf{u},s')}[T_\omega] - \mathbb{E}_{p(s,\mathbf{u}|s,\mathbf{u},s')\otimes p(fd|s,\mathbf{u},s')}[e^{T_\omega}]\right] \\
&\propto \mathbb{E}_{s,\mathbf{u},s'}\left[\mathbb{E}_{p(s,\mathbf{u},fd|s,\mathbf{u},s')}[T_\omega] - \log \mathbb{E}_{p(s,\mathbf{u}|s,\mathbf{u},s')\otimes p(fd|s,\mathbf{u},s')}[e^{T_\omega}]\right] \\
&= \mathbb{E}_{s,\mathbf{u},s'}[I_\omega(s,\mathbf{u};fd \mid s,\mathbf{u},s')]
\end{aligned}
\tag{11}
$$

This decomposition ensures that behavior-relevant rewards derived from step-level samples remain consistent with the longer-term behavioral descriptors computed over fragment episodes.

4.3 Population-Based Target Value Estimation

When updating the Q-network, the target value $\max_{\mathbf{u}'} Q(s',\mathbf{u}',fd^*)$ arises. However, because the action space is continuous, computing this maximum exactly is intractable. To overcome this, we utilize the offspring stored in the archive to do the estimation more accurately: we sample $\hat{N}$ policies from it, obtain actions $\mathbf{u}_i = \boldsymbol{\pi}_i(s')$, and evaluate their values $Q(s',\mathbf{u}_i,fd^*)$. The maximum over these values,

$$
\max_{i\in 1,\dots,\hat{N}} Q(s',\mathbf{u}_i,fd^*), \tag{12}
$$

serves as a practical approximation of $\max_{\mathbf{u}} Q(s',\mathbf{u},fd^*)$.

To further improve the accuracy of this estimation, instead of sampling arbitrary policies, we select the $\hat{N}$ nearest policies in the archive to the target descriptor fd^*. These neighborhood policies provide more relevant candidate actions for the maximization, yielding a tighter approximation:

$$
\max_{\mathbf{u}'} Q(s',\mathbf{u}',fd^*) \approx \max_{i\in\mathcal{N}_K(fd^*)} Q(s',\boldsymbol{\pi}_i(s'),fd^*), \tag{13}
$$

where $\mathcal{N}_K(fd^*)$ denotes the set of $\hat{N}$ nearest archive entries to descriptor fd^*.

4.4 Team-Based Cross Entropy Method

To more effectively leverage the teams of agents generated during the iterative process and to promote the emergence of higher-performing joint policies, we integrate the Cross-Entropy Method into the optimization loop in a multi-agent fashion.

In each iteration, $\tilde{N}$ teams are generated, where each team is parameterized by $\boldsymbol{\theta}_i = \{\theta_{i,1},\theta_{i,2},\dots,\theta_{i,N}\}$, with $\theta_{i,j}$ denoting the policy parameters of agent j in team i.

We then select the top $\frac{\tilde{N}}{2}$ teams based on their joint performance to update the CEM distribution. Let $\{\boldsymbol{\theta}_1,\dots,\boldsymbol{\theta}_{\tilde{N}/2}\}$ denote these top-performing teams. The CEM update refines the distribution over team parameters as:

$$
\mu_{\text{new}} = \frac{1}{\tilde{N}/2}\sum_{i=1}^{\tilde{N}/2} w_i\boldsymbol{\theta}_i, \quad \Sigma_{\text{new}} = \frac{1}{\tilde{N}/2}\sum_{i=1}^{\tilde{N}/2} w_i(\boldsymbol{\theta}_i-\mu_{\text{new}})(\boldsymbol{\theta}_i-\mu_{\text{new}})^\top, \tag{14}
$$

where μ_{new} and Σ_{new} denote the updated mean and covariance of the joint team parameter distribution, w_i is the weight coefficient, same as EvoRainbow [16].

This team-based CEM update ensures that the search distribution concentrates around high-performing regions of the multi-agent parameter space, while still maintaining sufficient diversity across teams to encourage the discovery of novel cooperative behaviors.

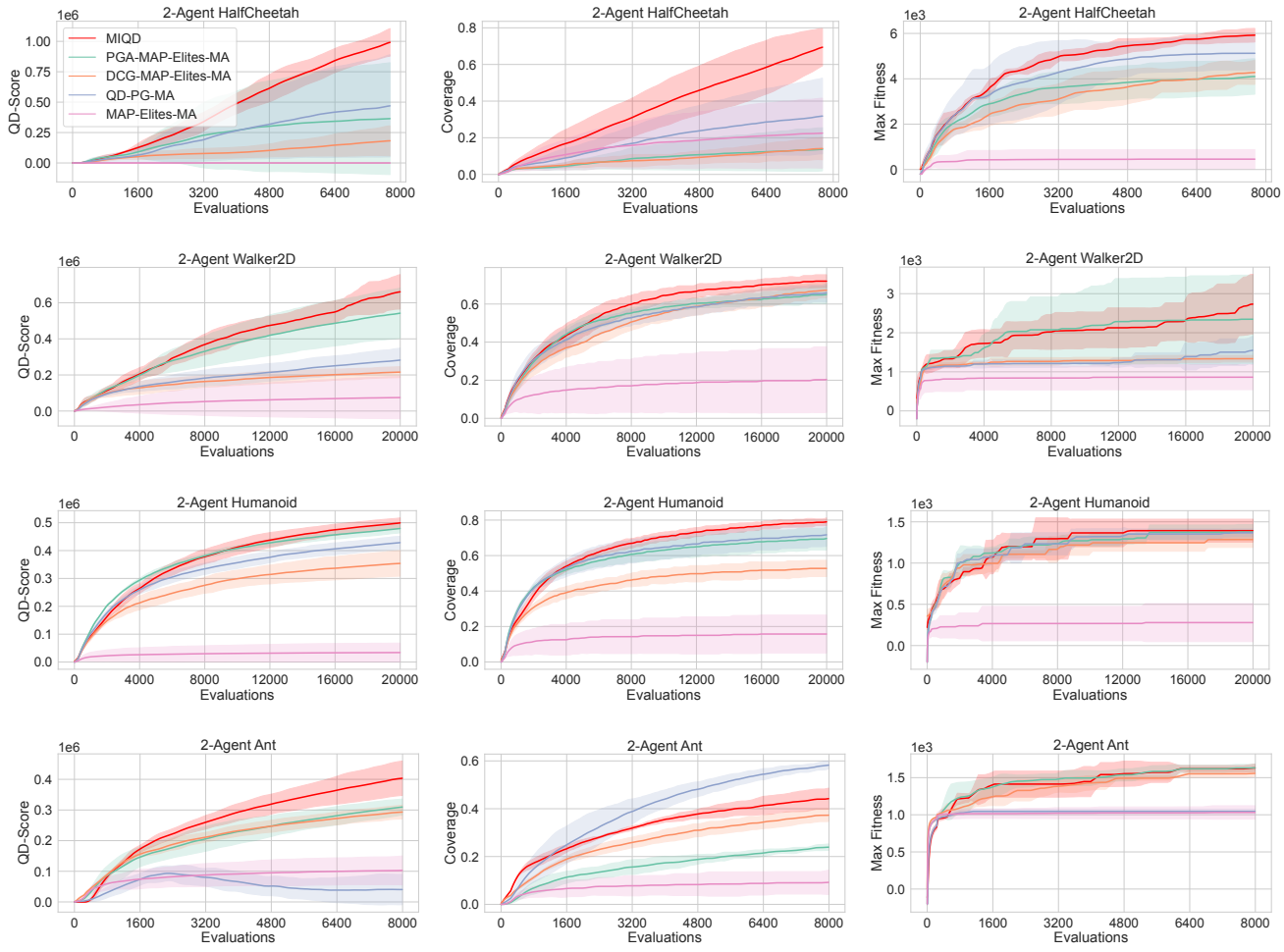


Figure 2: QD score, coverage and maximum fitness for MIQD and other baselines on MAMUJOCO tasks.

5 EXPERIMENTS

To evaluate our method, we consider four cooperative continuous control tasks from the MAMUJOCO benchmark: HalfCheetah, Ant, Walker, and Humanoid. These tasks cover a range of morphologies and control complexities, from planar running (HalfCheetah) to high-dimensional humanoid locomotion, providing a comprehensive testbed for assessing both performance and diversity. In each task, agents control different robot joints to achieve locomotion objectives such as standing, walking, or running. Each agent observes only its own joint states, making the tasks partially observable and requiring coordinated multi-agent control.

Agents must cooperate to achieve efficient locomotion while balancing speed and energy consumption. Policy fitness is measured as the cumulative reward over a full episode, according to MAMUJOCO's reward signals. The behavioral descriptor is defined as the proportion of time each foot is in contact with the ground during the episode:

$$\mathbf{BD} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{c}_t, \quad \mathbf{c}_t \in \{0,1\}^F, \tag{15}$$

where F denotes the number of feet, and $\mathbf{c}_t$ indicates whether each foot is in contact with the ground at timestep t (1 for contact, 0 for no contact). The BD thus represents the contact time ratio for each foot. The dimensionality of BD varies by task: 2 for HalfCheetah, 4 for Ant, and 2 for both Walker and Humanoid. Maintaining a quality-diversity archive adds some computational cost, which MIQD mitigates through parallelization techniques. In each iteration, 8 team policies are updated concurrently in separate processes with shared memory for efficient communication. As a result, training time remains comparable to mainstream MARL methods. Experiments are run on an Intel Xeon Platinum 8280 CPU (112 cores), providing sufficient capacity for archive-based updates.

5.1 Experimental settings

We compare MIQD against four representative QD baselines. Since most existing methods are designed for single-agent settings, we extend them to the multi-agent domain for a fair comparison: Map-Elites (MA), PGA-Map-Elites (MA), QD-PG (MA), and DCG-Map-Elites (MA). For gradient-based baselines, policy updates are performed using the MATD3 algorithm.

- **Map-Elites (MA)**: the classical QD algorithm that discretizes the BD space and stores the best-performing policy for each cell.
- **PGA-Map-Elites (MA)**: extends Map-Elites by applying policy gradient updates to sampled policies.
- **QD-PG (MA)**: leverages a diverse critic to encourage exploration of diverse state-space regions when updating policies.
- **DCG-Map-Elites (MA)**: conditions the critic on the full BD and introduces an intrinsic reward based on the distance between BDs.

In the original single-agent QD framework, offspring policies are generated through crossover operations between parent policies. However, in multi-agent scenarios, such a design is not directly applicable. We adapt the crossover mechanism to operate across agents in a team rather than exchanging neurons within individual policy networks [15], which better suits multi-agent coordination:

$$\begin{aligned} W_i', W_j' &= \left((W_i - W_i^{d_i}) \cup W_j^{d_j}, (W_j - W_j^{d_j}) \cup W_i^{d_i}\right) \\ &= \text{Crossover}(W_i, W_j), \end{aligned} \tag{16}$$

where W_i and W_j represent two chosen teams, and d_i and d_j correspond to randomly selected subsets of agent indices.

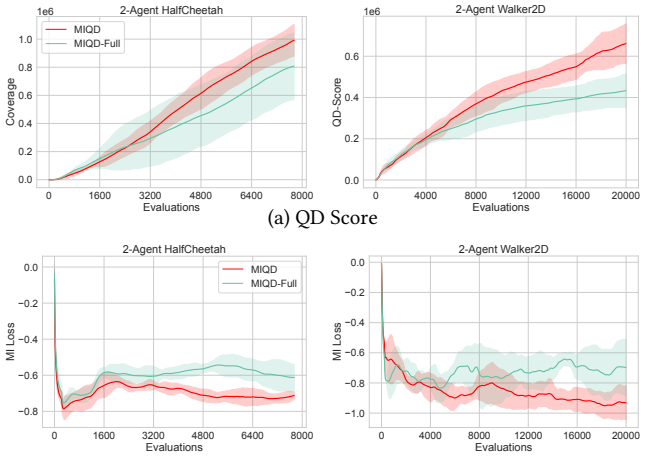
We evaluate all methods using three metrics:

- **QD-Score**: the aggregated fitness of all solutions stored in the archive, reflecting the overall quality of the population.
- **Coverage**: the proportion of archive cells occupied by at least one valid solution.
- **Max Fitness**: the highest fitness value among all solutions in the archive, representing the best-performing policy discovered.

5.2 Results

Figure 2 shows the performance of different algorithms across various tasks and evaluation metrics. The horizontal axis denotes the number of evaluation iterations, corresponding to how frequently team policies are evaluated during training. In each iteration, eight team policies are sampled for training. It can be observed that MIQD consistently achieves strong results across all four tasks and evaluation metrics, demonstrating its effectiveness. In particular, in the HalfCheetah task, MIQD significantly outperforms other methods across all metrics.

While QD-PG achieves high coverage in most tasks, its performance under the QD score metric is relatively poor. For example, in the Walker task, its coverage is comparable to PGA-Map-Elites-MA, but the QD score is lower. We attribute this to QD-PG focusing primarily on diversity rewards while neglecting performance. Even with the assistance of a quality critic, the combined QD score remains suboptimal.



(a) QD Score

(b) Loss of mutual information estimation

Figure 3: Ablation study of fragment behavioral descriptor.

In most tasks, MIQD outperforms PGA-Map-Elites-MA. This suggests that incorporating behavior-descriptor-based rewards into critic training enables the critic to better capture which actions align with the target BD. As a result, during policy gradient updates, agents are more effectively guided to produce behaviors consistent with the desired descriptors, enhancing overall diversity. Moreover, MIQD's use of mutual information to model the relationship between state–action pairs and BDs provides a more precise measure than simple BD-distance metrics, further improving performance across different behavioral dimensions.

5.3 Ablations

In the ablation study, we analyze the contributions of individual components within MIQD, including the fragment episode modeling, mutual information-based intrinsic reward, and the neighborhood polices in target value estimation. We first investigate the effect of fragment-based behavioral descriptors by comparing MIQD variants that use full-episode descriptors. Fragment descriptors mitigate the inaccuracies that occur when behavioral representations are derived from long trajectories. In our implementation, we set the fragment length to 50 timesteps. Using the Walker2D and HalfCheetah tasks, we evaluate how this design choice influences overall quality–diversity performance. As shown in Figure 3, incorporating fragment episodes consistently improves QD performance compared to using full episodes.

Additionally, we visualize the training loss of the mutual information discriminator across these tasks. We observe that using fragment BDs results in lower training loss, indicating that fragment episodes allow for a more precise characterization of behavioral descriptors, which in turn improves overall policy performance.

Next, we assess the effect of the mutual information–based reward by comparing MIQD with a variant that omits behavioral rewards. Using the coverage metric as shown in Figure 5, we find

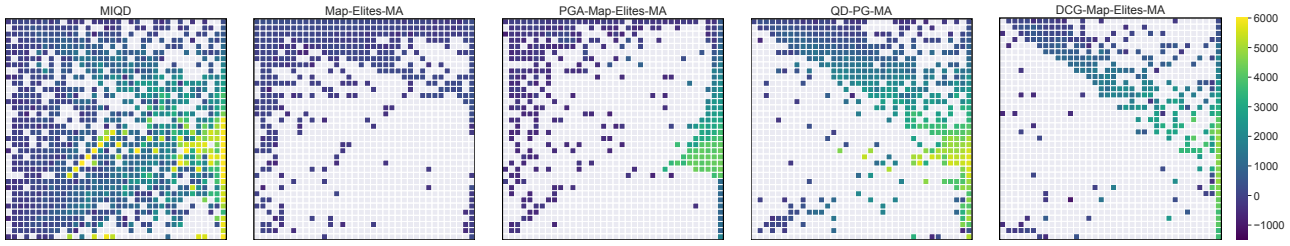


Figure 4: Visualization of the fitness distribution in the archive for different methods on the 2-Agent HalfCheetah task. The archive is represented as a two-dimensional grid, where white cells indicate uncovered regions.

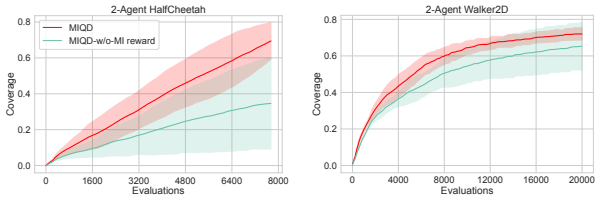


Figure 5: Ablation study of mutual-information-based reward.

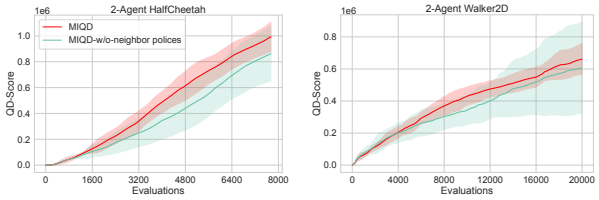


Figure 6: Ablation study of neighbor policies.

that MIQD achieves higher coverage than the baseline. This indicates that incorporating behavioral rewards to model the relationship between state–action pairs and BDs is critical for enhancing population diversity.

We then examine the impact of leveraging neighboring policies around the target descriptor's cell for target value estimation. As shown in Figure 6, this strategy substantially improves the QD-score. We attribute this improvement to the fact that neighboring policies are more likely to produce actions aligned with the target descriptor, providing a more stable and accurate training signal for the value function. Consequently, learning is easier, and overall performance is enhanced.

6 VISUALIZAITON

We visualize the distribution of solutions in the archive to examine how different methods balance performance and diversity. Specifically, we consider the 2-Agent HalfCheetah task, where the archive is two-dimensional. The visualizations reveal that baseline methods tend to concentrate solutions in the upper and right regions of the archive. Methods with policy gradient updates exhibit more yellow grids, indicating higher-performing solutions, while QD-PG shows particularly dense coverage in the upper-right corner due to its explicit diversity reward.

In contrast, MIQD achieves more uniform coverage across the archive, with fewer empty grids and a larger proportion of high-performing (yellow) cells. This demonstrates that MIQD enhances both performance and diversity. We attribute this improvement to the integration of mutual-information-based behavioral rewards, which more accurately capture the alignment between actions and behavioral descriptors while maintaining performance optimization.

7 CONCLUSION AND FUTURE WORK

To address the limitation of multi-agent reinforcement learning methods that prioritize performance at the expense of policy diversity, we propose MIQD, a quality–diversity approach that models multi-agent behavioral features using mutual information. To mitigate the inaccuracy of behavioral descriptors for individual state–action pairs in long trajectories, we introduce fragment BDs, which are incorporated into the critic to enable simultaneous optimization of agent performance and alignment with target behaviors. We model the relationship between actions and target BDs via mutual information, and since BDs are derived from multi-step state sequences, we decompose this information so that the resulting reward function captures how well single-step actions align with the target BD. Furthermore, to fully leverage high-performing individuals generated during training, we incorporate the Cross-Entropy Method and use a population-based approach to more accurately estimate target values during critic updates. We evaluate MIQD on multiple MAMUJOCO tasks and demonstrate strong performance across various metrics. Ablation studies confirm the contribution of each component in our method. In future work, we plan to explore alternative approaches for modeling reward functions and investigate the applicability of MIQD to a broader range of multi-agent reinforcement learning algorithms and settings.

ACKNOWLEDGMENTS

This paper is supported by the National Natural Science Foundation of China (NSFC) Youth Project for doctoral researchers, Grant No. 624B2101.

REFERENCES

[1] Johannes Ackermann, Volker Gabler, Takayuki Osa, and Masashi Sugiyama. 2019. Reducing overestimation bias in multi-agent domains using double centralized critics. *arXiv preprint arXiv:1910.01465* (2019).

[2] Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. 2019. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113* (2019).

[3] Antoine Cully, Jeff Clune, Danesh Tarapore, and Jean-Baptiste Mouret. 2015. Robots that can adapt like animals. *Nature* 521, 7553 (2015), 503–507.

[4] Agoston E Eiben and James E Smith. 2015. What is an evolutionary algorithm? In *Introduction to evolutionary computing*. Springer, 15–35.

[5] Maxence Faldor, Félix Chalumeau, Manon Flageat, and Antoine Cully. 2023. Map-elites with descriptor-conditioned gradients and archive distillation into a single policy. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 138–146.

[6] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[7] Matthew Fontaine and Stefanos Nikolaidis. 2021. Differentiable quality diversity. *Advances in Neural Information Processing Systems* 34 (2021), 10040–10052.

[8] Hu Fu, Yihua Tan, Hao Chen, and Pengyi Li. 2025. Tailoring Knowledge for Empowered Cooperative Actions in Multi-Agent Reinforcement Learning. *Neural Networks* (2025), 108023.

[9] Harsh Goel, Mohammad Omama, Behdad Chalaki, Vaishnav Tadiparthi, Ehsan Moradi Pari, and Sandeep Chinchali. 2025. R3DM: Enabling Role Discovery and Diversity Through Dynamics Models in Multi-agent Reinforcement Learning. *arXiv preprint arXiv:2505.24265* (2025).

[10] Shen Guicheng and Wang Yang. 2022. Review on dec-pomdp model for marl algorithms. In *Smart Communications, Intelligent Algorithms and Interactive Methods: Proceedings of 4th International Conference on Wireless Communications and Applications (ICWCA 2020)*. Springer, 29–35.

[11] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. 2018. MINE: mutual information neural estimation. *arXiv e-prints* (2018), arXiv–1801.

[12] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A. Al Sallab, Senthil Yogamani, and Patrick Pérez. 2022. Deep Reinforcement Learning for Autonomous Driving: A Survey. *IEEE Transactions on Intelligent Transportation Systems* 23, 6 (2022), 4909–4926. https://doi.org/10.1109/TITS.2021.3054625

[13] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. 2004. Estimating mutual information. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 69, 6 (2004), 066138.

[14] Chenghao Li, Tonghan Wang, Chengjie Wu, Qianchuan Zhao, Jun Yang, and Chongjie Zhang. 2021. Celebrating diversity in shared multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 34 (2021), 3991–4002.

[15] Pengyi Li, Jianye Hao, Hongyao Tang, Yan Zheng, and Xian Fu. 2023. Race: improve multi-agent reinforcement learning with representation asymmetry and collaborative evolution. In *International Conference on Machine Learning*. PMLR, 19490–19503.

[16] Pengyi Li, Yan Zheng, Hongyao Tang, Xian Fu, and Jianye Hao. 2024. Evorainbow: Combining improvements in evolutionary reinforcement learning for policy search. In *Forty-first International Conference on Machine Learning*.

[17] Shunyu Liu, Yihe Zhou, Jie Song, Tongya Zheng, Kaixuan Chen, Tongtian Zhu, Zunlei Feng, and Mingli Song. 2023. Contrastive identity-aware learning for multi-agent value decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 11595–11603.

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

[19] Sean Luke and Lee Spector. 1997. A comparison of crossover and mutation in genetic programming. *Genetic Programming* 97 (1997), 240–248.

[20] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[21] Olle Nilsson and Antoine Cully. 2021. Policy gradient assisted map-elites. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 866–875.

[22] Yiyuan Pan, Zhe Liu, and Hesheng Wang. 2025. Wonder Wins Ways: Curiosity-Driven Exploration through Multi-Agent Contextual Calibration. arXiv:2509.20648 [cs.LG] https://arxiv.org/abs/2509.20648

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

[24] Thomas Pierrot and Arthur Flajolet. 2023. Evolving populations of diverse RL agents with MAP-elites. *arXiv preprint arXiv:2303.12803* (2023).

[25] Thomas Pierrot, Valentin Macé, Felix Chalumeau, Arthur Flajolet, Geoffrey Cideron, Karim Beguir, Antoine Cully, Olivier Sigaud, and Nicolas Perrin-Gilbert. 2022. Diversity policy gradient for sample efficient quality-diversity optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 1075–1083.

[26] Justin K Pugh, Lisa B Soros, and Kenneth O Stanley. 2016. Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI* 3 (2016), 40.

[27] Roben Delos Reyes, Kyunghwan Son, Jinhwan Jung, Wan Ju Kang, and Yung Yi. 2022. Curiosity-Driven Multi-Agent Exploration with Mixed Objectives. *ArXiv* abs/2210.16468 (2022). https://api.semanticscholar.org/CorpusID:253237471

[28] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. 2017. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296* (2017).

[29] Simon Vanneste, Astrid Vanneste, Tom De Schepper, Siegfried Mercelis, Peter Hellinckx, and Kevin Mets. 2023. Distributed critics using counterfactual value decomposition in multi-agent reinforcement learning. In *Adaptive and Learning Agents Workshop (ALA), collocated with AAMAS, 29-30 May, 2023, London, UK*. 1–9.

[30] Vassilis Vassiliades, Konstantinos Chatzilygeroudis, and Jean-Baptiste Mouret. 2017. Using centroidal voronoi tessellations to scale up the multidimensional archive of phenotypic elites algorithm. *IEEE Transactions on Evolutionary Computation* 22, 4 (2017), 623–630.

[31] Tonghan Wang*, Jianhao Wang*, Yi Wu, and Chongjie Zhang. 2020. Influence-Based Multi-Agent Exploration. In *International Conference on Learning Representations*. https://openreview.net/forum?id=BJgy96EYvr

[32] Deheng Ye, Zhao Liu, Mingfei Sun, Bei Shi, Peilin Zhao, Hao Wu, Hongsheng Yu, Shaojie Yang, Xipeng Wu, Qingwei Guo, et al. 2020. Mastering complex control in moba games with deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 6672–6679.

[33] Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. 2022. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems* 35 (2022), 7103–7114.