LOD: LATENT OBJECTIVE DISCOVERY IN HETEROGENEOUS MULTI-AGENT REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In multi-agent reinforcement learning (MARL) settings, agents are often heterogeneous with diverse intrinsic utility functions. In this study, we introduce **L**atent **O**bjective **D**iscovery (**LOD**) as a framework for decomposing agents' distinct utilities into mixed underlying objectives in a low-dimensional latent space. Latent objective discovery aims to capture flexible and abstract coordination patterns among agents and each abstract objective is associated with a shared value function. During learning, all the learned values are aggregated by each agent according to their personalized weights learned in an adaptive manner. Therefore, agents inherit objective-level estimates for policy updates and value learning, enabling structured information sharing without requiring access to other agents' policies. Algorithmically, we incorporate our approach into Multi-Agent Proximal Policy Optimization (MAPPO) to exploit this structure. Theoretically, we establish convergence guarantees under linear function approximation within the actor-critic framework. Empirically, we extensively validate the advantage of introducing latent objective discovery in popular multi-agent testbeds with heterogeneous agents.

1 INTRODUCTION

Learning and understanding the *heterogeneous behaviors* of agents in multi-agent reinforcement learning (MARL) systems (Busoniu et al., 2008; Zhang et al., 2021; Albrecht et al., 2024) plays an increasingly crucial role in reliable deployment of reinforcement learning (RL) (Sutton et al., 1998). As a fundamental characteristic in MARL, heterogeneity (Bennett, 2024; Zhong et al., 2024) has recently been classified into five types: observation heterogeneity, response transition heterogeneity, effect transition heterogeneity, objective heterogeneity, and policy heterogeneity (Hu et al., 2026), where objective heterogeneity is associated with *agents' reward functions* and is the focus in this study. In many MARL applications, such as sensor networks (Rabbat & Nowak, 2004), smart grid (Dall'Anese et al., 2013; Zhang et al., 2018b), intelligent transportation systems (Adler & Blue, 2002; Zhang et al., 2018a), robotics (Corke et al., 2005), economics (Mazumdar et al., 2025; Tesauro & Kephart, 2002; Rădulescu et al., 2020), traffic signal control (Liu et al., 2023), and large language model fine-tuning (Ma et al., 2024b; Liu et al., 2025), agents are not perfectly homogeneous and often optimize compatible yet only partially aligned objectives. Such objectives may be unobserved and arise from differences in attributes, expertise, or preferences among agents, leading to intrinsically distinct or structured utilities, giving rise to objective heterogeneity in complex MARL systems.

Limitations of Existing MARL Paradigms for Objective Heterogeneity. Despite the increasing importance of accounting for heterogeneity, objective heterogeneity remains a fundamental yet underexplored challenge in existing MARL paradigms. As a canonical paradigm, centralized training with decentralized execution (CTDE) (Amato, 2024; Oliehoek et al., 2008; Rashid et al., 2020; Foerster et al., 2018; Mahajan et al., 2019; Du et al., 2019) trains centralized critics with access to global information, while allowing the agent to act independently. Nevertheless, CTDE methods are predominantly designed for fully cooperative tasks with *a shared global reward function* to simplify the analysis (Panait & Luke, 2005; Tuyls & Weiss, 2012; Zhang et al., 2021; Amato, 2024), and thus do not explicitly account for agents with distinct underlying objectives or preferences. Consequently, homogeneous MARL algorithms could have restricted performance and applicability (Zhong et al., 2024), and also pose critical challenges for multi-agent credit assignment (Chang et al., 2003; Foerster et al., 2018), i.e., identifying individual agents' contribution to the team's success. Moreover, fully decentralized MARL with networked agents (Varshavskaya et al., 2009; Zhang et al., 2018c; 2021)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

constitutes a complementary paradigm based on local centralization, accommodating heterogeneous rewards among agents under graph-structured interactions. However, the network topology deployed in this MARL paradigm is either explicitly pre-specified or heuristically defined by a local neighborhood among agents. We further provide an extended discussion of related work in Appendix A. These limitations expose a fundamental challenge in MARL:

How can we uncover objective-aware coordination structures among heterogeneous agents?

Our Contribution: Latent Objective Discovery (LOD) via Mixed Centralization. To characterize the objective heterogeneity among agents and the structure of their effective coordination, we aim to infer hidden objectives in a low-dimensional latent reward space, where each discovered objective represents a different reward-aligned behavioral abstraction, adaptive to agents' heterogeneity. The latent objectives are shared among agents with different connecting weights, adaptive to their distinct attributes, preferences, and utilities. This strategy of leveraging mixing weights is in the spirit of the widely adopted *mixed membership* concept in modeling heterogeneous generative densities (Fan et al., 2026; Shi, 2026), the underlying structure (Erosheva et al., 2004; Airoldi et al., 2014), and the social network (Airoldi et al., 2008; Graham, 2020; Jin et al., 2024). We realize latent objective discovery through *mixed centralization*, which can be viewed as a strict generalization of CTDE. Our LOD can also be viewed from the perspective of *networked agents*, where agents coordinate in a time-evolving network and heterogeneity of agents is naturally encoded in different clusters of the network. Unlike the classical networked MARL (Zhang et al., 2018c), the network topology induced in our paradigm is latent, non-local, task-adaptive and statistically principled. By enabling each agent to share multiple objectives, our method is capable of capturing more complicated underlying coordination patterns, contributing to role differentiation, modularity, and generalization in real-world multi-agent systems. Algorithmically, we introduce latent objective discovery into the state-of-the-art MARL algorithms, such as MAPPO. Our contributions can be summarized as follows:

- We introduce latent objective discovery into MARL with objective heterogeneity via a mixed centralization, adaptively capturing non-local and latent coordination patterns.
- We propose a practical actor-critic algorithm that effectively infers latent and overlapping objectives in MARL, with favorable theoretical properties.
- We empirically verify the benefits of latent objective discovery in MARL across various environments where agents inherently have distinct goals and incentives.

2 Our Method: Latent Objective Discovery (LOD) in MARL

Preliminaries. We study a Markov game (Littman, 1994), also known as a stochastic game (Shapley, 1953), which is defined by a tuple $\langle \mathcal{N}, \mathcal{S}, \{\mathcal{A}^i\}_{i\in\mathcal{N}}, P, \{R^i\}_{i\in\mathcal{N}}, \gamma\rangle$ with agents $i \in \mathcal{N} = \{1, ..., N\}$, the system state space $\mathcal{S}$, the joint action space $\mathcal{A} = \mathcal{A}^1 \times \ldots \times \mathcal{A}^N$, and the discount factor γ. $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the system state transition kernel and $R^i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the local (or shared) reward function for the agent i, giving the individual reward r_t^i at the time step t. The individual policy is denoted by $\pi^i : \mathcal{S} \to \Delta(\mathcal{A}^i)$ and the joint policy $\boldsymbol{\pi}$ is defined by $\boldsymbol{\pi}(\cdot|s_t) = \prod_{i=1}^N \pi^i(\cdot|s_t)$. We thus define the average and individual state-action value functions: $Q_{\text{ave}}^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = \mathbb{E}_{\boldsymbol{\pi}}\left[\frac{1}{N}\sum_{i=1}^N \sum_{t=0}^\infty \gamma^t r_t^i | s_0 = s, \boldsymbol{a}_0 = \boldsymbol{a}\right]$ and $Q_i^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{t=0}^\infty \gamma^t r_t^i | s_0 = s, \boldsymbol{a}_0 = \boldsymbol{a}\right]$ for $i \in \mathcal{N}$ given the system state s and the joint action $\boldsymbol{a} = (a^1, \ldots, a^N)$. Notably, we allow each agent to have its own reward function to accommodate the objective heterogeneity, which breaks the shared reward assumption and extends the cooperative settings commonly used in Dec-POMDP (Oliehoek et al., 2008; 2016). The network topology is not pre-specified as adopted in networked MARL (Zhang et al., 2018c) and we will construct it based on the agents' reward signals to discover their latent objectives. We assume full observability in the theoretical derivations for tractable results, but we compensate for partial observability in implementations by employing RNN that is commonly adopted by existing MARL work, e.g., QMIX (Rashid et al., 2020) and MAPPO (Yu et al., 2022).

2.1 Framework: Mixed Centralization for Latent Objectives

Mixed Centralization and Its Motivation. In the canonical CTDE paradigm, centralized critic methods, e.g., Lowe et al. (2017); Foerster et al. (2018); Yu et al. (2022), assume a single shared

critic function among agents and are seen as a hard global centralization. Such a globally shared critic, however, does not explicitly capture the natural heterogeneity in agents' underlying objectives, roles, and preferences. In contrast, we propose a mixed centralization regime that rests on multiple centralized critics, where each agent has its weights to *all* critics rather than being tied to a single centralized critic. The mixed centralization regime is rooted in a widely adopted concept called *mixed membership*. For example, according to people's purchase behaviors, they do not strictly belong to just one customer type. Instead, they are connected to core features of multiple consumer groups with distinct weights, such as tech enthusiasts, eco-conscious buyers, and budget-conscious shoppers.

Multiple Latent Objectives in MARL. The primary goal of mixed centralization is to discover multiple latent objectives in MARL, where agents typically possess different utility functions measured by distinct reward functions $\{R^i\}_{i\in\mathcal{N}}$. Mathematically, assume that there are K latent objectives among N agents and denote $\gamma_i(k) \geq 0$ as the weight for the agent i to the k-th latent objective, with $\sum_{k=1}^{K}\gamma_i(k) = 1$ for $\forall i \in \mathcal{N}$. The latent multiple objective paradigm in MARL is illustrated in Figure 1. Intuitively, the discrepancy among the rewards that agents receive reflects their underlying incentives and objectives in their coordination to achieve the pre-specified team goal. Here are some examples:

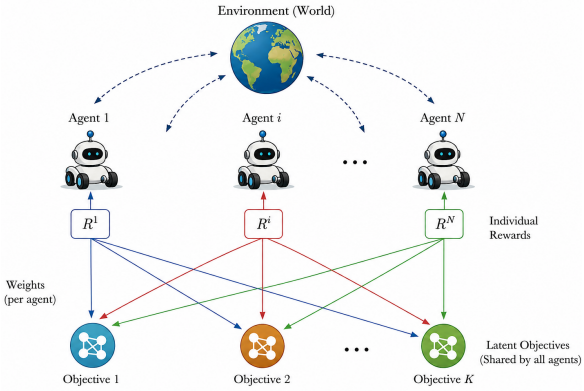


Figure 1: Illustration of Latent Objective Discovery.

① In cooperative navigation tasks, agents are required to cover areas 1 and 2 (objective 1) while avoiding collisions (objective 2). Thus, the global reward can be decomposed into these two objectives, where each agent's reward is viewed as a weighted sum to positively contribute to both objectives with the weights determined by their spatial positions and different attributes.

② In cooperative and competitive games with heterogeneous agents, incentives of each agent's behaviors can be grouped into three categories: maximizing the individual rewards, cooperating with allies, and suppressing opponents' advantages. These objectives are shared across agents but contribute differently to each agent due to distinct preferences, skills, and expertise.

③ In agentic RL for task-solving, LLM agents are often assigned specialized roles such as planner, executor, critic, verifier, or tool-user. These role-specific rewards reflect heterogeneous incentives while jointly contributing to the global task objective.

Mixed Centralization for Multiple Latent Objectives. We provide an algorithmic perspective to achieve latent objective discovery in the general actor-critic framework via mixed centralization. Denote state-action value functions $\{Q_i^{\boldsymbol{\pi}}\}_{i\in\mathcal{N}}$ and parameterized policies $\{\pi_{\theta_i}\}_{i\in\mathcal{N}}$. By the policy gradient theorem (Sutton et al., 1999), the policy gradient of each agent should depend on each critic $Q_i^{\boldsymbol{\pi}}$ to maximize their own objective $\mathcal{J}(\theta_i)$. The standard centralized critic method in CTDE replaces $Q_i^{\boldsymbol{\pi}}$ by a centralized (averaged) critic $Q_\omega^{\boldsymbol{\pi}}$ parameterized by ω, yielding the gradient of each π_{θ_i}:

$$\nabla_{\theta_i}\mathcal{J}\left(\theta_i\right) = \mathbb{E}\left[\nabla_{\theta_i}\log\pi_{\theta_i}\left(a^i|s\right)Q_\omega^{\boldsymbol{\pi}}\left(s,\boldsymbol{a}\right)\right]. \tag{1}$$

By contrast, our mixed centralization regime in principle requires K mixed centralized critics denoted by $\{Q_{\omega_i}^{\boldsymbol{\pi}}\}_{i=1}^{K}$ and their weighted sum forms the policy gradient:

$$\nabla_{\theta_i}\mathcal{J}_{\text{LOD}}\left(\theta_i\right) = \mathbb{E}\left[\nabla_{\theta_i}\log\pi_{\theta_i}\left(a^i|s\right)\left(\sum_{k=1}^{K}\gamma_i(k)Q_{\omega_k}^{\boldsymbol{\pi}}\left(s,\boldsymbol{a}\right)\right)\right], \tag{2}$$

where we replace the centralized critic $Q_\omega^{\boldsymbol{\pi}}$ in Eq. 1 by the weighted form $\sum_{k=1}^{K}\gamma_i(k)Q_{\omega_k}^{\boldsymbol{\pi}}$ to perform mixed centralization. To avoid degeneration, each $Q_{\omega_k}^{\boldsymbol{\pi}}$ should reflect distinct objectives shared among agents, which is specifically achieved by decomposing the reward signals in the latent space.

2.2 Reward Decomposition for Latent Objective Discovery

Latent Reward Decomposition. Let $\{\boldsymbol{R}_t\}_{t>0}$ be the N-dimensional time series with the random vector $\boldsymbol{R}_t = [R_t^1,\ldots,R_t^N]^\top \in \mathbb{R}^N$, where $R_t^i := R^i(s_t,\boldsymbol{a}_t)$ is the stochastic reward for the agent

i at the time step t. For an integer $K < N$, let $\gamma_i = [\gamma_i(1), \ldots, \gamma_i(K)]^\top \in \mathbb{R}^K$ be the weight vector for the agent i, forming a loading weight matrix $\Gamma = [\gamma_1, \ldots, \gamma_N]^\top \in \mathbb{R}^{N \times K}$. Let $\{\boldsymbol{F}_t\}_{t>0}$ be a K-dimensional covariance-stationary time series with $\boldsymbol{F}_t = [F_t^1, \ldots, F_t^K] \in \mathbb{R}^K$. Assume a population rank-K factor model (Lawley & Maxwell, 1962; Harman, 1976; Kline, 2014) on the reward time series $\{\boldsymbol{R}_t\}_{t>0}$ and there exists a *time-homogeneous* weight matrix Γ such that

$$\boldsymbol{R}_t = \Gamma \boldsymbol{F}_t + \boldsymbol{\epsilon}_t, \tag{3}$$

where $\{\boldsymbol{\epsilon}_t\}_{t>0}$ is the white noises. $\{\boldsymbol{F}_t\}_{t>0}$ are the latent K-dimensional factors and each latent time series $\{F_t^k\}_{t>0}$ for $k \in [K]$ corresponds to a distinct optimization signal shared across agents throughout the learning process, which we call a *latent objective*. Under the linear factor model assumption in Eq. 3, each agent's reward time series can be expressed as a linear combination of K latent objectives that jointly explain cross-agent reward variation over time.

Extension to Latent Critic Decomposition. Define the latent objective critic $\boldsymbol{Q}_{\boldsymbol{F}}^{\boldsymbol{\pi}}(s, \boldsymbol{a}) := \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{t=0}^{\infty} \gamma^t \boldsymbol{F}_t \mid s_0 = s, a_0 = \boldsymbol{a}\right] \in \mathbb{R}^K$ and denote $\boldsymbol{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = [Q_1^\pi(s, \boldsymbol{a}), \ldots, Q_N^\pi(s, \boldsymbol{a})]^\top \in \mathbb{R}^N$. Proposition 2.1 shows that latent reward decomposition in Eq. 3 implies a latent critic decomposition with the transitivity of the Γ-transformation from the reward space to the state-action value function space. The proof is provided in Appendix B.1.

Proposition 2.1 (Latent Critic Decomposition)**.** *By latent reward decomposition in Eq. 3, we have*

$$\boldsymbol{Q}^\pi(s, \boldsymbol{a}) = \Gamma \boldsymbol{Q}_{\boldsymbol{F}}^{\boldsymbol{\pi}}(s, \boldsymbol{a}). \tag{4}$$

Next, by applying the function approximation, the k-th element of $\boldsymbol{Q}_{\boldsymbol{F}}^{\boldsymbol{\pi}}$ is parameterized by Q_{ω_k}. Proposition 2.1 naturally leads to the mixed centralization of policy gradient in Eq. 2 of Section 2.1:

$$Q_i^{\boldsymbol{\pi}}(s, \boldsymbol{a}) \approx \sum_{k=1} \gamma_i(k) Q_{\omega_k}(s, \boldsymbol{a}). \tag{5}$$

Online Estimation of Γ and $\boldsymbol{F}_t$. Next, we perform an online estimation of Γ and $\boldsymbol{F}_t$ against the non-stationarity due to the constantly updated RL policies. Firstly, we perform rollout to collect the trajectory $\{s_t, \boldsymbol{a}_t, \boldsymbol{r}_t, s_{t+1}\}_{t=1}^T$ with the length T, where $\boldsymbol{r}_t = [r_t^1, \ldots, r_t^N]^\top \in \mathbb{R}^N$ is the N-dimensional observed reward vector. We denote $\boldsymbol{r}_{1:T} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_T] \in \mathbb{R}^{N \times T}$ as the observed reward matrix and the latent objective matrix $\boldsymbol{f} = [\boldsymbol{f}_1, \ldots, \boldsymbol{f}_T] \in \mathbb{R}^{K \times T}$ with $\boldsymbol{f}_t = [f_t^1, \ldots, f_t^K]^\top \in \mathbb{R}^K$. Estimating the loading matrix Γ and the latent objective $\boldsymbol{f}$ reduces to a rank-K low-rank approximation problem with convex constraints. The optimization is expressed as:

$$\min_{\Gamma, \boldsymbol{f}} \|\boldsymbol{r}_{1:T} - \Gamma \boldsymbol{f}\|_F^2 + \lambda \|\boldsymbol{f}\|_F^2 \quad \text{s.t. } \Gamma \geq 0, \Gamma \mathbf{1} = \mathbf{1}. \tag{6}$$

The solution is reminiscent of the truncated SVD and Principal Component Analysis (PCA). In practice, we can perform alternating update. By fixing $\boldsymbol{f}$, updating Γ is equivalent to solving simplex-constrained least squares $\min_{\Gamma \in \Delta^{K-1}} \|\boldsymbol{r}_{1:T} - \Gamma \boldsymbol{f}\|_F^2$. By fixing Γ, updating $\boldsymbol{f}$ reduces to a ridge-regularized least-squares estimation with the closed-form solution: $\boldsymbol{f} = (\Gamma^\top \Gamma + \lambda \mathbf{I})^{-1} \Gamma^\top \boldsymbol{r}_{1:T}$. The estimation of Γ and $\boldsymbol{F}_t$ can be performed online with Γ updated according to $\Gamma_{t+1} = (1 - \tau)\Gamma_t + \tau \Gamma$, where Γ is the newly estimated matrix via Eq. 6 and $\tau \in [0, 1]$ controls the update rate. This estimation procedure is a key component of the LOD module and supports its integration into existing MARL algorithms, e.g., MAPPO described in Section 2.3.

2.3 Algorithm

LOD is a general, modularized component that can be integrated into a wide range of MARL algorithms. In this study, we use MAPPO (Yu et al., 2022) as our primary instantiation, given its widespread adoption and strong empirical performance across diverse benchmarks (Papoudakis et al., 2021). We then discuss how LOD can be incorporated into a broader class of actor–critic algorithms.

LOD-MAPPO Algorithm. We parameterize K state-value functions $\{V_{\omega_k}\}_{k=1}^K$ instead of $\{Q_{\omega_k}\}_{k=1}^K$ to facilitate Generalized Advantage Estimation (GAE) (Schulman et al., 2015b). We also denote o_t^i as the observation for agent i at the time t to accommodate partial observability. After performing the online estimation in Eq. 6, we discover the latent objective signal f_t^k in the k-th objective for each step t and then we can define the Temporal-Difference (TD) error δ_t^k for V_{ω_k}: $\delta_t^k = f_t^k + \gamma V_{\phi_k}(s_{t+1}) - V_{\phi_k}(s_t)$. Consequently, the k-th latent advantage estimate is $\hat{A}_t^k = \sum_{\ell=0}^{T-t} (\gamma\lambda)^\ell \delta_{t+\ell}^k$

Algorithm 1: MAPPO Algorithm with Latent Objective Discovery (on-policy)

Input: Episodes L, batch size B, GAE parameter λ, steps per episode T, number of epochs M
Initialize: policy networks $\{\pi_{\theta_i^0}\}_{i\in\mathcal{N}}$, value networks $\{V_{\phi_k^0}\}_{k=1}^K$, rollout buffer $\mathcal{B}$
for $l = 1, 2, \ldots, L$ **do**
  Reset the environment and obtain the initial state s_1 and observations $\{o_1^i\}_{i\in\mathcal{N}}$.
  Collect a set of trajectories by running the joint policy $\boldsymbol{\pi}_{\theta^{l-1}} = (\pi_{\theta_1^{l-1}}, \ldots, \pi_{\theta_N^{l-1}})$
  Push transitions $\{s_t, \{o_t^i\}_{i\in\mathcal{N}}, \{a_t^i\}_{i\in\mathcal{N}}, \{r_t^i\}_{i\in\mathcal{N}}, s_{t+1}, \{o_{t+1}^i\}_{i\in\mathcal{N}}\}_{t=1}^T$ into $\mathcal{B}$
  Update Γ and decompose $\boldsymbol{f}$ by solving Eq. 6 based on $\boldsymbol{r}_{1:T}$. # Latent Objective Discovery.
  # Mixed Centralized Advantage Function via GAE.
  for $k = 1, \ldots K$ **do**
    $\hat{A}_t^k = \sum_{\ell=0}^{T-t}(\gamma\lambda)^\ell \delta_{t+\ell}^k$, and $\delta_t^k = f_t^k + \gamma V_{\phi_k^{l-1}}(s_{t+1}) - V_{\phi_k^{l-1}}(s_t)$.
  end for
  for $m = 1, \ldots, M$ **do**
    Sample a random minibatch of B transitions from $\mathcal{B}$
    # Actor update.
    for $i \in \mathcal{N}$ **do**
      Update each policy by maximizing the PPO-Clip objective with respect to θ_i:

$$\pi_{\theta_i^l} \leftarrow \arg\max_{\pi_{\theta_i}} \frac{1}{T}\sum_{t=0}^{T} \min\left(\rho_t(\theta_i)\bar{A}_t^i,\ \text{clip}(\rho_t(\theta_i), 1-\epsilon, 1+\epsilon)\bar{A}_t^i\right),$$

      where $\rho_t(\theta_i) = \frac{\pi_{\theta_i}(a_t^i|o_t^i)}{\pi_{\theta_i^{l-1}}(a_t^i|o_t^i)}$ and $\bar{A}_t^i = \sum_{k=1}^K \gamma_i(k)\hat{A}_t^k$.
    end for
    # Critic update.
    for $k = 1, \ldots, K$ **do**
      Compute the GAE return for the k-th objective by $\hat{R}_t^k = \hat{A}_t^k + V_{\phi_k^{l-1}}(s_t)$
      Update the value function by $V_{\phi_k^l} \leftarrow \arg\min_{V_{\phi_k}} \frac{1}{T}\sum_{t=1}^T \left[(V_{\phi_k}(s_t) - \hat{R}_t^k)^2\right]$
    end for
  end for
  Clear the rollout buffer $\mathcal{B}$
end for

via GAE for $k \in [K]$. For the agent i, we denote the personalized advantage estimate by $\bar{A}_t^i$, which is the linear combination between $\hat{A}_t^k$ and γ_i: $\bar{A}_t^i = \sum_{k=1}^K \gamma_i(k)\hat{A}_t^k$. The algorithmic description of LOD-MAPPO is provided in Algorithm 1. To improve memory and computational efficiency, we employ a multi-head with a shared backbone $h_\psi(s_t)$, with $V_{\omega_k}(s_t) = g_{\omega_k}(h_\psi(s_t))$ for $k \in [K]$.

Broader Actor-Critic Algorithms. Our latent objective discovery framework can accommodate a much broader actor-critic algorithm class, including both on-policy and off-policy branches. In Appendix C.2, we include a generic on-policy learning framework in Algorithm 3 that can also instantiate trust-region objective in TRPO (Schulman et al., 2015a). We also include an off-policy learning framework in Algorithm 4, where we reproject sampled rewards from the replay buffer to decompose them into latent objectives. The TD target Q_k can be instantiated using either the target network in DDPG (Lillicrap et al., 2015) or clipped double Q-learning in TD3 (Fujimoto et al., 2018). We also present a generic actor-critic algorithm in the average reward setting in Algorithm 2 of Appendix C.1, which is the focus for the theoretical analysis in Section 3.

Extension to Shared Global Rewards via Agent-Specific Advantage Surrogates. In many MARL applications, agents only have access to sparse and delayed rewards, such as a shared and trajectory-level reward when the episode completes, even though they may perform distinct roles and contribute differently to the final outcome. Without the heterogeneous rewards to perform LOD, we can still leverage the agent-specific local advantage function as a surrogate to measure the contribution of agent i's action relative to its local alternatives, based on which we perform LOD method. This strategy is also commonly adopted by Foerster et al. (2018); Zhang et al. (2018c); Zhong et al. (2024),

as the surrogate local reward signal. Particularly, the agent-specific advantage function is defined by

$$A_i^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = Q^{\boldsymbol{\pi}}(s, \boldsymbol{a}) - \sum_{\tilde{a}^i \in \mathcal{A}^i} \pi^i(\tilde{a}^i|s) Q^{\boldsymbol{\pi}}(s, (\tilde{a}^i, a^{-i})), \tag{7}$$

where a^{-i} is the action vector of all agents except a^i. Inspired by *difference rewards* (Wolpert & Tumer, 2001), this agent-specific advantage also has the same form of *counterfactual baseline* (Foerster et al., 2018) and *aristocrat utility* (Wolpert & Tumer, 2001) to represent their heterogeneous incentives. We validate the efficacy of the local advantage surrogate in Section 4.3 of our experiments.

3 Theoretical Results

For theoretical analysis, we consider the average-reward setting (Puterman, 2014; Marbach & Tsitsiklis, 2002; Abounadi et al., 2001; Zhang et al., 2018c; Chen & Zhao, 2023; Wu et al., 2020), where our policy optimization objective is $\max_\theta J(\theta) = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left(\sum_{t=0}^{T-1} \frac{1}{N}\sum_{i\in\mathcal{N}} r_{t+1}^i\right)$. Specifically, our theoretical results focus on the average-reward setting with a generic actor-critic algorithm, which we defer to Algorithm 2 in Appendix C. Under standard ergodicity assumptions, this average reward setting coincides with the normalized discounted reward setting in the limit as the discount factor γ approaches one, and the corresponding policy gradients agree in this limit, providing theoretical justification for the discounted reward setting in our LOD.

Assumption 3.1. *The latent factor f_t^k and the instantaneous reward r_t^i are uniformly bounded over all $1 \le k \le K, 1 \le i \le N$ and $t \ge 0$.*

This assumption on the latent factor model is standard and mild. If considering finite state and action spaces, it is automatically satisfied.

Assumption 3.2. *The update of the policy parameter θ_t^i includes a local projection operator $\mathcal{L}_0^i : \mathbb{R}^{m_i} \to \Theta^i \subset \mathbb{R}^{m_i}$, projecting any θ_t^i onto the compact set Θ^i. Additionally, we assume $\Theta = \prod_{i=1}^N \Theta^i$ is large enough to include at least one local minimum of $J(\theta)$.*

This projection assumption is commonly used and standard in many analyses for actor-critic algorithms in stabilizing the stochastic approximation. See Zhang et al. (2018c); Bhatnagar et al. (2007); Degris et al. (2012); Prasad et al. (2014).

Assumption 3.3. *The stepsizes $\eta_{\omega,t}$, $\eta_{\theta,t}$ satisfy $\sum_t \eta_{\omega,t} = \sum_t \eta_{\theta,t} = \infty$, $\sum_t \eta_{\omega,t}^2 + \eta_{\theta,t}^2 < \infty$. Moreover, $\eta_{\theta,t} = o(\eta_{\omega,t})$.*

Here, the critic step updates faster than the actor step, which behaves like a two-time-scale algorithm as standard single-agent actor-critic algorithms.

Assumption 3.4. *The online weight estimator γ_{it} is consistent in t, i.e., the weight vector $\gamma_{it} \to \gamma_i \in \mathbb{R}^K$ for all $i \in \mathcal{N}$.*

This assumption provides generality rather than restriction: our analysis accommodates any consistent estimator, including standard factor-based estimators under regularity conditions in Eq. 6.

Assumption 3.5. *The Q-function is parametrized as $Q_\omega(s,a) = \omega'\phi(s,a)$, where $\phi(s,a) = [\phi_1(s,a), \ldots, \phi_{m_\Omega}(s,a)]'$ is some feature vector associated with (s,a). Moreover, $\phi(s,a)$ is uniformly bounded for any (s,a). Define the feature matrix $\Phi \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|\times m_\Omega}$ whose m-th column is equal to $[\phi_m(s,a), s \in \mathcal{S}, a \in \mathcal{A}]'$ for $1 \le m \le m_\Omega$. Then, Φ is assumed to have full column rank and $\Phi u \ne \mathbf{1}$ for any $u \in \mathbb{R}^{m_\Omega}$.*

Here, for the convenience of theoretical analysis, we only focus on the linear function approximation. Actually, under nonlinear function approximation, TD-learning-based policy evaluation may fail to converge (Tsitsiklis & Van Roy, 1996). Based on such assumptions, most of the existing online actor-critic algorithms establish their theoretical convergence guarantees (Konda & Tsitsiklis, 1999; Bhatnagar et al., 2007; Bhatnagar, 2010; Zhang et al., 2018c). As for the assumption on the feature matrix Φ, it is also commonly adopted in many studies Bhatnagar et al. (2007); Tsitsiklis & Van Roy (1996; 1999); Zhang et al. (2018c).

Theorem 3.1 (Convergence of the critic step)**.** *Under Assumptions 3.1 and 3.3 to 3.5, for any policy π_θ given, the factor parameter $\omega_t^{(k)}$ converges: $\lim_t \omega_t^{(k)} = \omega_\theta$ almost surely and the limit ω_θ is the unique solution to*

$$\Phi' D_\theta^{s,a}(T_\theta^{Q,(k)}(\Phi\omega_\theta^{(k)}) - \Phi\omega_\theta^{(k)}) = 0,\ 1 \le k \le K.$$

For clarity, the full definitions of $D_\theta^{s,a}$, the Bellman operator $T_\theta^{Q,(k)}$ are provided in Appendix B.2.

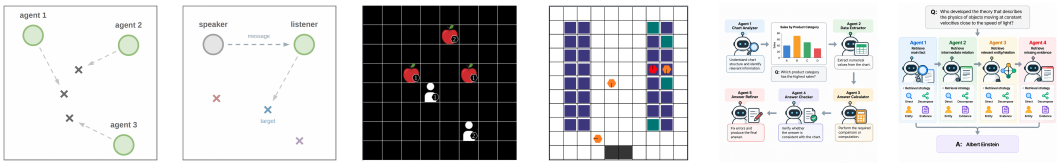
Theorem 3.2 (Convergence of the actor step)**.** *Under Assumptions 3.1 to 3.5, the agent policy parameter θ_t^i converges almost surely to a point in the set of the asymptotically stable equilibria of*

$$\dot{\theta}^i = \mathcal{L}^i(\mathbb{E}_{s_t \sim d_\theta, a_t \sim \pi_\theta}(A_{t,\theta}^i \psi_{t,\theta}^i)),\ i \in \mathcal{N}.$$

The definitions of $A_{t,\theta}^i$, $\psi_{t,\theta}^i$, and the projection operator $\mathcal{L}^i(\cdot)$ are provided in Appendix B.2.

4 Experiment

In this section, we evaluate and analyze the effect of latent objective discovery via LOD-MAPPO on an array of multi-agent benchmarks with heterogeneous rewards: Multi-Agent Particle Environment (MPE) (Lowe et al., 2017; Mordatch & Abbeel, 2018), Level-Based Foraging (LBF) (Albrecht & Ramamoorthy, 2013; Albrecht & Stone, 2017), Multi-Robot Warehouse (RWARE) (Christianos et al., 2020), Agentic RL tasks in Agent Lightning (Masry et al., 2022; He et al., 2026), including ChartQA and MuSiQue. These benchmarks are diverse in task difficulty, agent number, action type, observation and action space dimensionality, and cooperation strategy required, and hence provide a comprehensive assessment of the effectiveness, stability, robustness, and generality of our methods. The selected benchmarks are illustrated in Figure 2 and detailed descriptions of each benchmark are provided in Appendix D. Our theoretical derivations assume full observability, a common model choice in (Zhang et al., 2018c; Zhong et al., 2024), but our experiments compensate for partial observability by conditioning on past observations as adopted in MAPPO (Yu et al., 2022), QMIX (Rashid et al., 2020), and COMA (Foerster et al., 2018).



(a) Spread (b) Speaker–Listener (c) LBF (d) RWARE (e) ChartQA (f) MuSiQue

Figure 2: Illustrations of the selected MARL benchmarks for testing LOD-MAPPO.

4.1 Multi-Agent Particle Environment (MPE)

Environmental Setup. We consider two fully cooperative tasks in MPE across continuous and discrete action settings: `simple_spread` and `Speaker_listener`. (1) `simple_spread`: a cooperative game with N agents to cover N landmarks without collision, which implies two intrinsic objectives. We choose $N = 3$ and $K = 2$ in the main experiment and we also perform sensitivity analysis across various N and K; (2) `Speaker_listener`: one static speaker has to communicate

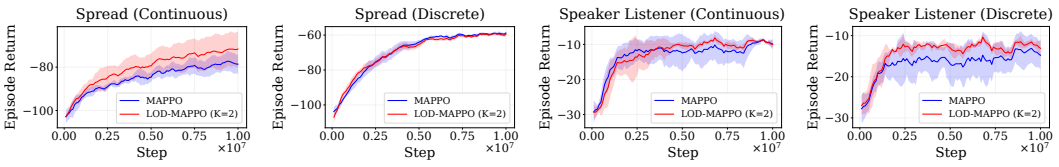


Figure 3: Comparison between LOD-MAPPO ($K = 2$) and MAPPO in two Multi-Agent Particle Environments (MPE) across continuous and discrete action settings over five random seeds.

a goal landmark to a listener capable of moving. We choose $N = 2$ and $K = 2$. Our implementations are adapted from the widely adopted HARL codebase from (Zhong et al., 2024).

Main Results. Figure 3 showcases that LOD-MAPPO outperforms MAPPO across two MPE tasks with both continuous and discrete action spaces, except for Spread (Discrete), where LOD-MAPPO is on par with MAPPO. In `simple_spread`, the reward combines a shared coverage component with agent-specific collision penalties. Our LOD method can exploit this structure by learning latent reward factors and reconstructing each agent's advantage through agent-specific mixing weights, which tend to improve value estimation and policy optimization. In `Speaker_listerner`, the speaker and the listener have complementary roles. The intermediate-stage gains in the continuous setting and the more persistent improvement in the discrete setting therefore demonstrate an empirical benefit of LOD. In Figure 6 of Appendix D.1, we vary N between 4 and 6, and a wide range of τ used in the soft updating of Γ on Spread (Continuous). The benefits of LOD-MAPPO are similar and further discussion is provided in Appendix D.1.

4.2 Level-Based Foraging (LBF) and Multi-Robot Warehouse (RWARE)

Environmental Setup. We evaluate LOD-MAPPO on Level-Based Foraging (LBF) and the Multi-Robot Warehouse (RWARE), two environments requiring coordination under sparse rewards. (1) `Level-Based Foraging (LBF)`: agents navigate a grid and collect items when their combined levels meet or exceed the item's level. Rewards are heterogeneous across agents, and the cooperative variant (denoted by `LBF-Coop`) requires joint collection, further increasing the difficulty of obtaining reward. We consider both variants on a 15×15 grid. (2) `Multi-Robot Warehouse (RWARE)`: partially observable robots must collect requested shelves and deliver them to designated locations. Rewards occur only upon successful delivery, requiring extended sequences of coordinated actions. We evaluate the tiny and small layouts. All tasks use $N = 4$ agents, and results are averaged over five random seeds. Our implementations are directly adapted from the EpyMARL framework (Papoudakis et al., 2021) with the same hyperparameters as MAPPO.

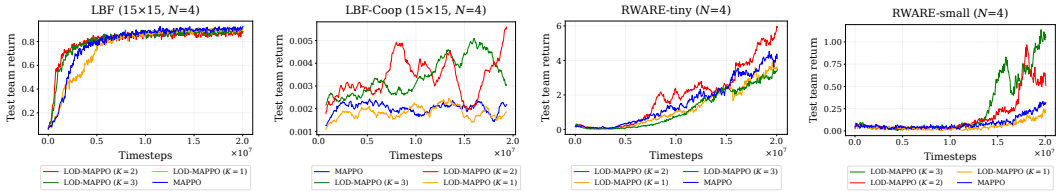


Figure 4: Comparison between LOD-MAPPO across a wide range of K and MAPPO in LBF and RWARE environments. The learning curves are the mean returns across five random seeds.

Main Results. Figure 4 shows the advantages of LOD-MAPPO over MAPPO, where the learning curves are the mean averaged across five random seeds to facilitate comparison. In standard LBF, LOD-MAPPO with $K = 2$ or $K = 3$ improves more rapidly during early training, while all methods eventually attain similar returns. In cooperative LBF, LOD-MAPPO with $K = 2$ or $K = 3$ achieves higher returns over substantial portions of training than MAPPO. Nevertheless, absolute returns remain low and the curves fluctuate considerably, indicating that sparse cooperative rewards remain a major challenge. In RWARE-tiny and RWARE-small, LOD-MAPPO with $K = 2$ and $K = 3$ achieves the highest final mean return, respectively. These results suggest that multiple latent reward factors can provide useful structure for learning from sparse, agent-dependent reward signals. By sharing latent value or advantage estimates while retaining agent-specific mixing weights, LOD may facilitate the use of limited reward information across agents. In addition, the optimal choice of K is task-dependent and may require task-specific tuning. The learning curves with shaded bands indicating variations are provided in Figure 7 in Appendix D.2.

4.3 Agentic RL Settings

Environmental Setup. In agentic RL, the environment that agents interact with often encodes a shared and sequence-level reward, even though they may perform distinct roles and contribute differently to the final outcome. Therefore, we deploy the local advantage alternative via Eq. 7 to characterize

objective heterogeneity for our latent objective discovery approach. We consider `ChartQA` and `MuSiQue Multi-Hop Retrieval` as two agentic RL tasks from Agent Lightning (Masry et al., 2022; He et al., 2026) and use Qwen2.5 (Hui et al., 2024) in policy learning. Agents occupy distinct functional roles and make heterogeneous strategy decisions, naturally motivating LOD to uncover the latent objectives shared across all agents. All agents receive a shared binary task reward, termed *Success*. In `ChartQA`, the reward is 1 if the workflow successfully produces the final answer after checking and refinement and zero otherwise, while in `MuSiQue`, success indicates the required supporting information is successfully retrieved and zero otherwise. We report the moving-average success rate during training. More experimental details are provided in Appendix D.3.

① **ChartQA:** The workflow consists of five successive roles: *Chart Analyzer*, which interprets the chart; *Data Extractor*, which extracts the relevant values; *Answer Calculator*, which performs the required computation; *Answer Checker*, which verifies the result; and *Answer Refiner*, which produces the final answer after correction. We treat these five roles as cooperating heterogeneous agents and each role selects among four role-specific strategies to make different types of decisions while jointly determining the task outcome.

② **MuSiQue Multi-Hop Retrieval.** It consists of questions that require combining information from multiple sources. The workflow consists of four successive retrieval steps, which we treat as cooperating agents. At each step, the corresponding agent selects among four retrieval strategies: *Direct*, which searches for the main fact; *Decompose*, which targets an unresolved intermediate relation; *Entity*, which focuses on a relevant entity and relation; and *Evidence*, which searches for information missing from the available evidence. The selected strategy is used to generate a search query and retrieve supporting information.

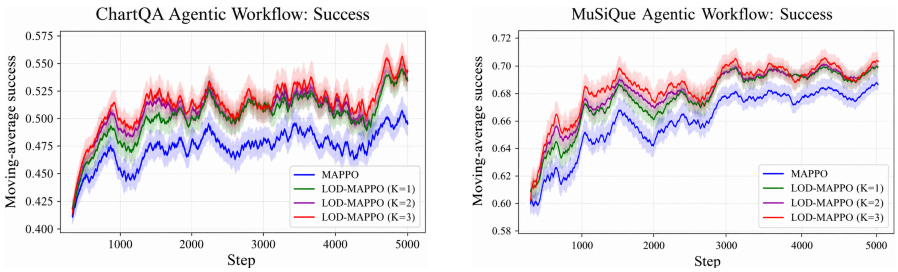


Figure 5: Comparison between LOD-MAPPO and MAPPO in two Agent Lightning environments.

Main Results. Figure 5 shows that LOD-MAPPO with diverse K exhibits faster improvement during the early training stage, indicating that learning the latent structure via LOD benefits the training when agents are still learning their roles under a shared reward. As training progresses, LOD-MAPPO maintains a consistent advantage over MAPPO, suggesting that latent objective learning continues to support coordination among heterogeneous agents. Increasing the number of latent objectives generally leads to better performance, but the benefit gradually saturates as the policies converge. The optimal choice of K may depend on the task, which requires prior knowledge.

5 Conclusion and Limitations

In this study, we introduce latent objective discovery as a new MARL framework that extends existing CTDE and networked MARL formulations by adaptively uncovering shared latent structure in agents' heterogeneous objectives. Inspired by the mixed membership concept, the framework learns shared latent objectives through multiple mixed centralized critics and captures each agent's relationship to these objectives through personalized weights. These weights offer an interpretable characterization of agents' objective preferences and support personalized policy learning in broader MARL settings.

Although the framework is, in principle, applicable to cooperative, competitive, and mixed settings (Amato, 2024), our empirical evaluation focuses on cooperative tasks. Evaluating its effectiveness in competitive and mixed settings and establishing theoretical guarantees under nonlinear function approximation remain important directions for future work. Our study also focuses specifically on objective heterogeneity and extending latent objective discovery to accommodate other forms of heterogeneity in MARL warrants further investigation.

Ethics Statement

This work studies multi-agent reinforcement learning in simulated environments. Improved coordination does not guarantee safe or equitable outcomes. Real-world deployment requires careful reward design and evaluation of safety, robustness, and impacts on the interaction of agents' behaviors in safety-critical scenarios.

Reproducibility Statement

Our implementations are directly adapted from the widely adopted MARL frameworks, including HARL (Zhong et al., 2024) and EpyMARL (Papoudakis et al., 2021). Detailed experimental descriptions are provided in Appendix D to facilitate reproducibility. The code is also submitted in the supplementary materials.

Statement on the Use of AI

In this work, we used generative AI tools to help guide and refine the experiment implementations and polish the writing of this study. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Jinane Abounadi, Dimitrib Bertsekas, and Vivek S Borkar. Learning algorithms for markov decision processes with average cost. *SIAM Journal on Control and Optimization*, 40(3):681–698, 2001.

Jeffrey L Adler and Victor J Blue. A cooperative multi-agent transportation management and route guidance system. *Transportation Research Part C: Emerging Technologies*, 10(5-6):433–454, 2002.

Edo M Airoldi, David Blei, Stephen Fienberg, and Eric Xing. Mixed membership stochastic blockmodels. *Advances in neural information processing systems*, 21, 2008.

Edoardo M Airoldi, David M Blei, Elena A Erosheva, and Stephen E Fienberg. Introduction to mixed membership models and methods. *Handbook of mixed membership models and their applications*, 100:3–14, 2014.

Stefano V Albrecht and Subramanian Ramamoorthy. A game-theoretic model and best-response learning method for ad hoc coordination in multiagent systems. *International Conference on Autonomous Agents and Multi-Agent Systems*, 2013.

Stefano V Albrecht and Peter Stone. Reasoning about hypothetical agent behaviours and their parameters. *International Conference on Autonomous Agents and Multi-Agent Systems*, 2017.

Stefano V Albrecht, Filippos Christianos, and Lukas Schäfer. *Multi-agent reinforcement learning: Foundations and modern approaches*. MIT Press, 2024.

Christopher Amato. An introduction to centralized training for decentralized execution in cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2409.03052*, 2024.

Michel Benaïm. Dynamics of stochastic approximation algorithms. In *Seminaire de probabilites XXXIII*, pp. 1–68. Springer, 2006.

Chris Bennett. *Heterogeneity in multi-agent systems*. PhD thesis, University of Bristol, 2024.

Shalabh Bhatnagar. An actor–critic algorithm with function approximation for discounted cost constrained markov decision processes. *Systems & Control Letters*, 59(12):760–766, 2010.

Shalabh Bhatnagar, Richard S Sutton, Mohammad Ghavamzadeh, and Mark Lee. Naturalgradient actor-critic algorithms. *Automatica*, 2007.

Vivek S Borkar and Vivek S Borkar. *Stochastic approximation: a dynamical systems viewpoint*, volume 9. Springer, 2008.

Lucian Busoniu, Robert Babuska, and Bart De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2):156–172, 2008.

Yash Chandak, Georgios Theocharous, James Kostas, Scott Jordan, and Philip Thomas. Learning action representations for reinforcement learning. In *International conference on machine learning*, pp. 941–950. PMLR, 2019.

Yu-Han Chang, Tracey Ho, and Leslie Kaelbling. All learning is local: Multi-agent learning in global reward games. *Advances in neural information processing systems*, 16, 2003.

Xuyang Chen and Lin Zhao. Finite-time analysis of single-timescale actor-critic. *Advances in Neural Information Processing Systems*, 36:7017–7049, 2023.

Filippos Christianos, Lukas Schäfer, and Stefano Albrecht. Shared experience actor-critic for multi-agent reinforcement learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 10707–10717. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/7967cc8e3ab559e68cc944c44b1cf3e8-Paper.pdf.

Filippos Christianos, Georgios Papoudakis, Muhammad A Rahman, and Stefano V Albrecht. Scaling multi-agent reinforcement learning with selective parameter sharing. In *International Conference on Machine Learning*, pp. 1989–1998. PMLR, 2021.

Tianshu Chu, Sandeep Chinchali, and Sachin Katti. Multi-agent reinforcement learning for networked system control. *arXiv preprint arXiv:2004.01339*, 2020.

Peter Corke, Ron Peterson, and Daniela Rus. Networked robots: Flying robot navigation using a sensor net. In *Robotics Research. The Eleventh International Symposium: With 303 Figures*, pp. 234–243. Springer, 2005.

Emiliano Dall'Anese, Hao Zhu, and Georgios B Giannakis. Distributed optimal power flow for smart microgrids. *IEEE Transactions on Smart Grid*, 4(3):1464–1475, 2013.

Thomas Degris, Martha White, and Richard S Sutton. Off-policy actor-critic. *arXiv preprint arXiv:1205.4839*, 2012.

Yali Du, Lei Han, Meng Fang, Ji Liu, Tianhong Dai, and Dacheng Tao. Liir: Learning individual intrinsic reward in multi-agent reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

Yali Du, Chengdong Ma, Yuchen Liu, Runji Lin, Hao Dong, Jun Wang, and Yaodong Yang. Scalable model-based policy optimization for decentralized networked systems. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9019–9026. IEEE, 2022.

Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, 101(suppl_1):5220–5227, 2004.

Jianqing Fan, Zheng Tracy Ke, and Zhaoyang Shi. Optimal demixing of nonparametric densities. *arXiv preprint arXiv:2603.27457*, 2026.

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

Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pp. 1587–1596. PMLR, 2018.

Bryan S Graham. Network data. In *Handbook of econometrics*, volume 7, pp. 111–218. Elsevier, 2020.

Harry H Harman. *Modern factor analysis*. University of Chicago press, 1976.

J Harold, G Kushner, and George Yin. Stochastic approximation and recursive algorithm and applications. *Application of Mathematics*, 35(10), 1997.

Zhiyuan He, Siwei Zhang, Zhiwen Zhou, Yuqing Yang, Yu Kang, Yuge Zhang, Luna K Qiu, Tin Yan Tsui, Jiahang Xu, and Chong Luo. Agent lightning v1. 0: Towards harnessed agentic rl. *arXiv preprint arXiv:2608.17528*, 2026.

Siyi Hu, Chuanlong Xie, Xiaodan Liang, and Xiaojun Chang. Policy diagnosis via measuring role diversity in cooperative multi-agent rl. In *International Conference on Machine Learning*, pp. 9041–9071. PMLR, 2022.

Tianyi Hu, Zhiqiang Pu, Yuan Wang, Tenghai Qiu, Min Chen, and Xin Yu. Heterogeneity in multi-agent reinforcement learning. *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2026.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Jiashun Jin, Zheng Tracy Ke, and Shengming Luo. Mixed membership estimation for social networks. *Journal of Econometrics*, 239(2):105369, 2024.

Woojun Kim and Youngchul Sung. Parameter sharing with network pruning for scalable multi-agent deep reinforcement learning. *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2023.

Paul Kline. *An easy guide to factor analysis*. Routledge, 2014.

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

Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. *International Conference on Learning Representations*, 2022.

Harold Joseph Kushner and Dean S Clark. *Stochastic approximation methods for constrained and unconstrained systems*, volume 26. Springer Science & Business Media, 2012.

Derrick Norman Lawley and Albert Ernest Maxwell. Factor analysis as a statistical method. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 12(3):209–229, 1962.

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

Dapeng Li, Na Lou, Bin Zhang, Zhiwei Xu, and Guoliang Fan. Adaptive parameter sharing for multi-agent reinforcement learning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6035–6039. IEEE, 2024.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Yiheng Lin, Guannan Qu, Longbo Huang, and Adam Wierman. Multi-agent reinforcement learning in stochastic networked systems. *Advances in neural information processing systems*, 34:7825–7837, 2021.

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

Pangpang Liu, Junwei Lu, and Will Wei Sun. Uncertainty quantification for large language model reward learning under heterogeneous human feedback. *arXiv preprint arXiv:2512.03208*, 2025.

Yilin Liu, Guiyang Luo, Quan Yuan, Jinglin Li, Lei Jin, Bo Chen, and Rui Pan. Gplight: Grouped multi-agent reinforcement learning for large-scale traffic signal control. In *IJCAI*, pp. 199–207, 2023.

Yuntao Liu, Yuan Li, Xinhai Xu, Yong Dou, and Donghong Liu. Heterogeneous skill learning for multi-agent tasks. *Advances in neural information processing systems*, 35:37011–37023, 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.

Chengdong Ma, Aming Li, Yali Du, Hao Dong, and Yaodong Yang. Efficient and scalable reinforcement learning for large-scale network control. *Nature Machine Intelligence*, 6(9):1006–1020, 2024a.

Hao Ma, Tianyi Hu, Zhiqiang Pu, Boyin Liu, Xiaolin Ai, Yanyan Liang, and Min Chen. Coevolving with the other you: Fine-tuning llm with sequential cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 37:15497–15525, 2024b.

Anuj Mahajan, Tabish Rashid, Mikayel Samvelyan, and Shimon Whiteson. Maven: Multi-agent variational exploration. *Advances in neural information processing systems*, 32, 2019.

Peter Marbach and John N Tsitsiklis. Simulation-based optimization of markov reward processes. *IEEE Transactions on Automatic Control*, 46(2):191–209, 2002.

Ahmed Masry, Jia Qing Tan, Shafiq Joty, Enamul Hoque, et al. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pp. 2263–2279, 2022.

Eric Mazumdar, Kishan Panaganti, and Laixi Shi. Tractable multi-agent reinforcement learning through behavioral economics. In *The Thirteenth International Conference on Learning Representations*, 2025.

Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Hyungho Na and Il-chul Moon. Lagma: Latent goal-guided multi-agent reinforcement learning. *International conference on machine learning*, 2024.

Hyungho Na, Kwanghyeon Lee, Sumin Lee, and Il-Chul Moon. Trajectory-class-aware multi-agent reinforcement learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Siddharth Nayak, Kenneth Choi, Wenqi Ding, Sydney Dolan, Karthik Gopalakrishnan, and Hamsa Balakrishnan. Scalable multi-agent reinforcement learning through intelligent information aggregation. In *International Conference on Machine Learning*, pp. 25817–25833. PMLR, 2023.

J Neveu. Discrete-parameter martingales.

Frans A Oliehoek, Matthijs TJ Spaan, and Nikos Vlassis. Optimal and approximate q-value functions for decentralized pomdps. *Journal of Artificial Intelligence Research*, 32:289–353, 2008.

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

Liviu Panait and Sean Luke. Cooperative multi-agent learning: The state of the art. *Autonomous agents and multi-agent systems*, 11(3):387–434, 2005.

Georgios Papoudakis, Filippos Christianos, Lukas Schäfer, and Stefano V. Albrecht. Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks, November 2021. URL http://arxiv.org/abs/2006.07869. arXiv:2006.07869 [cs.LG].

HL Prasad, LA Prashanth, and Shalabh Bhatnagar. Actor-critic algorithms for learning nash equilibria in n-player general-sum games. *arXiv preprint arXiv:1401.2086*, 2014.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Guannan Qu, Yiheng Lin, Adam Wierman, and Na Li. Scalable multi-agent reinforcement learning for networked systems with average reward. *Advances in Neural Information Processing Systems*, 33:2074–2086, 2020.

Michael Rabbat and Robert Nowak. Distributed optimization in sensor networks. In *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pp. 20–27, 2004.

Roxana Rădulescu, Patrick Mannion, Diederik M Roijers, and Ann Nowé. Multi-objective multi-agent decision making: a utility-based analysis and survey. *Autonomous Agents and Multi-Agent Systems*, 34(1):10, 2020.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178):1–51, 2020.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015a.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.

Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10): 1095–1100, 1953.

Zhaoyang Shi. Population-level generative modeling for ranking data. *arXiv preprint arXiv:2608.08422*, 2026.

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

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Guy Tennenholtz and Shie Mannor. The natural language of actions. In *International Conference on Machine Learning*, pp. 6196–6205. PMLR, 2019.

Justin K Terry, Nathaniel Grammel, Sanghyun Son, Benjamin Black, and Aakriti Agrawal. Revisiting parameter sharing in multi-agent deep reinforcement learning. *arXiv preprint arXiv:2005.13625*, 2020.

Gerald Tesauro and Jeffrey O Kephart. Pricing in agent economies using multi-agent q-learning. *Autonomous agents and multi-agent systems*, 5(3):289–304, 2002.

John Tsitsiklis and Benjamin Van Roy. Analysis of temporal-diffference learning with function approximation. *Advances in neural information processing systems*, 9, 1996.

John N Tsitsiklis and Benjamin Van Roy. Average cost temporal-difference learning. *Automatica*, 35 (11):1799–1808, 1999.

Karl Tuyls and Gerhard Weiss. Multiagent learning: Basics, challenges, and prospects. *Ai Magazine*, 33(3):41–41, 2012.

Paulina Varshavskaya, Leslie Pack Kaelbling, and Daniela Rus. Efficient distributed reinforcement learning through agreement. In *Distributed Autonomous Robotic Systems 8*, pp. 367–378. Springer, 2009.

Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. Qplex: Duplex dueling multi-agent q-learning. *International Conference on Learning Representations*, 2021a.

Tonghan Wang, Heng Dong, Victor Lesser, and Chongjie Zhang. Roma: Multi-agent reinforcement learning with emergent roles. *International conference on machine learning*, 2020.

Tonghan Wang, Tarun Gupta, Anuj Mahajan, Bei Peng, Shimon Whiteson, and Chongjie Zhang. Rode: Learning roles to decompose multi-agent tasks. *International Conference on Learning Representations*, 2021b.

David H Wolpert and Kagan Tumer. Optimal payoff functions for members of collectives. *Advances in Complex Systems*, 4(02n03):265–279, 2001.

Yue Frank Wu, Weitong Zhang, Pan Xu, and Quanquan Gu. A finite-time analysis of two time-scale actor-critic methods. *Advances in Neural Information Processing Systems*, 33:17617–17628, 2020.

Mingyu Yang, Jian Zhao, Xunhan Hu, Wengang Zhou, Jiangcheng Zhu, and Houqiang Li. Ldsa: Learning dynamic subtask assignment in cooperative multi-agent reinforcement learning. *Advances in neural information processing systems*, 35:1698–1710, 2022.

Yuxuan Yi, Ge Li, Yaowei Wang, and Zongqing Lu. Learning to share in networked multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 35:15119–15131, 2022.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in neural information processing systems*, 35:24611–24624, 2022.

Beining Zhang, Aditya Kapoor, and Mingfei Sun. Low-rank agent-specific adaptation (lorasa) for multi-agent policy learning. *The Seventh International Conference on Distributed Artificial Intelligence (DAI)*, 2025.

Kaiqing Zhang, Liqun Lu, Chao Lei, Hao Zhu, and Yanfeng Ouyang. Dynamic operations and pricing of electric unmanned aerial vehicle systems and power networks. *Transportation Research Part C: Emerging Technologies*, 92:472–485, 2018a.

Kaiqing Zhang, Wei Shi, Hao Zhu, Emiliano Dall'Anese, and Tamer Başar. Dynamic power distribution system management with a locally connected communication network. *IEEE Journal of Selected Topics in Signal Processing*, 12(4):673–687, 2018b.

Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International conference on machine learning*, pp. 5872–5881. PMLR, 2018c.

Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pp. 321–384, 2021.

Yifan Zhong, Jakub Grudzien Kuba, Xidong Feng, Siyi Hu, Jiaming Ji, and Yaodong Yang. Heterogeneous-agent reinforcement learning. *Journal of Machine Learning Research*, 25(32): 1–67, 2024.

Appendix

Table of Contents

A Related Work **17**

B Theoretical Results **18**
B.1 Proof of Proposition 2.1 18
B.2 Proof of Theoretical Convergence 18
B.3 Auxiliary Results 19
B.4 Proof of Theorem 3.1 20
B.5 Proof of Theorem 3.2 22

C Algorithms **23**
C.1 Pseudocode of Average-Reward Actor-Critic 23
C.2 Generic On-policy and Off-policy Policy Gradient Algorithms 23

D Experiment Details **25**
D.1 Multi-Agent Particle Environment (MPE) 25
D.2 Level-Based Foraging (LBF) and Multi-Robot Warehouse (RWARE) 26
D.3 Agentic RL 27

A Related Work

MARL with Heterogeneous Agents. Most MARL algorithms, including value factorization (Sunehag et al., 2017; Rashid et al., 2020; Wang et al., 2021a) and centralized critics (Lowe et al., 2017; Foerster et al., 2018; Yu et al., 2022), assume a shared global reward function in a cooperative setting and inherently imply homogeneous agent behaviors. To enable diverse and specialized behaviors, parameter sharing with diversity encouraging (Li et al., 2021) and specialized fine-tuning (Terry et al., 2020; Christianos et al., 2021; Li et al., 2024; Kim & Sung, 2023; Zhang et al., 2025; Hu et al., 2026) are proposed, but they often possess limited applicability (Hu et al., 2022; Zhong et al., 2024). Christianos et al. (2021) recognizes the behavior heterogeneity through hard clustering in a latent trajectory space to apply the selective parameter sharing. By contrast, our approach focuses on discovering latent and possibly overlapping objectives specifically in the reward space, capturing heterogeneity at the level of incentives beyond behavior similarity. Kuba et al. (2022); Zhong et al. (2024) performed a sequential update scheme based on the advantage decomposition to account for the heterogeneous agents in the regime with a shared reward function, but they did not explicitly define or characterize heterogeneity of policies among agents and only focused on the cooperative setting.

Latent Representation in MARL. Na & Moon (2024); Na et al. (2025) propose to generate a goal-reaching trajectory in latent state space to provide a latent goal-guided incentive or identify the class of trajectories in latent trajectory space to perform task-aware execution. Task division methods, such as role-based task factorization methods (Wang et al., 2020; 2021b; Tennenholtz & Mannor, 2019; Chandak et al., 2019), subtask assignment (Yang et al., 2022), and skill learning (Liu et al., 2022), factor the agent's identity or action space to define different subtasks, skills, or roles in latent spaces. In contrast, our latent objective discovery operates the factorization at complementary levels of abstraction: the former structures how agents act, while the latter structures why agents act.

Networked MARL. Networked MARL (Varshavskaya et al., 2009; Zhang et al., 2018c; 2021; Lin et al., 2021) is typically seen as a theoretical framework that connects heterogeneous agents with individual reward functions through a graph and executes decentralized control by leveraging the decentralized information. Although this paradigm naturally contributes to local privacy and communication efficiency in network control (Qu et al., 2020; Chu et al., 2020; Yi et al., 2022; Du et al., 2022; Nayak et al., 2023; Ma et al., 2024a), the network topology above is often explicitly pre-specified, heuristically designed, or externally observed, restricting coordination to hard-coded local structure. By contrast, our latent objective discovery induces a latent, non-local, and statistically principled network topology to learn objective abstractions guided by agents' diverse incentives.

B THEORETICAL RESULTS

B.1 PROOF OF PROPOSITION 2.1

Proof. For the agent i, the latent reward decomposition in Eq. 3 indicates that

$$R_t^i = \gamma_i^\top \boldsymbol{F}_t + \epsilon_t^i.$$

Recall the definition of the action-value function $Q_i^{\boldsymbol{\pi}}(s, \boldsymbol{a})$ for each agent i:

$$Q_i^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t R_t^i \mid s_0 = s, a_0 = \boldsymbol{a} \right].$$

Next, we plug the equation of R_t^i into the definition of $Q_i^{\boldsymbol{\pi}}$ above, which results in

$$\begin{aligned} Q_i^{\boldsymbol{\pi}}(s, \boldsymbol{a}) &\overset{(a)}{=} \gamma_i^\top \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \boldsymbol{F}_t \mid s_0 = s, a_0 = \boldsymbol{a} \right] + \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \epsilon_t^i \mid s_0 = s, a_0 = \boldsymbol{a} \right] \\ &\overset{(b)}{=} \gamma_i^\top \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \boldsymbol{F}_t \mid s_0 = s, a_0 = \boldsymbol{a} \right] \\ &:= \gamma_i^\top \boldsymbol{Q}_{\boldsymbol{F}}^{\boldsymbol{\pi}}(s, \boldsymbol{a}), \end{aligned}$$

where (a) is true due to the fact that Γ is time-independent and (b) holds because ϵ_t^i is the white noise and $\mathbb{E}_\pi \left[\epsilon_t^i \mid s_0 = s, a_0 = \boldsymbol{a} \right] = 0$. By the definition of $\boldsymbol{Q}_{\boldsymbol{F}}$, the simplification above holds, which further implies that

$$\boldsymbol{Q}^\pi(s, \boldsymbol{a}) = \Gamma \boldsymbol{Q}_{\boldsymbol{F}}^{\boldsymbol{\pi}}(s, \boldsymbol{a}). \tag{8}$$

□

B.2 PROOF OF THEORETICAL CONVERGENCE

In Theorems 3.1 and 3.2, we provided the theoretical guarantee for our proposed generic actor-critic algorithm (See Algorithm 2 in Appendix C.1). In this section, we present more details on the theoretical analysis, including the full mathematical definitions of the equations appeared in these theorems.

To this end, we introduce some additional notations for our theoretical results. With a slight abuse of the notation, let $P^\theta(s', a'|s, a) = P(s'|s, a)\pi_\theta(s', a')$. Here, it should be distinguished within context easily from the transition probability matrix $P^\theta(s'|s)$ of the Markov chain $\{s_t\}_{t\geq 0}$ under the policy π_θ. Let $D_\theta^{s,a} = \text{diag}(d_\theta(s)\pi_\theta(s, a), s \in \mathcal{S}, a \in \mathcal{A})$. For each factor k, let $R^{(k)} = [R^{(k)}(s, a), s \in \mathcal{S}, a \in \mathcal{A}]' \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ and $J^{(k)}(\theta) = \sum_{s\in\mathcal{S}, a\in\mathcal{A}} d_\theta(s)\pi_\theta(s, a)R^{(k)}(s, a)$, for $1 \leq k \leq K$. They represent the k-th latent factor of the objectives reward R^i and $J(\theta)$ induced by the factor model equation 3. We also define an operator $T_\theta^{Q,(k)} : \mathbb{R}^{|\mathcal{S}||\mathcal{A}|} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ for any action-value vector $Q \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ as

$$T_\theta^{Q,(k)}(Q) = R^{(k)} - J^{(k)}(\theta) \cdot \mathbf{1} + P^\theta Q, \ 1 \leq k \leq K.$$

Now, we first provide the details of Theorem 3.1.

Theorem B.1. *Under Assumption 3.1 and 3.3 to 3.5, for any policy π_θ given, the factor parameter $\omega_t^{(k)}$ converges: $\lim_t \omega_t^{(k)} = \omega_\theta$ almost surely and the limit ω_θ is the unique solution to*

$$\Phi' D_\theta^{s,a}(T_\theta^{Q,(k)}(\Phi\omega_\theta^{(k)}) - \Phi\omega_\theta^{(k)}) = 0, \ 1 \leq k \leq K.$$

Next we provide the details of Theorem 3.2. During the actor step, in order to show the convergence of the update in θ, we introduce:

$$\begin{aligned} A_{t,\theta}^i &= \phi_t' \omega_\theta^i - \sum_{a^i \in \mathcal{A}} \pi_{\theta^i}^i(s_t, a^i)\phi(s_t, a^i, a_t^{-i})'\omega_\theta^i, \\ \psi_{t,\theta}^i &= \nabla_{\theta^i} \log \pi_{\theta^i}^i(s_t, a_t^i), \end{aligned} \tag{9}$$

where $\phi_t = \phi(s_t, a_t)$ and ω_θ^i is defined as follows. Since the actor step is updating θ for each agent's policy $\pi_{\theta^i}^i$ and we already defined the limiting factor parameter $\omega_\theta^{(k)}$ in Theorem B.1, the limiting agent parameter ω_θ^i is thus defined by enforcing the factor model structure $\omega_\theta^i = \sum_{k=1}^K \gamma_i(k)\omega_\theta^{(k)}$. Note here that due to the linear assumption of the Q-function, it suffices to only linearly aggregate its parameter ω (otherwise, we need to strictly aggregate Q-functions instead). Furthermore, for each $i \in \mathcal{N}$, define a vector

$$\mathcal{L}^i(g(\theta)) := \lim_{\eta \to 0^+} \frac{\mathcal{L}_0^i(\theta^i + \eta g(\theta)) - \theta^i}{\eta},$$

for any $\theta \in \Theta$ and $g : \Theta \to \mathbb{R}^{\sum_{i\in\mathcal{N}} m_i}$ a continuous function. If the above limit is not unique, we define $\mathcal{L}^i(g(\theta))$ to be the set of all possible limit points of the right-hand side. The details of Theorem 3.2 are included in the following theorem.

Theorem B.2. *Under Assumption 3.1 to Assumption 3.5, the agent policy parameter θ_t^i converges almost surely to a point in the set of the asymptotically stable equilibria of*

$$\dot{\theta}^i = \mathcal{L}^i(\mathbb{E}_{s_t\sim d_\theta, a_t\sim\pi_\theta}(A_{t,\theta}^i \psi_{t,\theta}^i)),\ i \in \mathcal{N}.$$

B.3 Auxiliary Results

B.3.1 Results on Stochastic Approximation

In this section, we introduce a basic result of stochastic approximation shown in Borkar & Borkar (2008). More general conclusions can also be found in Benaïm (2006); Harold et al. (1997).

Consider the following n-dimensional ($n \in \mathbb{N}$) stochastic approximation iteration:

$$x_{t+1} = x_t + \zeta_t(h(x_t, Y_t) + M_{t+1} + \eta_{t+1}),\ t \geq 0,$$

where $\zeta_t > 0$ and $\{Y_t\}_{t\geq 0}$ is a Markov chain on a finite set A.

Theorem B.3. *Suppose the following assumptions hold:*

(1). $h : \mathbb{R}^n \times A \to \mathbb{R}^n$ is Lipschitz continuous in its first argument;

(2). $\{Y_t\}_{t\geq 0}$ is an irreducible Markov chain with stationary distribution π;

(3). The stepsize sequence ζ_t is such that $\sum_t \zeta_t = \infty$ and $\sum_t \zeta_t^2 < \infty$;

(4). $\{M_t\}$ is a martingale difference sequence, i.e., $\mathbb{E}(M_{t+1}|x_\tau, M_\tau, Y_\tau, \tau \leq t) = 0$, and it holds that for some $C > 0$ and $t \geq 0$,

$$\mathbb{E}\left(\|M_{t+1}\|^2 | x_\tau, M_\tau, Y_\tau, \tau \leq t\right) \leq C(1 + \|x_t\|^2);$$

(5). The sequence $\{\eta_t\}$ is a bounded random sequence with $\lim_t \eta_t = 0$ almost surely.

Then the asymptotic behavior of the above iteration is associated with the behavior of the solution to the ODE:

$$\dot{x} = \bar{h}(x) := \sum_i \pi(i) h(x, i).$$

Moreover, suppose the above ODE has a unique globally asymptotically stable equilibrium x^*. We have:

(a). If $\sup_t \|x_t\| < \infty$ almost surely, then $x_t \to x^*$.

(b). Suppose $\lim_{c\to\infty} \frac{\bar{h}(cx)}{c} = h_\infty(x)$ exists uniformly on compact sets for some $h_\infty \in C(\mathbb{R}^n)$. If the ODE $\dot{y} = h_\infty(y)$ has origin as the unique globally asymptotically stable equilibrium, then

$$\sup_t \|x_t\| < \infty,\ \textit{almost surely}.$$

B.3.2 KUSHNER-CLARK LEMMA

The Kushner-Clark Lemma is a well-known result shown in Kushner & Clark (2012).

Let $\mathcal{L}_0$ be an operator projecting a vector onto a compact set $\mathcal{X} \subset \mathbb{R}^N$. Define an operator $\mathcal{L}$ such that for any $x \in \mathcal{X}$ and $h : \mathcal{X} \to \mathbb{R}^N$ continuous

$$\mathcal{L}(h(x)) := \lim_{\eta \to 0^+} \frac{\mathcal{L}_0(x + \eta h(x)) - x}{\eta}.$$

Now, consider the following update in N dimensions:

$$x_{t+1} = \mathcal{L}(x_t + \zeta_t(h(x_t) + M_t + \eta_t)).$$

Its asymptotic behavior is associated with the following ODE:

$$\dot{x} = \mathcal{L}(h(x)).$$

Theorem B.4. *Suppose the following assumptions hold:*

(1). $h(\cdot)$ is a continuous $\mathbb{R}^N$-valued function;

(2). The sequence $\{\eta_t\}$ is a bounded random sequence with $\lim_t \eta_t = 0$ almost surely;

(3). The stepsize sequence ζ_t is such that $\sum_t \zeta_t = \infty$ and $\lim_t \zeta_t = 0$;

(4). The sequence $\{M_t\}$ satisfies: for any $\epsilon > 0$,

$$\lim_t \mathbb{P}\left(\sup_{n \geq t} \left\| \sum_{t=\tau}^{n} \zeta_\tau M_\tau \right\| \geq \epsilon \right) = 0.$$

Then, if the above ODE has a compact set $\mathcal{K}^*$ its set of asymptotically stable equilibria, x_t converges almost surely to $\mathcal{K}^*$ as $t \to \infty$.

B.4 PROOF OF THEOREM 3.1

Recall Algorithm 2 in Appendix C. For notational convience, in this section, we write $r_t^{(k)} := f_t^k$ as the k-th latent reward for $1 \leq k \leq K$. Let $e_t^{(k)} = f_t^k - \mu_{t-1}^k + Q_{\omega_k^{t-1}}(s_{t+1}, \tilde{\boldsymbol{a}}_{t+1}) - Q_{\omega_k^{t-1}}(s_t, \boldsymbol{a}_t)$, and $y_{t+1} = [e_t^{(1)}\phi_t', \ldots, e_t^{(K)}\phi_t']' \in \mathbb{R}^{Km_\Omega}$. Here, all quantities with (k) superscript correspond to their k-th latent factors. Moreover, since we have assumed $\eta_{\theta,t} = o(\eta_{\omega,t})$, in the spirit of the classical two-time-scale SA analysis in Chapter 6.1 of (Borkar & Borkar, 2008), we can let the policy parameter be fixed as $\theta_t \equiv \theta$ when analyzing the convergence of the critic step, which implies we can show the convergence of ω_t to some ω_θ depending on θ. For notational simplicity, we eliminate the notations associated with θ unless otherwise noted.

The update in Algorithm 2 in Appendix C is then expressed as

$$\mu_{t+1}^{(k)} = (1 - \eta_{\omega,t})\mu_t^{(k)} + \eta_{\omega,t} r_{t+1}^{(k)},$$
$$\omega_{t+1}^{(k)} = \omega_t^{(k)} + \eta_{\omega,t} y_{t+1}^{(k)}.$$

Here, we will apply Theorem B.3 as a technical result. To this end, we first verify all the assumptions in Theorem B.3. We write the above update as

$$\mu_{t+1}^{(k)} = \mu_t^{(k)} + \eta_{\mu,t}\mathbb{E}\left(r_{t+1}^{(k)} - \mu_t^{(k)} | \mathcal{F}_{t,1}\right) + \eta_{\mu,t} A_{\mu,t+1}^{(k)},$$
$$\omega_{t+1}^{(k)} = \omega_t^{(k)} + \eta_{\omega,t}\mathbb{E}\left(e_t^{(k)}\phi_t | \mathcal{F}_{t,1}\right) + \eta_{\omega,t} A_{\omega,t+1}^{(k)}, \tag{10}$$

where $A_{\mu,t+1}^{(k)} = r_{t+1}^{(k)} - \mathbb{E}\left(r_{t+1}^{(k)} | \mathcal{F}_{t,1}\right)$ and $A_{\omega,t+1} = y_{t+1}^{(k)} - \mathbb{E}\left(e_t^{(k)}\phi_t | \mathcal{F}_{t,1}\right)$. Here, $\mathbb{E}\left(e_t^{(k)}\phi_t | \mathcal{F}_{t,1}\right)$ is Lipschitz continuous in $\omega_t^{(k)}$ and $\mu_t^{(k)}$. Moreover, by definition, $\{A_{\mu,t}^{(k)}\}$ is a martingale difference sequence satisfying: there exists a constant $C > 0$ such that

$$\mathbb{E}\left(\|A_{\mu,t+1}^{(k)}\|^2 | \mathcal{F}_{t,1}\right) \leq C(1 + \|\omega_t^{(k)}\|^2 + \|\mu_t^{(k)}\|^2),$$

given $r_{t+1}^{(k)}$ is uniformly bounded by Assumption 3.1. Furthermore, we will show $\{A_{\omega,t}^{(k)}\}$ is also a martingale difference sequence. Note that by definition,

$$\mathbb{E}\left(y_{t+1}^{(k)}|\mathcal{F}_{t,1}\right) = \mathbb{E}\left(e_t^{(k)}\phi_t|\mathcal{F}_{t,1}\right),$$

and

$$\mathbb{E}\left(\|A_{\omega,t+1}^{(k)}\|^2|\mathcal{F}_{t,1}\right) \leq 2\left(\mathbb{E}\left(\|y_{t+1}^{(k)}\|^2|\mathcal{F}_{t,1}\right) + \|\mathbb{E}(e_t^{(k)}\phi_t|\mathcal{F}_{t,1})\|^2\right). \tag{11}$$

Thus, the first term can be bounded over the set $\{\sup_t \|z_t^{(k)}\| \leq M\}$ for any $M > 0$. Note that Here, by definition and Jensen's inequality, we have:

$$\begin{aligned}
\mathbb{E}(\|y_{t+1}^{(k)}\|^2|\mathcal{F}_{t,1}) &= \mathbb{E}\left(\|\delta_t^{(k)}\phi_t\|^2|\mathcal{F}_{t,1}\right) \\
&= \mathbb{E}\left(\|(r_{t+1}^{(k)} - \mu_t^{(k)} + \phi_{t+1}'\omega_t^{(k)} - \phi_t'\omega_t^{(k)})\phi_t\|^2|\mathcal{F}_{t,1}\right) \\
&\leq 3\mathbb{E}\left(\left(\|r_{t+1}^{(k)}\phi_t\|^2 + \|\mu_t^{(k)}\phi_t\|^2 + \|\phi_t(\phi_{t+1}' - \phi_t')\|^2(\omega_t^{(k)})^2\right)|\mathcal{F}_{t,1}\right).
\end{aligned}$$

By our assumption, $\mathbb{E}\left(\|\phi_t(\phi_{t+1}' - \phi_t')\|^2|\mathcal{F}_{t,1}\right)$ and $\mathbb{E}\left(\|\phi_t\|^2|\mathcal{F}_{t,1}\right)$ are both uniformly bounded for any $s_t \in \mathcal{S}, a_t \in \mathcal{A}$. Also, due to Assumption 3.1, $\mathbb{E}\left(|r_{t+1}^{(k)}|^2|\mathcal{F}_{t,1}\right) = \mathbb{E}\left(|r_{t+1}^{(k)}|^2|s_t, a_t\right)$ is uniformly bounded. Then, over the set $\{\sup_t \|z_t^{(k)}\| \leq M\}$, it holds that: there exists a constant $C > 0$,

$$\mathbb{E}\left(\|y_{t+1}^{(k)}\|^2|\mathcal{F}_{t,1}\right) \cdot \mathbb{1}(\sup_t \|z_t^{(k)}\| \leq M) \leq C.$$

Now, the second term in the right-hand side of equation 11 can be directly bounded by

$$\|\mathbb{E}(e_t^{(k)}\phi_t|\mathcal{F}_{t,1})\|^2 \leq C(1 + \|\omega_t^{(k)}\|^2 + \|\mu_t^{(k)}\|^2),$$

for some constant $C > 0$ since we assume $r_{t+1}^{(k)}$ and ϕ_t are bounded by Assumption 3.1 and Assumption 3.5. Then, we have: for any $M > 0$,

$$\mathbb{E}\left(\|A_{\omega,t+1}^{(k)}\|^2|\mathcal{F}_{t,1}\right) \leq C(1 + \|\omega_t^{(k)}\|^2 + \|\mu_t^{(k)}\|^2),$$

over the set $\{\sup_t \|z_t^{(k)}\| \leq M\}$. This verifies assumption (4) in Theorem B.3.

Now, according to Theorem B.3, we consider the ODE associated with the asymptotic behavior of equation 10:

$$\dot{z}^{(k)} = \begin{pmatrix}\dot{\mu}^{(k)} \\ \dot{\omega}^{(k)}\end{pmatrix} = \begin{pmatrix}-1 & 0 \\ -\Phi' D_\theta^{s,a}\mathbf{1} & \Phi' D_\theta^{s,a}(P^\theta - I)\Phi\end{pmatrix}\begin{pmatrix}\mu^{(k)} \\ \omega^{(k)}\end{pmatrix} + \begin{pmatrix}J^{(k)}(\theta) \\ \Phi' D_\theta^{s,a} R^{(k)}\end{pmatrix},$$

where recall that $D_\theta^{s,a} = \text{diag}(d_\theta(s)\pi_\theta(s,a), s \in \mathcal{S}, a \in \mathcal{A})$ and $J^{(k)}(\theta)$ and $R^{(k)}$ are defined in Theorem 3.1. In order to apply the conclusions of Theorem B.3, we then examine the equilibrium to the above ODE. Denote by $h^{(k)}(z^{(k)})$ the right-hand side of the above ODE. It is straightforward to verify $h^{(k)}(\cdot)$ is Lipschitz, satisfying assumption (1) there. Moreover, by the Perron-Frobenius theorem, the stochastic matrix P^θ has a simple eigenvalue of 1 while the rest of the eigenvalues have real parts less than 1. Thus, $P^\theta - I$ has a simple eigenvalue of 0 with all other eigenvalues having negative real parts, so does the matrix $\Phi' D_\theta^{s,a}(P^\theta - I)\Phi$, given that we assume Φ is full column rank in Assumption 3.5. The simple eigenvalue of 0 has eigenvector ν such that $\Phi\nu = \varrho\mathbf{1}$ for some $\varrho > 0$ by checking that $\varrho\mathbf{1}$ lies in the eigenspace of $D_\theta^{s,a}(P^\theta - I$ associated with 0. Consequently, the above ODE is globally asymptotically stable and has its equilibrium satisfying:

$$\begin{cases}\mu^{(k)} = J^{(k)}(\theta) \\ \Phi' D_\theta^{s,a}(R^{(k)} - \mu^{(k)}\mathbf{1} + P^\theta\Phi\omega^{(k)} - \Phi\omega^{(k)}) = 0.\end{cases}$$

Note here that the solution for $\omega^{(k)}$ has the form $\omega_\theta^{(k)} + \varrho\nu$ with any $\varrho > 0$ and $\nu \in \mathbb{R}^{m_\Omega}$ such that $\Phi\nu = \mathbf{1}$. Note that by Assumption 3.5, $\Phi\nu \neq \mathbf{1}$. This implies $\omega_\theta^{(k)}$ is unique such that $\Phi' D_\theta^{s,a}(T_\theta^{Q,(k)}(\Phi\omega_\theta^{(k)}) - \Phi\omega_\theta^{(k)}) = 0$ with $T_\theta^{Q,(k)}$ given in Theorem 3.1.

Now, by (Zhang et al., 2018c, Lemma B.1 and B.2), it implies that $\{\mu_t^{(k)}\}$ and $\{\omega_t^{(k)}\}$ are both bounded almost surely, showing that $\{z_t^{(k)}\}$ is bounded almost surely. Therefore, all assumptions in Theorem B.3 are satisfied. Consequently, we obtain $\lim_t \mu_t^{(k)} = J^{(k)}(\theta)$ and $\lim_t \omega_t^{(k)} = \omega_\theta^{(k)}$ over the set $\{\sup_t \|z_t^{(k)}\| \leq M\}$ for any $M > 0$, which holds almost surely. Then, it completes the proof for k. Since the above arguments hold for any $1 \leq k \leq K$, we obtain the proof for Theorem 3.1.

B.5 Proof of Theorem 3.2

Let $\mathcal{F}_{t,2} = \sigma(\{\theta_\tau\}_{\tau \leq t})$ be the σ-field generated by $\{\theta_\tau\}_{\tau \leq t}$. Now, define:

$$B_{t+1,1}^i := A_t^i \psi_t^i - \mathbb{E}_{s_t \sim d_{\theta_t}, a_t \sim \pi_{\theta_t}}(A_t^i \psi_t^i | \mathcal{F}_{t,2}),$$
$$B_{t+1,2}^i := \mathbb{E}_{s_t \sim d_{\theta_t}, a_t \sim \pi_{\theta_t}}((A_t^i - A_{t,\theta_t}^i)\psi_t^i) | \mathcal{F}_{t,2}),$$

where A_{t,θ_t}^i is defined in equation 9 with $\theta = \theta_t$. Recall that the update in Algorithm 2 can be written with a local projection $\mathcal{L}^i$:

$$\theta_{t+1}^i = \mathcal{L}^i(\theta_t^i + \eta_{\theta,t}\mathbb{E}_{s_t \sim d_{\theta_t}, a_t \sim \pi_{\theta_t}}(A_{t,\theta_t}^i \psi_t^i | \mathcal{F}_{t,2}) + \eta_{\theta,t} B_{t+1,1}^i + \eta_{\theta,t} B_{t+1,2}^i),$$

where $B_{t+1,2}^i = o(1)$ since $\eta_{\theta,t} = o(\eta_{\omega,t})$ by Assumption 3.3, then the critic convergence in Theorem 3.1 is faster thus implying $A_t^i \to A_{t,\theta}^i$. Furthermore, let $M_t^i = \sum_{\tau=0}^t \eta_{\theta,\tau} B_{t+1,1}^i$. It is a martingale sequence and since $\{\omega_t^{(k)}\}$, $\{\gamma_{it}(k)\}$, $\{\psi_t^i\}$ and $\{\phi_t\}$ are all bounded, by Assumption 3.3, it holds that

$$\sum_t \mathbb{E}\left(\|M_{t+1}^i - M_t^i\|^2 | \mathcal{F}_{t,2}\right) = \sum_{t \geq 1} \|\eta_{\theta,t} B_{t+1,1}^i\|^2 < \infty, \text{ almost surely.}$$

By the martingale convergence theorem (Neveu, Proposition VII-2-3(c)), $\{M_t^i\}$ converges almost surely and it implies that for any $\epsilon > 0$,

$$\lim_t \mathbb{P}\left(\sup_{n \geq t} \left\| \sum_{\tau=t}^n \eta_{\theta,t} B_{\tau,1}^i \right\| \geq \epsilon\right) = 0$$

Now, let

$$g^i(\theta_t) = \mathbb{E}_{s_t \sim d_{\theta_t}, a_t \sim \pi_{\theta_t}}(A_{t,\theta_t}^i \psi_t^i | \mathcal{F}_{t,2}) = \sum_{s_t \in \mathcal{S}, a_t \in \mathcal{A}} d_{\theta_t}(s_t)\pi_{\theta_t}(s_t, a_t) A_{t,\theta_t}^i \psi_{t,\theta_t}^i.$$

In order to apply Theorem B.4 to show the convergence of θ_t^i, we need to verify that $g^i(\theta_t)$ is continuous in θ_t^i. First, ψ_t^i is continuous in θ_t^i. Next, note that $d_{\theta_t}(s_t)\pi_{\theta_t}(s_t, a_t)$ is the stationary distribution and thus is the solution to the equation $d_{\theta_t}(s)\pi_{\theta_t}(s, a) = \sum_{s' \in \mathcal{S}, a' \in \mathcal{A}} P^{\theta_t}(s', a' | s, a) d_{\theta_t}(s')\pi_{\theta_t}(s', a')$ and $\sum_{s \in \mathcal{S}, a \in \mathcal{A}} d_{\theta_t}(s)\pi_{\theta_t}(s, a) = 1$, where $P^{\theta_t}(s', a' | s, a) = P(s' | s, a)\pi_{\theta_t}(s', a')$. The unique solution to this set of linear equations can be verified to be continuous in θ_t with $\pi_{\theta_t}(s, a) > 0$. Moreover, A_{t,θ_t}^i is continuous in θ_t^i too as $\omega_{\theta_t} = \sum_{k=1}^K \gamma_{it}(k)\omega_{\theta_t}^{(k)}$, where $\omega_{\theta_t}^{(k)}$ is also continuous due to the fact that it is the unique solution to the linear equation $\Phi' D_\theta^{s,a}(T_\theta^{Q,(k)}(\Phi\omega_\theta^{(k)}) - \Phi\omega_\theta^{(k)}) = 0$, which can also be verified to be continuous in θ_t. Therefore, the convergence of θ_t^i is a direct consequence of applying Theorem B.4 for each $i \in \mathcal{N}$.

C ALGORITHMS

- Appendix C.1: we include a generic on-policy average-reward actor–critic in Algorithm 2. Off-policy variants (e.g., TD3-style) can be obtained by replacing the target action sampling with a target policy and replay buffer.
- Appendix C.2: we include a generic on-policy and off-policy policy gradient algorithm framework in Algorithm 3 and Algorithm 4, respectively.

C.1 PSEUDOCODE OF AVERAGE-REWARD ACTOR-CRITIC

Algorithm 2: Actor-Critic Algorithm with Latent Objective Discovery in the Average-Reward Setting

Initialize: $\{Q_{\omega_k^0}\}_{k=1}^K$, $\{\pi_{\theta_i^0}\}_{i\in\mathcal{N}}$, the historical average reward $\{\mu_0^k\}_{k=1}^K$, Γ_0, $\boldsymbol{f}_0$, the step sizes $\eta_{\omega,t}$ and $\eta_{\theta,t}$.
for $t = 1$ to max-episode-length **do**
  for $i \in \mathcal{N}$ **do**
    Observe s_t, take action a_t^i, receive r_t^i
  end for
  Form $\boldsymbol{a}_t$ and $\boldsymbol{r}_t$, observe s_{t+1}.
  # Latent Reward Discovery.
  Online estimation of $\boldsymbol{f}_t$ and Γ_t from $\boldsymbol{r}_t$ given Γ_{t-1} and $\boldsymbol{f}_{t-1}$ via Eq. 6.
  # Critic Update.
  for $i \in \mathcal{N}$ **do**
    form $\tilde{\boldsymbol{a}}_{t+1}$ by $\tilde{a}_{t+1}^i \sim \pi_{\theta_i^{t-1}}(\cdot|s_{t+1})$
  end for
  for $k = 1$ to K **do**
    $\delta_t^k \leftarrow f_t^k - \mu_{t-1}^k + Q_{\omega_k^{t-1}}(s_{t+1}, \tilde{\boldsymbol{a}}_{t+1}) - Q_{\omega_k^{t-1}}(s_t, \boldsymbol{a}_t)$ # Average-Reward Temporal Difference
    $\omega_k^t \leftarrow \omega_k^{t-1} + \eta_{\omega,t}\delta_t^k \nabla_{\omega_k} Q_{\omega_k^{t-1}}(s_t, \boldsymbol{a}_t)$
    $\mu_t^k \leftarrow (1-\eta_{\omega,t})\mu_{t-1}^k + \eta_{\omega,t} f_t^k$
  end for
  # Actor Update.
  for $i \in \mathcal{N}$ **do**
    $Q_i^t \leftarrow \sum_{k=1}^K \gamma_i^t(k) Q_{\omega_k^t}(s_t, \boldsymbol{a}_t)$ # Mixed Centralization.
    $A_i^t \leftarrow Q_i^t - \sum_{\tilde{a}_t^i} \pi_{\theta_i^{t-1}}(\tilde{a}_t^i|s_t) Q_i^t(s_t, (\tilde{a}_t^i, a_t^{-i}))$
    $\theta_i^t \leftarrow \theta_i^{t-1} + \eta_{\theta,t} A_i^t \nabla_{\theta_i^{t-1}} \log \pi_{\theta_i^{t-1}}(a_t^i|s_t)$
  end for
end for

C.2 GENERIC ON-POLICY AND OFF-POLICY POLICY GRADIENT ALGORITHMS

On-Policy Multi-agent Policy Gradient. The algorithm description is provided in Algorithm 3.

Off-Policy Multi-agent Policy Gradient. We define μ_{θ_i} as the deterministic policy. The latent critic loss with the TD target under the latent objective discovery are written as

$$\mathcal{L}_Q(\omega) = \frac{1}{B}\sum_{t\in\mathcal{B}}\sum_{k=1}^{K}\left(Q_{\phi_k}(s_t, a_t) - y_t^k\right)^2, \quad y_t^k = f_t^k + \gamma Q_{\bar{\phi}_k}\left(s_{t+1}, \boldsymbol{a}_{t+1}^-\right), \tag{12}$$

where $Q_{\bar{\phi}_k}$ is the target network, which determines $\boldsymbol{a}_{t+1}^-$ as a greedy action, and is updated periodically. The algorithm description is provided in Algorithm 4.

Algorithm 3: Generic **On-Policy** Multi-Agent Actor-Critic with Latent Objective Discovery for the Episode Update

Initialize: stochastic policies $\{\pi_{\theta_i}\}_{i\in\mathcal{N}}$ and latent value networks $\{V_{\phi_k}\}_{k=1}^{K}$, Γ_0 and $\boldsymbol{f}_0$.
Collect $\{s_t, \{o_t^i\}_{i\in\mathcal{N}}, \boldsymbol{a}_t, \boldsymbol{r}_t, s_{t+1}, \{o_{t+1}^i\}_{i\in\mathcal{N}}\}_{t=1}^{T}$ in the rollout buffer.
Update $\boldsymbol{f}$ and Γ from $\boldsymbol{r}_{1:T}$. # Online Estimation for Latent Objectives
for $k = 1, \ldots, K$ **do**
    $\hat{A}^k \leftarrow \text{GAE}(V_{\phi_k}, \{f_t^k\}_{t=1}^{T})$ # Mixed Centralized Advantage Function via GAE.
end for
# Actor update given the current Γ_T.
Update each policy with π_{θ_i} by maximizing clipped PPO or TRPO objective based on $\sum_{k=1}^{K} \gamma_i(k)\hat{A}^k$
# Critic update.
for $k = 1, \ldots, K$ **do**
    Evaluate GAE return: $\hat{R}_t^k = \hat{A}^k(s_t, \boldsymbol{a}_t) + V_{\phi_k^{\text{old}}}(s_t)$
    Update value function with V_{ϕ_k} by minimizing $\frac{1}{T}\sum_{t=1}^{T}\left[(V_{\phi_k}(s_t) - \hat{R}_t^k)^2\right]$
end for

Algorithm 4: Generic **Off-Policy** Multi-Agent Actor-Critic with Latent Objective Discovery for the Episode Update

Initialize: deterministic actors $\{\mu_{\theta_i}\}_{i\in\mathcal{N}}$ and latent centralized critics $\{Q_{\phi_k}\}_{k=1}^{K}$ with targets, Γ_0 and $\boldsymbol{f}_0$.
for $t = 1, \ldots, T$ **do**
    Sample $a_t^i = \mu_{\theta_i}(o_t^i) + \mathcal{N}_t^i$, execute $\boldsymbol{a}_t$, receive $\boldsymbol{r}_t$, observe $s_{t+1}, \{o_{t+1}^i\}_{i\in\mathcal{N}}$; store transitions in replay buffer.
    # Update Γ_t in Latent Objective Discovery
    Update Γ_t and $\boldsymbol{f}_t$ from $\boldsymbol{r}_t$ given Γ_{t-1} and $\boldsymbol{f}_{t-1}$.
    Sample $\{s_b, \{o_b^i\}_{i\in\mathcal{N}}, \boldsymbol{a}_b, \boldsymbol{r}_b, s_{b+1}, \{o_{b+1}^i\}_{i\in\mathcal{N}}\}_{b=1}^{B}$.
    Decompose $\boldsymbol{f}_b$ from $\boldsymbol{r}_b$ via Γ_t. # Projection
    # Critic update via a general TD target Q_k
    Evaluate TD target Q_k (DDPG/TD3) and update Q_{ϕ_k}

$$\min_{\phi_k} \frac{1}{B}\sum_{b}(f_b^k + \gamma\, Q_{\bar{\phi}_k}(s_{b+1}, \boldsymbol{a}_{b+1}) - Q_{\phi_k}(s_b, \boldsymbol{a}_b))^2,$$

    where $Q_{\bar{\phi}_k}$ is the target value network.
    # Actor update given the current Γ_t
    Evaluate mixed centralized critics for each agent i:

$$\bar{Q}_i(s, \boldsymbol{a}) = \sum_{k=1}^{K} \gamma_{i,t}(k)\, Q_{\phi_k}(s, \boldsymbol{a}).$$

    Update each policy μ_{θ_i} via the gradient:

$$\nabla_{\theta_i} J(\theta_i) = \frac{1}{B}\sum_{b=1}^{B} \nabla_{\theta_i}\mu_{\theta_i}(o_b^i)\, \nabla_{a^i}\bar{Q}_i(s_b, \boldsymbol{a}_b)\big|_{a^i\sim\mu_{\theta_i}}.$$

    Update target actors and value networks by Polyak averaging.
end for

D EXPERIMENT DETAILS

D.1 MULTI-AGENT PARTICLE ENVIRONMENT (MPE)

Environment Description. The Multi-Agent Particle Environments (MPE) comprise a collection of two-dimensional navigation tasks. We consider two coordination-focused scenarios:`simple_spread` and `Speaker_listerner`. Agents receive observations represented by high-level feature vectors, including the relative positions of other agents and landmarks, and select actions to navigate in two dimensions. All scenarios are fully observable except `Speaker_listerner`, which additionally involves binary communication. Because rewards depend strongly on joint actions, MPE provides a benchmark for evaluating multi-agent coordination and learning under non-stationarity. Ineffective coordination among agents can substantially degrade performance.

- `simple_spread`: In this task, N agents learn to cover N landmarks while avoiding collisions. Each agent observes its own position and velocity, together with the relative positions of the other agents and all landmarks, and selects from five discrete movement actions. The reward combines a shared coverage component, the negative sum of the distances from each landmark to its nearest agent, with agent-specific collision penalties. Therefore, the rewards are heterogenous.
- `Speaker_listerner`: In this task, a static speaker must communicate a target landmark to a listener in an environment containing three landmarks. Both agents receive a shared reward equal to the negative Euclidean distance between the listener and the target. The speaker observes only the target's colour, whereas the listener observes its own velocity, the relative positions of all landmarks, and the speaker's message. The speaker learns to encode the target using three discrete communication actions, while the listener selects from the five standard discrete movement actions used in MPE.

Implementation Details. Our implementations are directly adapted from HARL framework (Zhong et al., 2024) and deploy the same default hyperparameters in both MAPPO and LOD-MAPPO. For the LOD module, we choose $\tau = 0.2$ for the soft updating of Γ after the line search as the ablation study and set $K = 2$ ($K \leq N$) due to the underlying objectives in the MPE environment. We choose window size as 3, indicating that we estimate Γ and f_t by using data across the last $T \times 3$ steps.

Ablation Study. In Figure 6, we focus on spread (continuous) by increasing N and varying τ, the hyperparameter that controls the soft updating of Γ via $\Gamma_{t+1} = (1-\tau)\Gamma_t + \tau\Gamma$. With four agents, LOD-MAPPO generally achieves higher mean returns during the middle and later stages of training, while with six agents, LOD-MAPPO performs similarly to MAPPO. Setting $\tau = 0$ freezes the randomly initialized mixing weights, while increasing τ renders more emphasis on the newly collected data in estimating Γ_{t+1}. Figure 6 shows that an overly large or small τ, e.g., $\tau = 0.0$ or 0.6 are inferior to $\tau = 0.2$ in MPE tasks. After tuning the hyperparameters, we fix them across the remaining MPE tasks.

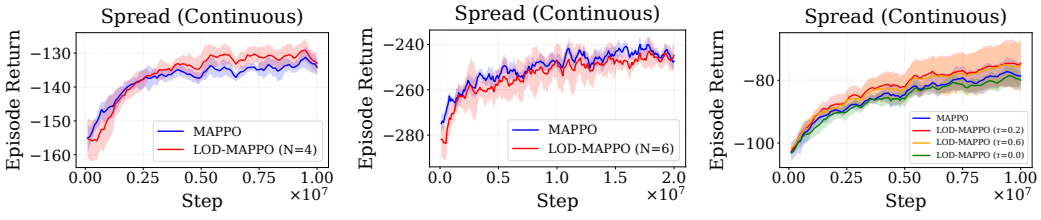


Figure 6: Ablation study in Spread (Continuous) across N and τ in the soft updating of Γ.

D.2 LEVEL-BASED FORAGING (LBF) AND MULTI-ROBOT WAREHOUSE (RWARE)

Environment Descriptions. We choose Level-Based Foraging (LBF) and Multi-Robot Warehouse (RWARE) to test our LOD-MAPPO algorithms, where the agents have access to heterogeneous rewards.

- `LBF`: The Level-Based Foraging environment consists of tasks focusing on the coordination of involved agents. The task for each agent is to navigate the grid-world map and collect items. Each agent and item is assigned a level and items are randomly scattered in the environment. In order to collect an item, agents have to choose a certain action next to the item. However, such collection is only successful if the sum of involved agents' levels is equal or greater than the item level. Agents receive rewards equal to the level of the collected item. The tasks that were selected are denoted first by the grid-world size, 15×15 means a 15 by 15 grid-world. Rewards are heterogeneous in this environment. The **cooperation version** of LBF requires agents to coordinate to be able to gather any reward, making the task very hard to solve given the sparsity of the rewards.
- `RWARE`: The multi-robot warehouse environment is a set of collaborative, partially observable multi-agent tasks simulating a warehouse operated by robots. Each agent controls a single robot aiming to collect requested shelves. At all times, N shelves are requested and each timestep a request is delivered to the goal location, a new (currently unrequested) shelf is uniformly sampled and added to the list of requests. Agents are only rewarded whenever an agent is delivering a requested shelf to a goal position. Therefore, a very specific and long sequence of actions is required to receive any non-zero rewards, making this environment **very sparsely rewarded**.

Implementation Details. Our implementations are directly adapted from EpyMARL framework (Papoudakis et al., 2021) and deploy the same default hyperparameters in both MAPPO and LOD-MAPPO. In LBF, we choose $\tau = 0.2$, while in RWARE-tiny, we set $\tau = 1 \times 10^{-3}$. By contrast, we find a smaller τ in RWARE-small can achieve better performance after a line search. This is because RWARE-small is even harder than RWARE-tiny, which requires a more stable updating (with a smaller τ) for Γ_t. The window size is 1.

Learning curves with shades. Figure 7 suggests the learning curves of LOD-MAPPO and MAPPO with the shades that indicate the standard deviations. It showcases that algorithms in LBF-Coop have more variations than LBF, indicating a more difficult environment. In addition, algorithms in RWARE also have very large variations due to the very sparse rewards in this environment.

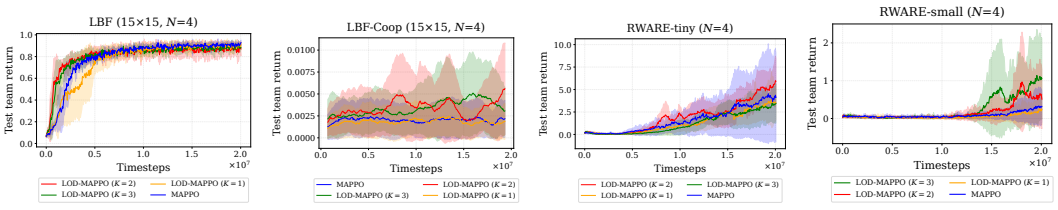


Figure 7: Comparison between LOD-MAPPO across a wide range of K and MAPPO in LBF and RWARE environments. The learning curves are the means averaged across five random seeds, and the shades indicate standard deviations.

D.3 AGENTIC RL

D.3.1 AGENT LIGHTNING CHARTQA

We implement the ChartQA experiment based on the Agent Lightning agentic RL framework (He et al., 2026). The workflow consists of five successive agent nodes corresponding to the original ChartQA pipeline: *Chart Analyzer*, *Data Extractor*, *Answer Calculator*, *Answer Checker*, and *Answer Refiner*. Each agent selects one strategy from its role-specific strategy set, and the sequence of decisions determines the final answer. We treat these five workflow nodes as cooperating heterogeneous agents and apply MAPPO and LOD-MAPPO for policy optimization.

Language Model and Environment. We use Qwen2.5 as the underlying language model for all agents. The model is used only to provide LLM-based grounding of strategy effectiveness. Specifically, for each role and strategy, we obtain empirical strategy quality estimates by evaluating the LLM on ChartQA tasks. The ChartQA environment contains 200 generated tasks with diverse question types, including maximum, minimum, summation, difference, average, and comparison queries.

Example of LLM-based Strategy Grounding

User Prompt

Role: Data Extractor
Strategy: Extract only queried values.

Chart:
C1: 35, C2: 82, C3: 57, C4: 41

Question:
What is the average value of C1, C2, and C3?

Return the chart values needed to answer the question.

Qwen2.5 Output

The queried values are:

$$\text{C1} = 35, \qquad \text{C2} = 82, \qquad \text{C3} = 57.$$

Evaluation
The extracted values are compared with the ground-truth chart values. The role–strategy pair receives a binary success score of 1 if the extracted information is correct and 0 otherwise. Aggregating these scores over ChartQA tasks produces the empirical strategy effectiveness used for RL training.

RL Training Configuration. All methods are trained using MAPPO or LOD-MAPPO with the same hyperparameters. For MAPPO, we use a batch size of 128, four MAPPO optimization epochs, and a minibatch size of 256. The actor and critic learning rates are set to 10^{-3} and 3×10^{-3}, respectively. We use PPO clipping coefficient $\epsilon = 0.2$, entropy coefficient 0.003, value loss coefficient 0.5, and gradient clipping with maximum norm 0.5. The hidden dimension of all policy and value networks is 64. For LOD-MAPPO, we evaluate different numbers of latent objectives with $K \in \{1, 2, 3\}$. The latent loading matrix is estimated from counterfactual local advantages according to Eq. 7 while using ridge-regularized factorization. We use ridge coefficient 10^{-3}, update the loading matrix every 256 environment steps using a window of 1000 recent samples, and optimize the factorization for 400 iterations with learning rate 0.08.

Reward and Evaluation. All agents receive the same binary task-level reward from the ChartQA workflow. The reward is one when the final answer produced by the complete workflow is correct

and zero otherwise. For LOD-MAPPO, this shared reward is first converted into agent-specific counterfactual local advantages, which are then factorized into latent objective signals for policy optimization. We train each method with five random seeds and report the moving-average success rate during training. The curves are smoothed using a moving window of 300 training steps, and the shaded regions represent the standard error across seeds.

D.3.2 AGENT LIGHTNING MUSIQUE

We also implement the MuSiQue experiment based on the Agent Lightning agentic RL framework (He et al., 2026). The workflow consists of four successive retrieval decision positions, which we treat as cooperating agents. At each step, the corresponding agent selects one retrieval strategy from four alternatives: *Direct*, which searches for the main fact needed to answer the question; *Decompose*, which targets an unresolved intermediate relation; *Entity*, which focuses on a relevant entity and relation; and *Evidence*, which searches for information missing from the currently available evidence. The selected strategy is converted into a search query and used to retrieve supporting information. We treat the four retrieval turns as heterogeneous agents and apply MAPPO and LOD-MAPPO for policy optimization.

Example of LLM-based Strategy Grounding

User Prompt

Question:
Which city was the author of *Book A* born in?

Evidence Already Found:
The author of *Book A* is John Smith.

Turn:
2 of 4

Search Strategy:
ENTITY: center the query on the relevant named entity and the relation needed next.

Generate the next search query. Output only the query.

Qwen2.5 Output

John Smith birthplace city

Evaluation
The generated query is used for retrieval over candidate passages. The strategy receives success score 1 if the retrieved top-k passages contain an unresolved supporting paragraph, and 0 otherwise. Aggregating these scores over MuSiQue tasks produces empirical strategy effectiveness estimates.

Language Model and Environment. We also use Qwen2.5 as the underlying language model for all retrieval agents. Specifically, for each retrieval turn and strategy, Qwen generates a search query conditioned on the current question and available evidence. The generated query is evaluated using the MuSiQue supporting-paragraph annotations: a strategy succeeds if its retrieved passages contain an unresolved supporting paragraph. Aggregating these outcomes over benchmark tasks provides empirical strategy effectiveness estimates for RL training.

The MuSiQue environment contains 200 real benchmark tasks with multi-hop questions requiring information aggregation from multiple sources. Each task contains supporting evidence annotations and is categorized according to its hop number. The retrieval process consists of four temporal decision steps, where each step selects among the same four retrieval strategies.

RL Training Configuration. The hyperparameters of MAPPO and LOD-MAPPO are set the same as ChartQA above.

Reward and Evaluation. The reward and evaluation strategy are also same as ChartQA. We train each method with five random seeds and report the moving-average success rate during training. The curves are smoothed using a moving window of 300 training steps, and shaded regions represent the standard error across seeds.