Heterogeneity in Multi-Agent Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Heterogeneity is a fundamental property in multi-agent reinforcement learning (MARL), which is closely related not only to the functional differences of agents, but also to policy diversity and environmental interactions. However, the MARL field currently lacks a rigorous definition and deeper understanding of heterogeneity. This paper systematically discusses heterogeneity in MARL from the perspectives of *definition*, *quantification*, and *utilization*. First, based on an agent-level modeling of MARL, we categorize heterogeneity into five types and provide mathematical definitions. Second, we define the concept of heterogeneity distance and propose a practical quantification method. Third, we design a heterogeneity-based multi-agent dynamic parameter sharing algorithm as an example of the application of our methodology. Case studies demonstrate that our method can effectively identify and quantify various types of agent heterogeneity. Experimental results show that the proposed algorithm, compared to other parameter sharing baselines, has better interpretability and stronger adaptability. The proposed methodology will help the MARL community gain a more comprehensive and profound understanding of heterogeneity, and further promote the development of practical algorithms.

1 Introduction

Multi-agent reinforcement learning (MARL) has achieved success in various real-world applications, such as swarm robotic control [Kalashnikov et al., 2018], autonomous driving [Zhou et al., 2021], and large language model fine-tuning [Ma et al., 2024]. However, most MARL studies focus on policy learning for homogeneous multi-agent systems (MAS), overlooking in-depth discussions of heterogeneous multi-agent scenarios [Ning and Xie, 2024]. *Heterogeneity* is a common phenomenon in multi-agent systems. For example, in nature, different species of fish collaborate to find food [Burns et al., 2019]; in human society, diverse teams demonstrate higher intelligence and resilience [Dall'Anese et al., 2013, Young, 1993]; and in artificial systems, aerial drones and ground vehicles cooperate to monitor forest fires [Lwowski et al., 2017]. Heterogeneity can enhance system functionality, reduce costs, and improve robustness, but effectively leveraging heterogeneity remains a key challenge in multi-agent system [Bennett, 2024]. As an approach of learning through environmental interactions, MARL can effectively enable multi-agent systems to learn collaborative policies. Hence, exploring heterogeneity from a reinforcement learning perspective would significantly broaden the applicability of MARL.

In the current MARL field, although some works explicitly or implicitly mention agent heterogeneity, only a few focus on its definition and identification. Regarding explicit discussion of heterogeneity, studies have explored communication issues [Seraj et al., 2021], credit assignment [Yu et al., 2024], and zero-shot generalization [Guo et al., 2024] in heterogeneous MARL. However, these works limit their focus to agents with clear functional differences and lack definitions of agent heterogeneity. On the other hand, many studies explore policy diversity in MARL. Some encourage agents to learn distinguishable behaviors based on identity or trajectory information [Jiang and Lu, 2021, Li et al., 2021], some works group agents using specific metrics [Wang et al., 2021, Christianos et al., 2021], and some quantify policy differences [Bettini et al., 2023b, Hu et al., 2024] and design algorithms to control policy diversity [Bettini et al., 2024]. However, these works do not adequately address where policy diversity originates or how it fundamentally relates to agent differences. In terms of defining and classifying heterogeneity in MARL, [Bettini et al., 2023a] divides heterogeneity into physical and behavioral types but lacks a mathematical definition. [Seraj et al., 2021] provides extended POMDP for heterogeneous MARL settings, but do not classify or define heterogeneity. Others introduce the concept of local transition heterogeneity [Yu et al., 2024], but does not cover all elements of MARL. Overall, heterogeneity is not only a characteristic that exists in MAS with traditional functional differences, but also a fundamental property across the entire MARL field. Currently, there is still a lack of *systematic analysis of agent heterogeneity from the MARL perspective*.

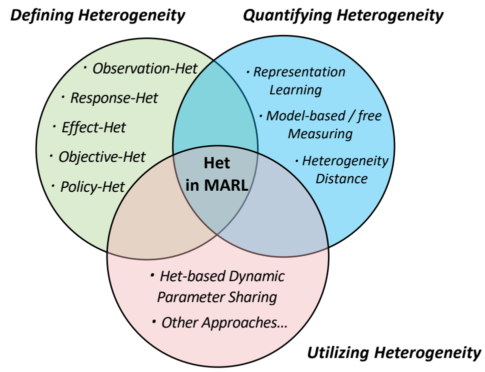

Figure 1: Our Philosophy. We aim to systematically discuss heterogeneity in MARL, establishing methodologies for defining, quantifying and utilizing heterogeneity.

To fill the aforementioned gaps, we conduct a series of studies on defining, quantifying, and utilizing heterogeneity from the perspective of MARL, the philosophy of our study can be found in Figure 1. Our contributions are summarized as follows:

- **Defining Heterogeneity:** Based on an agent-level model of MARL, we categorize heterogeneity into observation heterogeneity, response transition heterogeneity, effect transition heterogeneity, objective heterogeneity, and policy heterogeneity, and provide corresponding definitions.

- **Quantifying Heterogeneity:** We define the heterogeneity distance, and propose a quantification method based on representation learning, applicable to both model-free and model-based settings. Additionally, we give the concept of meta-transition heterogeneity to quantify agents' comprehensive heterogeneity.

- **Utilizing Heterogeneity:** We develop a multi-agent dynamic parameter-sharing algorithm based on heterogeneity quantification, which offers better interpretability and fewer task-specific hyperparameters compared to other related parameter-sharing algorithms.

In this paper, we adopt a discussion approach that progresses *from theory to practice* and *from general to specific*. The overall structure is organized as follows: Section 2 introduces the agent-level modeling of the MARL primal problem; Section 3 provides the classification and definition of heterogeneity in MARL; Section 4 proposes the method for quantifying heterogeneity and presents case studies; Section 5 describes the dynamic parameter-sharing algorithm; Section 6 provides the related experimental results; and Section 7 summarizes the entire paper.

2 Preliminaries

Primal Problem of MARL. In this paper, we use Partially Observable Markov Game (POMG) [Littman, 1994, Kochenderfer et al., 2022] as the general model for the primal problem of MARL.[1] To better study agent heterogeneity, we adopt an agent-level modeling approach similar to that in [Seraj et al., 2021, Gronauer and Diepold, 2022]. A POMG is defined as an 8-tuple, represented as follows:

$$\text{POMG} := \langle N, \{S^i\}_{i\in N}, \{O^i\}_{i\in N}, \{A^i\}_{i\in N}, \{\Omega^i\}_{i\in N}, \{\mathcal{T}^i\}_{i\in N}, \{r_i\}_{i\in N}, \gamma\rangle, \tag{1}$$

Among all elements in equation 1, N is the set of all agents, $\{S^i\}_{i\in N}$ is the global state space which can be factored as $\{S^i\}_{i\in N} = \times_{i\in N} S^i \times S^E$, where S^i is the state space of an agent i, and S^E is the environmental state space, corresponding to all the non-agent components. $\{O^i\}_{i\in N} = \times_{i\in N} O^i$ is

[1]POMG is an extension of POMDP for multi-agent settings, with the basic extension path being MDP $\rightarrow$ POMDP $\rightarrow$ POMG [Sun et al., 2023]. Please refer to Appendix C to see a more detailed explanation of POMG.

the joint observation space and $\{A^i\}_{i\in N} = \times_{i\in N} A^i$ is the joint action space of all agents. $\{\Omega^i\}_{i\in N}$ is the set of observation functions. $\{\mathcal{T}^i\}_{i\in N} = (\mathcal{T}^1, \cdots, \mathcal{T}^{|N|}, \mathcal{T}^E)$ is the collection of all agents' transitions and the environmental transition. Finally, $\{r_i\}_{i\in N}$ is the set of reward functions of all agents and γ is the discount factor.

Here, we give the independent and dependent variables for each function and their notation. At each time step t, an agent i receives an observation $o_t^i \sim \Omega^i(\cdot|\hat{s}_t)$, where $\hat{s}_t \in \{S^i\}_{i\in N}$ is the global state at time t. Then, agent i makes a decision based on its observation, resulting in an action $a_t^i \sim \pi_i(\cdot|o_t^i)$. The environment then collects actions from all agents to form the global action $\hat{a}_t = (a_t^1, \ldots, a_t^{|N|})$. We assume that the local state transition of agent i is influenced by the global state and global action, so its local state transitions to a new state $s_{t+1}^i \sim \mathcal{T}^i(\cdot|\hat{s}_t, \hat{a}_t)$. Similarly, the states of other agents and the environment also transition, yielding the next global state $\hat{s}_{t+1} = (s_{t+1}^1, \ldots, s_{t+1}^{|N|}, s_{t+1}^E) \sim (\mathcal{T}^1(\cdot|\hat{s}_t, \hat{a}_t), \ldots, \mathcal{T}^{|N|}(\cdot|\hat{s}_t, \hat{a}_t), \mathcal{T}^E(\cdot|\hat{s}_t, \hat{a}_t)) = \{\mathcal{T}^i\}_{i\in N}(\cdot|\hat{s}_t, \hat{a}_t)$. At the same time, all agents receive rewards, with the reward for a specific agent i given by $r_t^i \sim r^i(\cdot|\hat{s}_t, \hat{a}_t)$.

The objective of MARL is to solve POMG by finding an optimal joint policy that maximizes the cumulative reward for all agents. We denote the individual optimal policy for agent i as π_i^* and the optimal joint policy as $\hat{\pi}^*$, which can be expressed as $\hat{\pi}^* = (\pi_1^*, \ldots, \pi_{|N|}^*)$. The optimal joint policy for a POMG can be obtained through the following equation:

$$\pi_i^* = \arg\max_{\hat{\pi}} \mathbb{E}_{\hat{\pi}} \left[\sum_{k=0}^{\infty} \gamma^k \sum_{i\in N} r_{t+k}^i \middle| \hat{s}_t = \hat{s}_0 \right], \tag{2}$$

where γ is the discount factor, and the expectation is taken over the trajectories induced by the joint policy $\hat{\pi}$ starting from the initial global state $\hat{s}_0$.

3 Definition and Taxonomy of Heterogeneity in MARL

Heterogeneity in MAS. Our goal is to define agent heterogeneity from the perspective of MARL. Before achieving this, we need to discuss heterogeneity in MAS across various disciplines. Early studies [Dudek et al., 1996, Parker, 2000] define heterogeneity as differences in physical structure or functionality of agents, which aligns with common understanding. Later work [Panait and Luke, 2005] describes heterogeneity as differences in agent behavior, further expanding its meaning. Recently, [Bennett, 2024] points out that heterogeneity may be a complex phenomenon, related not only to the inherent properties of agents, but also to their interactions with environment. Thus, heterogeneity in MARL should not be limited to inherent functional differences of agents, but should also fully consider various coupling effects of agents within the environment.

Heterogeneity in MARL. In the context of MARL, the fundamental modeling of MARL and its primal problem provides considerable convenience for defining heterogeneity. This modeling clearly specifies all MARL elements, delineating the boundaries of the problem discussion [2] and ensuring the completeness of the discussion.

We focus on the heterogeneity *among agents* within a same POMG. As discussed in Section 2, the function in a POMG can connect agent-level elements. Therefore, we categorize agent heterogeneity into five types centered around the functions. This approach not only avoids overly redundant classification but also ensures comprehensive coverage of each agent-level element. Regarding definition, the condition for heterogeneity is obtained by *taking the negation of the necessary and sufficient conditions for homogeneity.*

Specifically, these five types of heterogeneity and their related definitions are as follows:

• *Observation heterogeneity* describes the differences of agents in observing global information. The relevant elements include the agent's observation space and observation function.

Definition 1. Agents i and j are observation heterogeneous if the following conditions do not hold at the same time: ① $O^i = O^j$; ② $\forall \hat{s} \in \{S^i\}_{i\in N}, \Omega^i(\cdot|\hat{s}) = \Omega^j(\cdot|\hat{s})$.

[2] In this paper, we focus on the heterogeneity of MARL under the conventional POMG problem. Additional discussions on unconventional heterogeneity types are provided in Appendix D.

• *Response transition heterogeneity* describes the differences of agents in how their state transitions are affected by global environment components (*environment-to-self*). The relevant elements include the agent's state space and local state transition function.

Definition 2. Agents i and j are response transition heterogeneous if the following conditions do not hold at the same time: ① $S^i = S^j$; ② $\forall \hat{s} \in \{S^i\}_{i\in N}$, $\hat{a} \in \{A^i\}_{i\in N}$, $\mathcal{T}^i(\cdot|\hat{s},\hat{a}) = \mathcal{T}^j(\cdot|\hat{s},\hat{a})$.

• *Effect transition heterogeneity* describes the differences of agents in how their states and actions impact global state transitions (*self-to-environment*). The relevant elements include the agent's action space, state space, and global state transition function.

Definition 3. Agents i and j are effect transition heterogeneous if the following conditions do not hold at the same time: ① $S^i = S^j$; ② $A^i = A^j$; ③ $\forall s' \in S^{-i}, a' \in A^{-i}, s \in S^i, a \in A^i$, $\mathcal{T}^{-i}(\cdot|s', s, a', a) = \mathcal{T}^{-j}(\cdot|s', s, a', a)$.

In the above definition, $S^{-i} = \times_{k\in N, k\neq i} S^k \times S^E$ represents the joint state space of all agents except agent i, reflecting the influence of the agent on other states. Similarly, A^{-i} denotes the joint action space excluding agent i, and $\mathcal{T}^{-i}$ is the collection of state transitions excluding agent i.

• *Objective heterogeneity* describes the differences of agents in the objective they aim to achieve. The relevant element is the agent's reward function.

Definition 4. Agents i and j are objective heterogeneous if the following condition do not hold:

① $\forall \hat{s} \in \{S^i\}_{i\in N}$, $\hat{a} \in \{A^i\}_{i\in N}$, $r^i(\cdot|\hat{s},\hat{a}) = r^j(\cdot|\hat{s},\hat{a})$.

• *Policy heterogeneity* describes the differences of agents in their autonomous decision-making based on observations. The relevant elements include the observation space, action space, and policy.

Definition 5. Agents i and j are policy heterogeneous if the following conditions do not hold at the same time: ① $O^i = O^j$; ② $A^i = A^j$; ③ $\forall o \in O^i$, $\pi_i(\cdot|o) = \pi_j(\cdot|o)$.

In the five types of heterogeneity mentioned above, we assume that all functions follow the Markov property, making them independent of the agent's trajectory. Therefore, the first four types of heterogeneity can be considered environment-related, which reflect the heterogeneity in the MARL primal problem. The last type describes the policy heterogeneity of agents before, during, and after training, which reflects the heterogeneity of optimization objectives (policies) in the primal problem.

4 Quantifying Heterogeneity in MARL

4.1 Heterogeneity Distance Based on Representation Learning

Heterogeneity Distance. In this section, we present the method to quantify the above five types of heterogeneity. According to the definition, each type of heterogeneity corresponds to a core function which connects relevant elements in the heterogeneity type. Therefore, we quantify the differences in these core functions to characterize the degree of heterogeneity.[3] To make the quantification results simpler and more practical, we propose the concept of heterogeneity distance.

Let the core function corresponding to a certain heterogeneity type F be denoted as $y \sim F(\cdot|x)$. The formula for calculating the F-heterogeneous distance between two agents i and j is given by:

$$d_{ij}^F = \int_{x\in X} D[F_i(\cdot|x) \parallel F_j(\cdot|x)] \cdot p(x)\, dx, \tag{3}$$

where X is the space of independent variables, $p(x)$ is the probability density function, and $D[\cdot \parallel \cdot]$ is a measure that quantifies the difference between distributions. The core idea of heterogeneity distance is to examine the cumulative differences between two agents' functions throughout the space of independent variables, which captures any potential local differences. When the independent variables x consist of multiple factors, the above integral becomes a multivariate integral. Based on Equation 3, we provide the specific expressions for quantifying all heterogeneous distances in Appendix F and discuss the properties of heterogeneous distance below.

[3]Quantifying space elements is feasible and even easier to implement. But a space element may appear across multiple heterogeneity types, making it unsuitable as unique identifiers for specific heterogeneity types.

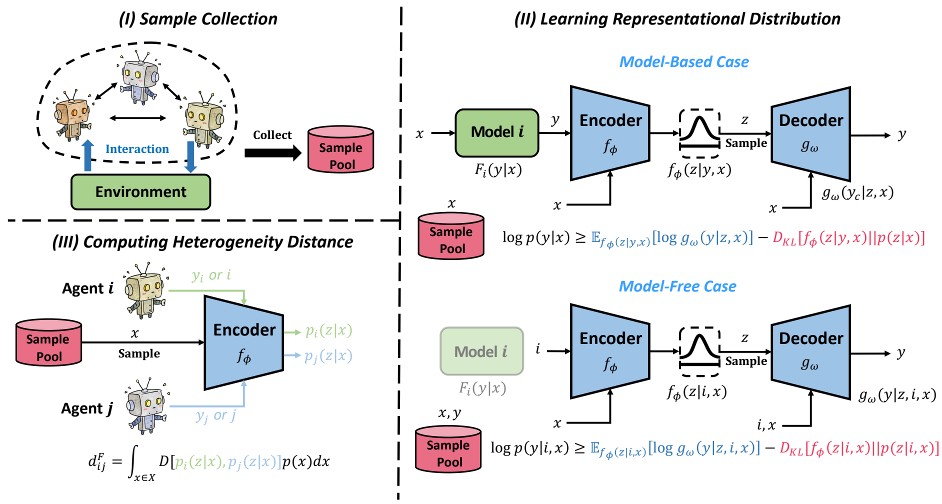

Figure 2: The method of measuring heterogeneity distance based on representation learning.

Proposition 1. (*Properties of Heterogeneity Distance*) ① *Symmetry*: $d_{ij}^F = d_{ji}^F$; ② *Non-negativity*: $d_{ij}^F \geq 0$; ③ *Identity of indiscernibles*: $d_{ij}^F = 0$ if and only if agents i and j are F-homogeneous; ④ *Triangle inequality*: $d_{ij}^F \leq d_{ik}^F + d_{kj}^F$ $(i, j, k \in N)$. This proposition holds as long as the measure D satisfies ①②③④. The proof is provided in Appendix E.

Practical Method. Although the heterogeneity distance has a simple form, some issues may arise during practical computation. First, computing the distribution distance via sampling is computationally complex, while computing the distance using analytical solutions requires knowing the distribution type. In real-world scenarios, the distributions may be unknown or of different types [4]. Second, the independent variable space may be very large, making traversal-based computation infeasible.

For the first issue, our approach is to **standardize the original distributions**. By learning a representation mapping, for all independent variables x, a measurable distribution $p_i(z|x)$ is used to capture the characteristics of the original distribution $F_i(y|x)$, replacing the original one for measure computation. For the second issue, our approach is **sampling based on the interaction between agents and the environment**. Instead of simply traversing the space or using random policy exploration for sampling, we construct a sample pool using trajectories from the training phase of MARL. This significantly reduces computational load and filters out excessive marginal spaces that interfere with MARL, benefiting the use of heterogeneity distance in subsequent MARL tasks (Section 5). Combining these ideas, we propose a practical method as shown in Figure 2.

In the first step, the agents interact with the environment during MARL training to build a sample pool. Notably, the sample pool data is shuffled to ensure that the learned function follows the Markov property (independent of historical information), similar to the original function.

In the second step, the representational distributions are learned. We discuss this in both model-based and model-free settings, corresponding to cases where the function is known and unknown, respectively. We adopt the conditional variational autoencoder (CVAE) framework [Sohn et al., 2015] for representation learning. In the model-based case, CVAE performs a reconstruction task [Lopez-Martin et al., 2017]. The optimization goal is to maximize the likelihood of the reconstructed variable $\log p(y|x)$. Through derivation, we obtain the evidence lower bound (ELBO) as:

$$ELBO_{\text{model-based}} = \mathbb{E}_{f_\phi(z|y,x)} \left[\log g_\omega(y|z,x)\right] - D_{KL} \left[f_\phi(z|y,x) \parallel p(z|x)\right], \tag{4}$$

where f_ϕ and g_ω represent the encoder and decoder, respectively, and $p(z|x)$ is the prior conditional latent distribution. We designed the relevant loss based on ELBO, including a reconstruction term and a prior-matching term. The derivation for this part can be found in Appendix H.

[4]For example, the action distribution of an agent i is a Gaussian distribution, while that of agent j is a bimodal distribution.

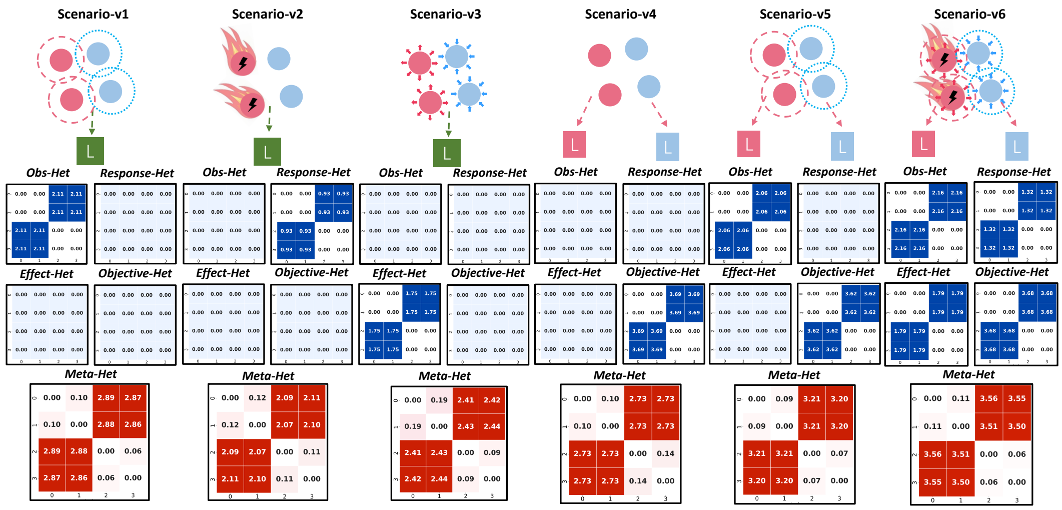

Figure 3: The scenario illustration and heterogeneity distance matrices in our case study. In v1, the observations of agents in different groups are shuffled in different orders. In v2, the max move speed of agents in different groups is different. In v3, one group of agents applies repulsive force to surrounding entities, while the other applies attractive force. In v4, agents in different groups need to move to different landmarks. In v5, both the observations and objectives of agents are heterogeneous. In v6, all the above properties of agents are heterogeneous. Below each scenario illustration, the corresponding heterogeneity distance matrices are shown. Specifically, *Obs-Het*, *Response-Het*, *Effect-Het*, *Objective-Het*, and *Meta-Het* correspond to observation / response transition / effect transition / objective / meta-transition heterogeneity, respectively.

In the model-free case, CVAE essentially performs a prediction task [Zhang et al., 2021], capturing the model characteristics of each agent. The optimization goal is to maximize the likelihood of the predicted variable y given conditions i and x, where i is the agent ID. Similarly, we derive the corresponding ELBO:

$$ELBO_{\text{model-free}} = \mathbb{E}_{f_\phi(z|i,x)} \left[\log g_\omega(y|z,i,x)\right] - D_{KL}\left[f_\phi(z|i,x) \parallel p(z|i,x)\right]. \tag{5}$$

In the third step, the heterogeneity distances for multi-agents are computed. For each x, we obtain the distribution representation using the encoder in either the model-based or model-free manner. The distance under a specific x is computed using the *Wasserstein distance* [Vaserstein, 1969] of the prior distribution (*standard Gaussian*). The heterogeneity distance is then calculated via multi-rollout Monte Carlo sampling. In practice, we parallelize this operation [5], enabling simultaneous computation of distances between all agents on GPUs, significantly improving computational efficiency.

Meta-Transition. The aforementioned method can quantify the heterogeneity of agents for specific types. In practical applications, researchers may also want to quantify the **comprehensive** heterogeneity of agents to enable operations such as grouping. To this end, we give the *Meta-Transition* model (see Appendix G for details). By measuring the differences between meta-transitions, the comprehensive heterogeneity related to environment can be quantified. We refer to this as the meta-transition heterogeneity distance.

4.2 Case Study

We design a multi-agent spread scenario for case study. In the basic scenario, there are two groups, each with two agents, and their goal is to move close to randomly generated landmarks. Based on the basic scenario, we create 6 extended versions to show the quantitative results of different types of heterogeneity and meta-transition heterogeneity. As shown in Figure 3, the first 4 versions correspond to the 4 environment-related types of heterogeneity, while the last 2 versions represent

[5] Our code is provided in the supplementary material.

cases where multiple types of heterogeneity exist. We use the model-based manner to compute the four heterogeneity distance matrices mentioned above, and the model-free manner to compute the meta-heterogeneity distance matrix for the agents.

The results show that for each type of heterogeneity, our method can accurately capture and identify the differences. For meta-transition heterogeneity, the distance between agents in the same group is much smaller than that in different groups. Moreover, as the number of heterogeneity types increases, the distance between different groups also increases. These results demonstrate the effectiveness of our method for various environment-related heterogeneities.

We further quantify the policy heterogeneity distance (*Policy-Het*) and meta-transition heterogeneity distance (*Meta-Het*) of agents during the training process. We select two algorithms at the extreme ends of parameter sharing: fully parameter sharing (FPS) and no parameter sharing (NPS) for training in the above scenarios. Figure 4 shows the measurement results at 500 and 1500 updates. From the *Policy-Het* results, the policy distance can effectively reveal the evolutionary relationship of agent policy differences in MARL. From the *Meta-Het* results, the comprehensive agent heterogeneity measurement remains consistent across different learning algorithms, and can identify environmental heterogeneous characteristics in scenarios more rapidly compared to policy evolution.

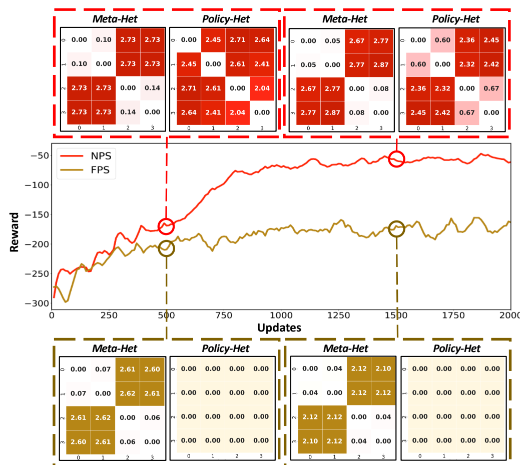

Figure 4: Meta-transition heterogeneity and policy heterogeneity distance matrices during training in our case study.

5 Multi-Agent Dynamic Parameter Sharing Based on Heterogeneity Quantification: An Application

Based on the case study in Section 4.2, the proposed method can not only accurately quantify all types of heterogeneity, but also the comprehensive heterogeneity among agents. Additionally, the method is independent of the parameter-sharing type used in MARL and can be deployed online, thereby further enhancing its practicality. In this section, we provide a practical application of our methodology to demonstrate its potential in empowering MARL.

We select parameter sharing in MARL as our application context. As a common technique in MARL, parameter sharing can reduce computational consumption while improving sample utilization efficiency [KIM and Sung, 2023], but its excessive use may inhibit agents' policy heterogeneity expression [Hu et al., 2024]. Many works have attempted to find a balance between parameter sharing and policy heterogeneity [Li et al., 2024b]. However, existing approaches suffer from two main problems: *poor interpretability*, unable to explain why policy heterogeneity is necessary and to what extent; and *poor adaptability*, manifested by numerous task-specific hyperparameters and inability to dynamically adapt policy training. (For a more detailed discussion of these algorithms, see the experimental section 6.1)

To address these issues, we propose a Heterogeneity-based multi-agent Dynamic Parameter Sharing algorithm (HetDPS) with two core ideas(More details can be found in Appendix I):

♠ **Grouping agents for parameter sharing through heterogeneity distances**. We utilize distance-based clustering methods to group agents, thus avoiding the introduction of task-specific hyperparameters like group number [Christianos et al., 2021, Li et al., 2024a] or fusion thresholds [Hu et al., 2024]. The heterogeneity distance matrices also enhance the algorithm's interpretability.

♣ **Periodically quantifying heterogeneity and modifying agents' parameter sharing paradigm**. This can help the sample pool become more aligned with policy training. This approach can also help policies escape local optima [Lyle et al., 2024], the effectiveness of such a mechanism has been verified in the MARL domain [Li et al., 2024b], and even in broader RL areas such as large model fine-tuning [Noukhovitch et al., 2023, Ma et al., 2024].

6 Experiments

In the experimental section, we conduct comprehensive comparisons between HetDPS and other parameter sharing algorithms. Beyond performance comparisons, we also analyze the heterogeneity characteristics of each MARL task with our proposed methodology, to demonstrate the algorithm's interpretability. Additionally, we conduct hyperparameter experiments and efficiency and resource consumption experiments to show the adaptability and practicality of HetDPS.

6.1 Experimental Setups

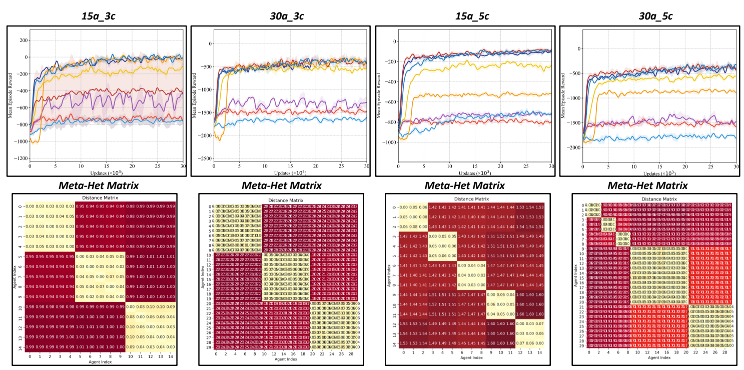

Figure 5: Results on Partical-based Multi-agent Spreading.

Environments. Partical-based Multi-agent Spreading [Hu et al., 2024] is a typical environment in the policy diversity domain. In this environment, multiple agents are randomly generated in the center of the map, while multiple landmarks are randomly generated near the periphery. Both agents and landmarks have various colors, and agents need to move to landmarks with matching colors. Additionally, agents need to form tight formations when they reach the vicinity of landmarks. We employ 4 typical tasks, corresponding to different numbers and color distributions, as detailed in Table 1. **The StarCraft Multi-Agent Challenge (SMAC)** [Samvelyan et al., 2019] is a popular MARL benchmark, where multiple ally units controlled by MARL algorithms aim to defeat enemy units controlled by the game's built-in AI.

Baselines and training. We compare HetDPS with other parameter sharing baselines, as listed in Table 2. We analyze these baselines along three dimensions: parameter sharing paradigm, adaptability, and relationship with heterogeneity utilization. As seen from the table, current methods can not effectively utilize heterogeneity. Although some methods implicitly use certain heterogeneity quantification results, the elements they involve are not comprehensive. MADPS, as the only method that explicitly uses policy distance for dynamic grouping, relies on the assumption that policy learning can effectively capture heterogeneity, which lacks practicality. We use official implementations of the baselines where available. For further discussion on related work and experiments in this paper, see the supplementary materials.

Table 1: Task information for particle-based multi-agent spreading.

Task	Agent Type Distribution
15a_3c	$5-5-5$
30a_3c	$10-10-10$
15a_5c	$3-3-3-3-3$
30a_5c	$3-3-3-12-9$

6.2 Results

Performance and interpretability. We tested the performance of all comparison algorithms in the two environments mentioned above. The reward curves and corresponding heterogeneity distance matrices are shown in Figure 5 and Figure 6. From the reward curve results, we can see that HetDPS achieves either optimal or comparable results in all tasks above.

We quantified the meta-transition heterogeneity distances for all tasks. The results show that our heterogeneity quantification results in the Multi-agent Spreading scenario are highly consistent with

Table 2: Comparison of different methods and their properties.

Method	Paradigm	Adaptive	Relation to Heterogeneity Utilization
NPS	No Sharing	No	None
FPS	Full Sharing	No	None
FPS+id	Full Sharing	No	None
Kaleidoscope [Li et al., 2024b]	Partial Sharing	Yes	No utilization, increases agent policy heterogeneity as the bias
SePS [Christianos et al., 2021]	Group Sharing	No	Implicitly utilizes objective heterogeneity and response transition heterogeneity
AdaPS [Li et al., 2024a]	Group Sharing	Yes	Implicitly utilizes objective heterogeneity and response transition heterogeneity
MADPS [Hu et al., 2024]	Group Sharing	Yes	Explicitly utilizes policy heterogeneity only
HetDPS (ours)	Group Sharing	Yes	Explicitly utilizes heterogeneity, leveraging heterogeneous distance

Table 3: Training efficiency metrics across different methods. Results are normalized with respect to the FPS method, and averaged across all tasks.

	NPS	FPS	FPS+id	Kaleidoscope	SePS	AdaPS	MADPS	HetDPS (ours)
Training Speed	0.952x	1.000x	0.992x	0.974x	0.986x	0.614x	0.539x	0.712x

the type distribution in Table 1. This demonstrates the effectiveness of our method in identifying agent heterogeneity. Additionally, we made some interesting discoveries in the SMAC environment. We found that in simpler tasks like *3s5z* and *MMM*, the agent heterogeneity quantification results often do not closely match the original agent types. In *MMM*, agents even tend toward homogeneous policies to improve training efficiency. However, in more difficult tasks such as *3s5z_vs_3s6z* and *MMM2*, agents' quantification results closely match their original types to achieve better coordination. This confirms our view that agent heterogeneity is related not only to the agents' original functional attributes but also to how agents interact with the environment.

Cost Analysis. We conducted an experiment to investigate training efficiency. The experimental results are shown in Table 3. The results indicate that although our method introduces periodic heterogeneity quantification, it does not significantly reduce algorithm efficiency.

7 Conclusion

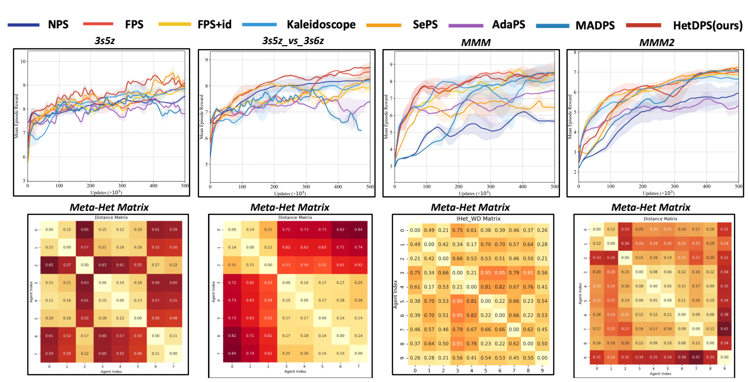

Figure 6: Results on StarCraft Multi-Agent Challenge.

Heterogeneity manifests in various aspects of MARL. It is not only related to the inherent properties of agents themselves but also to the coupling factors arising from agent-environment interactions. Consequently, agents that appear homogeneous may develop heterogeneity under environmental influences. In this paper, we categorize heterogeneity in MARL into five types and provide respective definitions. Meanwhile, we propose methods for quantifying these heterogeneity types and conduct case studies. Under our theoretical framework, policy diversity is merely a manifestation of policy heterogeneity, fundamentally originating from the division of labor necessitated by agents' environmental heterogeneity (*cause*), serving as an inductive bias (*result*) for solving optimal joint policies. Thus, we introduce the quantification of heterogeneity as prior knowledge into multi-agent parameter-sharing learning. The result is HetDPS, an algorithm with strong interpretability and adaptability. HetDPS is not the endpoint of our research, but rather a starting point for heterogeneity applications. We believe that by systematically studying the definition, quantification, and application of heterogeneity, future MARL research will more profoundly understand the complex collaboration mechanisms between agents, and pave the way for more intelligent and adaptive collective decision-making systems.

References

Richard Bellman. A markovian decision process. *Journal of mathematics and mechanics*, pages 679–684, 1957.

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

Daniel S Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of markov decision processes. *Mathematics of operations research*, 27(4): 819–840, 2002.

Matteo Bettini, Ajay Shankar, and Amanda Prorok. Heterogeneous multi-robot reinforcement learning. In *AAMAS*, 2023a.

Matteo Bettini, Ajay Shankar, and Amanda Prorok. System neural diversity: Measuring behavioral heterogeneity in multi-agent learning. *arXiv preprint arXiv:2305.02128*, 2023b.

Matteo Bettini, Ryan Kortvelesy, and Amanda Prorok. Controlling behavioral diversity in multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 3611–3636. PMLR, 2024.

Alicia L Burns, Alexander DM Wilson, and Ashley JW Ward. Behavioural interdependence in a shrimp-goby mutualism. *Journal of Zoology*, 308(4):274–279, 2019.

Anthony Rocco Cassandra. *Exact and approximate algorithms for partially observable Markov decision processes*. Brown University, 1998.

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*, pages 1989–1998. PMLR, 2021.

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.

Gregory Dudek, Michael RM Jenkin, Evangelos Milios, and David Wilkes. A taxonomy for multi-agent robotics. *Autonomous Robots*, 3:375–397, 1996.

Sven Gronauer and Klaus Diepold. Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 55(2):895–943, 2022.

Xudong Guo, Daming Shi, Junjie Yu, and Wenhui Fan. Heterogeneous multi-agent reinforcement learning for zero-shot scalable collaboration. *arXiv preprint arXiv:2404.03869*, 2024.

Tianyi Hu, Zhiqiang Pu, Xiaolin Ai, Tenghai Qiu, and Jianqiang Yi. Measuring policy distance for multi-agent reinforcement learning. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pages 834–842, 2024.

Jiechuan Jiang and Zongqing Lu. The emergence of individuality. In *International Conference on Machine Learning*, pages 4992–5001. PMLR, 2021.

Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pages 651–673. PMLR, 2018.

WOOJUN KIM and Youngchul Sung. Parameter sharing with network pruning for scalable multi-agent deep reinforcement learning. In *The 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. AAMAS, 2023.

Mykel J Kochenderfer, Tim A Wheeler, and Kyle H Wray. *Algorithms for decision making*. MIT press, 2022.

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)*, pages 6035–6039. IEEE, 2024a.

Xinran Li, Ling Pan, and Jun Zhang. Kaleidoscope: Learnable masks for heterogeneous multi-agent reinforcement learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b.

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

Manuel Lopez-Martin, Belen Carro, Antonio Sanchez-Esguevillas, and Jaime Lloret. Conditional variational autoencoder for prediction and feature recovery applied to intrusion detection in iot. *Sensors*, 17(9):1967, 2017.

Jonathan Lwowski, Patrick Benavidez, John J Prevost, and Mo Jamshidi. Task allocation using parallelized clustering and auctioning algorithms for heterogeneous robotic swarms operating on a cloud network. *Autonomy and artificial intelligence: A threat or savior?*, pages 47–69, 2017.

Clare Lyle, Zeyu Zheng, Khimya Khetarpal, James Martens, Hado P van Hasselt, Razvan Pascanu, and Will Dabney. Normalization and effective learning rates in reinforcement learning. *Advances in Neural Information Processing Systems*, 37:106440–106473, 2024.

Hao Ma, Tianyi Hu, Zhiqiang Pu, Liu Boyin, 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, 2024.

Dung Nguyen, Phuoc Nguyen, Svetha Venkatesh, and Truyen Tran. Learning to transfer role assignment across team sizes. *arXiv preprint arXiv:2204.12937*, 2022.

Zepeng Ning and Lihua Xie. A survey on multi-agent reinforcement learning and its application. *Journal of Automation and Intelligence*, 3(2):73–91, 2024.

Michael Noukhovitch, Samuel Lavoie, Florian Strub, and Aaron C Courville. Language model alignment with elastic reset. *Advances in Neural Information Processing Systems*, 36:3439–3461, 2023.

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*, 11:387–434, 2005.

Lynne E Parker. Lifelong adaptation in heterogeneous multi-robot teams: Response to continual variation in individual robot performance. *Autonomous Robots*, 8:239–267, 2000.

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

Esmaeil Seraj, Zheyuan Wang, Rohan Paleja, Matthew Sklar, Anirudh Patel, and Matthew Gombolay. Heterogeneous graph attention networks for learning diverse communication. *arXiv preprint arXiv:2108.09568*, 2021.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

Matthijs TJ Spaan. Partially observable markov decision processes. In *Reinforcement learning: State-of-the-art*, pages 387–414. Springer, 2012.

Lijun Sun, Yu-Cheng Chang, Chao Lyu, Ye Shi, Yuhui Shi, and Chin-Teng Lin. Toward multi-target self-organizing pursuit in a partially observable markov game. *Information Sciences*, 648:119475, 2023.

Leonid Nisonovich Vaserstein. Markov processes over denumerable products of spaces, describing large systems of automata. *Problemy Peredachi Informatsii*, 5(3):64–72, 1969.

T Wang, T Gupta, B Peng, A Mahajan, S Whiteson, and C Zhang. Rode: learning roles to decompose multi- agent tasks. In *Proceedings of the International Conference on Learning Representations*. OpenReview, 2021.

H Peyton Young. The evolution of conventions. *Econometrica: Journal of the Econometric Society*, pages 57–84, 1993.

Xiaoyang Yu, Youfang Lin, Xiangsen Wang, Sheng Han, and Kai Lv. Ghq: grouped hybrid q-learning for cooperative heterogeneous multi-agent reinforcement learning. *Complex & Intelligent Systems*, 10(4):5261–5280, 2024.

Chen Zhang, Riccardo Barbano, and Bangti Jin. Conditional variational autoencoder for learned image reconstruction. *Computation*, 9(11):114, 2021.

Ming Zhou, Jun Luo, Julian Villella, Yaodong Yang, David Rusu, Jiayu Miao, Weinan Zhang, Montgomery Alban, Iman Fadakar, Zheng Chen, et al. Smarts: An open-source scalable multi-agent rl training school for autonomous driving. In *Conference on robot learning*, pages 264–285. PMLR, 2021.

A Limitations

Although our proposed heterogeneity distance can effectively quantify agent heterogeneity and identify various potential heterogeneities, there remain some limitations in its practical implementation. One limitation is in scaling with the number of agents. Typically, the heterogeneity distance quantification algorithm outputs a heterogeneity distance matrix for the entire multi-agent system, with a computational complexity of $O(N^2)$. When the number of agents increases significantly, matrix computation becomes costly. However, if only studying heterogeneity between specific agents in the MAS is required, the method remains effective. One only needs to remove data from other agents during CVAE training and sampling computation.

Additionally, the practical algorithms for heterogeneity quantification are built on the assumption that agent-related variables are vectors. If certain agent variables, such as observation inputs, are multimodal, operations like padding in the proposed algorithm become difficult to implement. But this does not affect the correctness of the theory. As the relevant theory still holds in this situation, additional tricks are needed for practical calculation implementation.

B Broader Impacts

Our work systematically analyzes heterogeneity in MARL, which has strong correlations with a series of works in MARL. Under our theoretical framework, research on agent policy diversity in MARL can be categorized within the domain of policy heterogeneity. Our work can give a new perspective for studying policy diversity. Our proposed quantification methods can not only help these works with policy evolution analysis but also explain the relationship between policy diversity and agent heterogeneity. Furthermore, our proposed HetDPS, as an application case, can also be classified among parameter sharing-based works.

Additionally, some traditional heterogeneous MARL works can be categorized within environment-related heterogeneity domains. Our quantification and definition methods are orthogonal to these works, which can fully utilize our proposed methodology for further advancement. For instance, observation heterogeneity quantification can be used to enhance agents' ability to aggregate heterogeneous observation information; transition heterogeneity quantification can help design intrinsic rewards to assist heterogeneous multi-agents in learning cooperative policies.

In conclusion, our work not only expands the scope of heterogeneity in MARL but also closely connects with many current hot topics, contributing to the further development of these works.

C An introduction to POMG

Partially Observable Markov Game (POMG) is essentially an extension of Partially Observable Markov Decision Process (POMDP), which in turn extends Markov Decision Process (MDP). MDP [Bellman, 1957, Kaelbling et al., 1996] is a mathematical framework that describes sequential decision-making by a single agent in a fully observable environment. In an MDP, the agent can fully observe the environment's state, select actions based on the current state, and aim to maximize cumulative rewards. Compared to MDP, the key extension of POMDP [Kaelbling et al., 1998, Cassandra, 1998] is the consideration of partial observability, making it suitable for modeling both single-agent partially observable problems [Spaan, 2012] and multi-agent problems [Bernstein et al., 2002, Oliehoek et al., 2016]. In multi-agent POMDPs, agents typically operate in a fully cooperative mode, where their rewards are usually team-shared.

The key extension of POMG over POMDP lies in modeling mixed game relationships among multiple agents. Unlike POMDP, agents in POMG do not share a common reward function; instead, each agent has its own (agent-level) reward function, making POMG more general [Sun et al., 2023, Gronauer and Diepold, 2022]. This design enables POMG to handle competitive, cooperative, and mixed interaction scenarios, better reflecting the complexity of real-world multi-agent systems. The logical relationships among Markov decision processes and their variants are illustrated in Figure 7 and Figure 8. As shown in these figures, POMG is the most general framework for modeling original problems in the MARL domain. For these reasons, we chose POMG as the foundation for discussing heterogeneity in MARL.

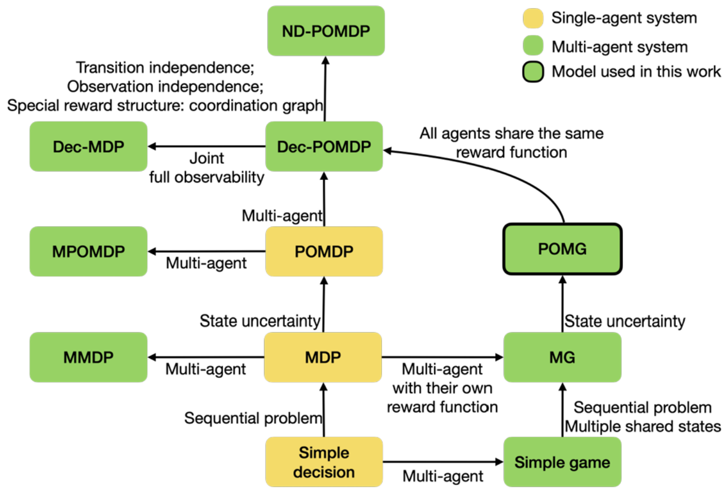

Figure 7: Common multi-agent problem formulations [Kochenderfer et al., 2022].

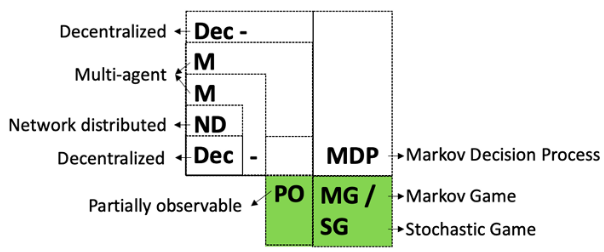

Figure 8: Common nomenclature for multi-agent models [Sun et al., 2023].

D Other potential types of heterogeneity in MARL

Benefiting from the reinforcement learning modeling based on POMG, we have clearly defined the boundaries of heterogeneity discussed in this paper. In fact, within the realm of unconventional multi-agent systems, there might be other types of heterogeneity.

For instance, agents may have different length of decision timesteps, with some agents inclined towards long-term high-level decisions, while others tend to make short-term low-level decisions. Agents may also have different discount factors, some works try to assign varying discount factors to different agents during algorithm training [Nguyen et al., 2022], to encourage agents to develop "*myopic*" or "*far-sighted*" policy behaviors, thereby promoting agent cooperation. However, differences in discount factors are more reflective of algorithmic design variations rather than environmental distinctions, and thus fall outside the scope of this paper. Moreover, there may be heterogeneity among agents regarding communication, agents might have different communication channels due to hardware variations. However, the establishment of communication protocols aims to enable agents to receive more information when making decisions, potentially overcoming non-stationarity and partial observability issues [Gronauer and Diepold, 2022]. These communication messages are essentially mappings of global information processed in the environment, which are then input into the action-related network modules. From this perspective, agent communication can be modeled as a more generalized observation function that maps global information to local observations for agent decision-making, and communication heterogeneity can be categorized under observation heterogeneity. From a learning perspective, agents might also have heterogeneous available knowledge, such as differences in initial basic policies or variations in supplementary knowledge accessible during execution phase. Moreover, heterogeneity might extend beyond abstract issues, including computational resource differences among agents during learning.

Overall, even from the perspective of multi-agent reinforcement learning, heterogeneity in multi-agent systems remains a domain with extensive discussion space, warranting further subsequent research.

E Properties of heterogeneity distance

Recap. The heterogeneity distance between two agents in Section 4 can be computed as follows:

$$d_{ij}^F = \int_{x \in X} D[F_i(\cdot|x), F_j(\cdot|x)] \cdot p(x)\, dx, \tag{6}$$

where X is the space of independent variables, $p(x)$ is the probability density function, and $D[\cdot,\cdot]$ is a measure that quantifies the difference between distributions.

Proposition 1. (*Properties of Heterogeneity Distance*) ① *Symmetry*: $d_{ij}^F = d_{ji}^F$; ② *Non-negativity*: $d_{ij}^F \geq 0$; ③ *Identity of indiscernibles*: $d_{ij}^F = 0$ if and only if agents i and j are F-homogeneous; ④ *Triangle inequality*: $d_{ij}^F \leq d_{ik}^F + d_{kj}^F$ $(i, j, k \in N)$. This proposition holds as long as the measure D satisfies Property ①②③④.

Proof. It can be proven that when D satisfies Property ①②③④, heterogeneity distance also satisfies Property ①②③④.

1) Proof of Symmetry:

$$d_{ij}^F = \int_{x \in X} D\left[F_i(\cdot|x), F_j(\cdot|x)\right] \cdot p(x)dx = \int_{x \in X} W\left[F_j(\cdot|x), F_j(\cdot|x)\right] \cdot p(x)dx = d_{ji}^F. \tag{7}$$

2) Proof of Non-negativity:

$$d_{ij}^F = \int_{x \in X} D\left[F_i(\cdot|x), F_j(\cdot|x)\right] \cdot p(x)dx \geq \int_{x \in X} 0 \cdot p(x)dx = 0. \tag{8}$$

3) Proof of Identicals of indiscernibility (necessary conditions):

if agent i and agent j are F-homogeneous, then we have: $X^{(i)} = X^{(j)}$, $\forall x \in X = X^{(i)}$, $F_i(\cdot|x) = F_j(\cdot|x)$,

$$\begin{aligned} d_{ij}^F &= \int_{x \in X} D\left[F_i(\cdot|x), F_j(\cdot|x)\right] \cdot p(x)dx \\ &= \int_{x \in X} D\left[F_i(\cdot|x), F_i(\cdot|x)\right] \cdot p(x)dx \\ &= \int_{x \in X} 0 \cdot p(x)dx \\ &= 0. \end{aligned} \tag{9}$$

4) Proof of Identicals of indiscernibility (sufficient conditions):

$$\begin{aligned} d_{ij}^F = 0 &\xrightarrow{\text{Prop.②}} D\left[F_i(\cdot|x), F_i(\cdot|x)\right] = 0, \forall x \in X^{(i)} or X^{(j)} \\ &\xrightarrow{\text{Prop.②of } D} F_i(\cdot|x) = F_i(\cdot|x), \forall x \in X, X = X^{(i)} = X^{(j)}, \end{aligned} \tag{10}$$

then we have agent i and agent j are F-homogeneous.

5) Proof of Triangle Inequality:

$$\begin{aligned} d_{ij}^F &= \int_{x \in X} D\left[F_i(\cdot|x), F_j(\cdot|x)\right] \cdot p(x)\, dx \\ &\leq \int_{x \in X} \left(D\left[F_i(\cdot|x), F_k(\cdot|x)\right] + D\left[F_k(\cdot|x), F_j(\cdot|x)\right]\right) \cdot p(x)\, dx \\ &= \int_{x \in X} D\left[F_i(\cdot|x), F_k(\cdot|x)\right] \cdot p(x)\, dx + \int_{x \in X} D\left[F_k(\cdot|x), F_j(\cdot|x)\right] \cdot p(x)\, dx \\ &= d_{ik}^F + d_{kj}^F. \end{aligned} \tag{11}$$

In this paper, we choose the *Wasserstein Distance* [Vaserstein, 1969] as the metric to quantify the distance between distributions, which satisfies the property ①②③④ [Bettini et al., 2023b].

Discussion. In practical computation, we adopt a representation learning-based approach to find an alternative latent variable distribution $p_i(z|x)$ to replace the original distribution $F_i(y|x)$ for quantification. It can be easily proved that when using latent variable distributions to compute heterogeneous distances, these distances still satisfy properties ①, ②, and ④ (following the same proof method as above).

In the model-based case, $p_i(z|x) = f_\phi(y_i, x)$, where f_ϕ represents the encoder of the CVAE. When two agents have the same independent and dependent variables (identical agent functions), their latent variable distributions are also identical. In this case, it is straightforward to prove that property ③ still holds under the model-based case.

In the model-free case, $p_i(z|x) = f_\phi(i, x)$. Due to the lack of an environment model, even agents with identical mappings may learn different representation distributions through their encoders, thus not satisfying property ③. However, as demonstrated in Section 4.2, although we cannot strictly determine agent homogeneity using $d_{ij}^F = 0$, the heterogeneity distances measured between homogeneous agents in the model-free case are sufficiently small. Moreover, the model-free manner is adequate to distinguish between homogeneous and heterogeneous agents, and still maintains the ability to quantify the degree of heterogeneity (as shown in Sections 4.2 and 6).

F More details of computing heterogeneity distance

Here, we present five formulas for calculating heterogeneity distances, corresponding to the five types of heterogeneity discussed in this paper.

Regarding **observation heterogeneity**, its relevant elements include the agent's observation space and observation function. For two agents i and j, let their observation heterogeneity distance be denoted as d_{ij}^Ω. The corresponding calculation formula is:

$$d_{ij}^\Omega = \int_{\hat{s}\in\{S^i\}_{i\in N}} D\left[\Omega_i(\cdot|\hat{s}), \Omega_j(\cdot|\hat{s})\right] \cdot p(\hat{s})\, d\hat{s}, \tag{12}$$

where $D[\cdot,\cdot]$ represents a measure of distance between two distributions, and $p(\cdot)$ is the probability density function (this notation applies to subsequent equations). Here, $\hat{s}$ denotes the global state, $\{S^i\}_{i\in N}$ represents the global state space, and Ω_i and Ω_j are the observation functions of agents i and j, respectively.

Regarding **response transition heterogeneity**, its relevant elements include the agent's action space, state space, and global state transition function. For two agents i and j, let their response transition heterogeneity distance be denoted as $d_{ij}^{\mathcal{T}}$. The corresponding calculation formula is:

$$d_{ij}^{\mathcal{T}} = \int_{\hat{s}\in\{S^i\}_{i\in N}} \int_{\hat{a}\in\{A^i\}_{i\in N}} D\left[\mathcal{T}^i(\cdot|\hat{s},\hat{a}), \mathcal{T}^j(\cdot|\hat{s},\hat{a})\right] \cdot p(\hat{s},\hat{a})\, d\hat{a}d\hat{s}, \tag{13}$$

where $p(\cdot,\cdot)$ represents the joint probability density function. $\hat{s}$ and $\hat{a}$ denote the global state and global action respectively, $\{S^i\}_{i\in N}$ and $\{A^i\}_{i\in N}$ represent the global state space and global action space, and $\mathcal{T}_i$ and $\mathcal{T}_j$ are the local state transition functions of agents i and j, respectively.

Regarding **effect transition heterogeneity**, its relevant elements include the agent's action space, state space, and global state transition function. For convenience, we denote $S^{-i} = \times_{k\in N,k\neq i} S^k \times S^E$ as the joint state space of all agents except agent i, $A^{-i} = \times_{k\in N,k\neq i} A^k$ as the joint action space of all agents except agent i, and $\mathcal{T}^{-i}$ as the collection of state transitions excluding agent i. For two agents i and j, let their effect transition heterogeneity distance be denoted as $d_{ij}^{\mathcal{T}^-}$. The corresponding calculation formula is:

$$d_{ij}^{\mathcal{T}^-} = \int_{s'\in S^{(-i)}} \int_{s\in A^i} \int_{a'\in A^{(-i)}} \int_{a\in A^i} D\left[\mathcal{T}^{-i}(\cdot|x), \mathcal{T}^{-j}(\cdot|x)\right] \cdot p(x) da da' ds ds', \tag{14}$$

where for convenience, we denote $x = (s', s, a', a)$, and p is the joint probability density function.

The calculation of effect transition heterogeneity distance differs from the previous two types of heterogeneity distances in two significant ways. The first difference lies in its introduction of agent-level elements as variables rather than global variables. When two agents have different agent-level variable spaces, it becomes challenging to calculate the heterogeneity distance under this definition. The second difference is that it involves a quadruple integral, making its computational complexity much higher than the single or double integrals of the previous two distances.

These two differences make the calculation of effect transition heterogeneity distance more challenging. Fortunately, through our proposed meta-transition model, we can simplify the calculation of effect transition heterogeneity distance to a double integral that only involves the agent's local states and actions. Additionally, the distance measurement through representation learning also reduces the constraints on the similarity of agents' variable spaces. Even when two agents have different variable spaces (for example, one agent's local state space is 10-dimensional while another's is 20-dimensional), we can still process the variable inputs through techniques like padding and then map them to the same dimension using encoder networks. This demonstrates that the approach based on representation learning and meta-transition significantly extends the applicability of heterogeneity distance measurement, which also holds true in the quantification of heterogeneous types discussed below.

Regarding **objective heterogeneity**, its relevant element is the agent's reward function. For two agents i and j, let their objective heterogeneity distance be denoted as d^r_{ij}. The corresponding calculation formula is:

$$d^r_{ij} = \int_{\hat{s}\in\{S^i\}_{i\in N}} \int_{\hat{a}\in\{A^i\}_{i\in N}} D\left[r^i(\cdot|\hat{s},\hat{a}), r^j(\cdot|\hat{s},\hat{a})\right] \cdot p(\hat{s},\hat{a})\, d\hat{a} d\hat{s}, \tag{15}$$

where $p(\cdot,\cdot)$ represents the joint probability density function. $\hat{s}$ and $\hat{a}$ denote the global state and global action respectively, $\{S^i\}_{i\in N}$ and $\{A^i\}_{i\in N}$ represent the global state space and global action space, and r_i and r_j are the reward functions of agents i and j, respectively.

Regarding **policy heterogeneity distance**, its relevant elements include the agent's observation space, action space, and policy function. For two agents i and j, let their policy heterogeneous distance be denoted as d^π_{ij}. The corresponding calculation formula is:

$$d^\pi_{ij} = \int_{o\in O^i} D\left[\pi_i(\cdot|o), \pi_j(\cdot|o)\right] \cdot p(o)\, do, \tag{16}$$

where $D[\cdot,\cdot]$ represents a measure of distance between two distributions, and $p(\cdot)$ is the probability density function. Here, o denotes the observation, O^i represents the observation space, and π_i and π_j are the policy functions of agents i and j, respectively.

G Meta-Transition and its Heterogeneity Distance

To quantify an agent's comprehensive heterogeneity, we introduce the concept of meta-transition. Meta-transition is a modeling approach that explores an agent's own attributes from its perspective. Our goal is to quantify an agent's comprehensive heterogeneity using only the agent's local information (as global information is typically difficult to obtain in practical MARL scenarios).

Based on this, we provide the definition of meta-transition. Let the meta-transition of agent i be denoted as M_i. It is a mapping $M_i : S_i \times A_i \to S_i \times R \times \Omega_i$. At time step t, the inputs of meta-transition are the agent's local state s^i_t and local action a^i_t, and the outputs are the next time step's local state s^i_{t+1}, the next time step's local observation o^i_{t+1}, and the current time step's reward r^i_t based on the state and action.

We explain why the above relationship can reflect all agent-level elements in POMG. The input local state and local action of meta-transition actually correspond to the inverse mapping to the global state and global action. This inverse mapping potentially restores the local state and action to global information, and then obtains the next time step's global state according to the global state transition function, which is mapped to local observation through the observation function. Therefore, this process reflects the agent's effect transition heterogeneity and observation heterogeneity. Additionally, the potential global state and global action also determine the agent's local state and corresponding reward at the next time step, which reflect the agent's response transition heterogeneity and objective heterogeneity, respectively.

It is worth noting that meta-transition is not a function that actually exists in POMG, but an implicitly defined mapping. We aim to quantify this mapping difference to capture the agent's comprehensive heterogeneity. Therefore, meta-transition heterogeneity is quantified in a model-free manner.

Moreover, meta-transition is not limited to the aforementioned form. It can be transformed into different forms according to the modular settings of independent and dependent variables. For example, by removing the agent's reward, meta-transition can reflect the agent's observation heterogeneity, response transition heterogeneity, and effect transition heterogeneity.

After determining the input and output of meta-transition, the relevant heterogeneity distance can be calculated using the same model-free method as before. Since meta-transition involves multiple variables, and the dimensions between these variables may differ significantly (for example, the dimension of reward is 1, while the dimension of observation might be 100), directly fitting with deep networks may struggle to capture information corresponding to low-dimensional variables. We address this issue through a dimension replication trick. In practice, we typically replicate the reward dimension to be similar to the dimensions of observation or action, ensuring that the autoencoder network can capture information related to objective heterogeneity during learning.

H Derivation of ELBO

The Evidence Lower Bound (ELBO) of the likelihood can be derived as follows:

$$
\begin{aligned}
\log p(y|x) &= \log \int p(y,z|x)dz && \text{(a)} \\
&= \log \int \frac{p(y,z|x) f_\phi(z|y,x)}{f_\phi(z|y,x)} dz && \text{(b)} \\
&= \log \mathbb{E}_{f_\phi(z|y,x)} \left[\frac{p(y,z|x)}{f_\phi(z|y,x)} \right] && \text{(c)} \\
&\geq \mathbb{E}_{f_\phi(z|y,x)} \left[\log \frac{p(y,z|x)}{f_\phi(z|y,x)} \right] && \text{(d)} \\
&= ELBO_{\text{model-based}},
\end{aligned}
\tag{17}
$$

where $f_\phi(z|y,x)$ represents the posterior probability distribution of the latent variable generated by the encoder, and $p(y,z|x)$ denotes a joint probability distribution concerning the customized feature and latent variable, conditioned on o. Throughout the derivation of the formula, (a) employs the properties of the joint probability distribution, (b) multiplies both numerator and denominator by $f_\phi(z|y,x)$, (c) applies the definition of mathematical expectation, and (d) invokes the Jensen's inequality.

Considering that the ELBO includes an unknown joint probability distribution, we can further decompose it by using the posterior probability distributions from the encoder and decoder:

$$
\begin{aligned}
ELBO_{\text{model-based}} &= \mathbb{E}_{f_\phi(z|y,x)} \left[\log \frac{p(y,z|x)}{f_\phi(z|y,x)} \right] \\
&= \mathbb{E}_{f_\phi(z|y,x)} \left[\log \frac{g_\omega(c|z,x) p(z|x)}{f_\phi(z|y,x)} \right] && \text{(a)} \\
&= \mathbb{E}_{f_\phi(z|y,x)} \left[\log g_\omega(c|z,x) \right] \\
&\quad + \mathbb{E}_{f_\phi(z|y,x)} \left[\log \frac{p(z|x)}{f_\phi(z|y,x)} \right] && \text{(b)} \\
&= \underbrace{\mathbb{E}_{f_\phi(z|y,x)} \left[\log g_\omega(c|z,x) \right]}_{\text{reconstruction term}} - \underbrace{D_{\text{KL}} \left[f_\phi(z|y,x) \| p(z|x) \right]}_{\text{prior matching term}}, && \text{(c)}
\end{aligned}
\tag{18}
$$

where $f_\phi(z|y,x)$ and $g_\omega(c|z,x)$ are the posteriors from the encoder and decoder, respectively. The conditional joint probability distribution $p(y,z|x)$ is a imaginary construct in mathematical terms and lacks practical significance. It can be formulated using the probability chain rule, constructed from the posterior distribution of the customized feature and the prior distribution of the latent variable (step (a)). Step (b) decomposes the expectation, and step (c) applies the definition of the KL divergence.

Thus, the ELBO can be decomposed into a reconstruction term of the customized feature, and a prior matching term of the posterior and the prior. By maximizing the ELBO, the reconstruction likelihood can be maximized while minimizing the KL divergence between the posterior and the prior. In the model-free case, the same approach can be used to derive the ELBO and corresponding loss function.

I Details of HetDPS

HetDPS is a novel algorithm designed to efficiently manage the allocation of neural network parameters across multiple agents in MARL. This algorithm leverages the Wasserstein distance matrix to cluster agents based on their similarities, and subsequently assigns them to suitable neural networks. The pseudocode of HetDPS is shown in Algorithm 1.

The algorithm begins by computing the affinity matrix from the Wasserstein distance matrix, which is then used as input to the Affinity Propagation clustering algorithm. This process yields a new set of cluster assignments for the agents. If it is the first time the algorithm is executed, the cluster assignments are directly used as network assignments.

In subsequent iterations, the algorithm compares the new cluster assignments with the previous ones to determine the optimal network assignments. This is achieved by constructing an overlap matrix that captures the similarity between the old and new cluster assignments. Based on the number of old and new clusters, the algorithm handles three distinct cases:

1. Equal number of old and new clusters: In this scenario, the algorithm establishes a one-to-one mapping between the old and new clusters using the Hungarian algorithm. It then constructs a mapping from old clusters to networks and assigns each agent to a network based on its new cluster assignment.

2. More new clusters than old clusters: When the number of new clusters exceeds the number of old clusters, the algorithm handles network splitting. It uses the Hungarian algorithm to find the best matching between old and new clusters and establishes a mapping from new clusters to old clusters. For new clusters without a clear match, the algorithm either finds the most similar old cluster or identifies the closest network. It then executes a splitting operation to copy parameters from the source network to the new network.

3. More old clusters than new clusters: In this case, the algorithm handles network merging. It uses the Hungarian algorithm to find the best matching between old and new clusters and establishes a mapping from old clusters to new clusters. For each new cluster, it identifies the networks to be merged and executes a merging operation based on the specified merge mode (majority, random, average, or weighted). The algorithm then assigns each agent to a network based on its new cluster assignment.

HetDPS offers a flexible and efficient approach to managing neural network parameters in multi-agent systems. By dynamically adjusting network assignments based on agent similarities, the algorithm enables effective parameter sharing and reduces the need for redundant computations.

Algorithm 1 HetDPS

```
Initialize policies and parameter sharing paradigm
for episode = 1 to maxEpisodes do
    Interact with environment to collect data
    Add data to reinforcement learning (RL) sample pool
    Add data to heterogeneity distance sample pool
    if episode % trainingPeriod = 0 then
        Update policies using RL sample pool
    end if
    if episode % quantizationPeriod = 0 then
        Compute heterogeneity distance matrix D (Section 4)
        Cluster agents using Affinity Propagation on D
        if no previous clustering exists then
            Assign networks to agents based on clusters
            Copy network parameters as needed
        else
            Compute maximum overlap matching between current and previous clusters
            if number of clusters unchanged then
                Map new clusters to previous networks
            else if new clusters > previous clusters then
                Split networks: copy parameters for unmatched clusters
            else
                Merge networks: combine parameters based on merge mode
            end if
            Assign networks to agents
        end if
    end if
end for
```

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Both the abstract and introduction clearly state that our main contribution and scope: this work systematically establishes a theoretical framework for heterogeneity in multi-agent reinforcement learning, advancing both theoretical development and practical applications in this field.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in Section A.

   Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The study meticulously enumerates all underlying assumptions and furnishes a comprehensive and mathematically rigorous proof for each theoretical result.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper includes detailed descriptions of the experimental setup, algorithms, model architectures in Section 6.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have provided open access to both the data and code necessary to reproduce our main experimental results.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All experimental settings are clearly stated in Section 6.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All training curves in Section 6 are plotted with the mean $\pm$ std across two random seeds.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All necessary information are provided in Section 6 .

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Our research fully adheres to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Our paper thoroughly discusses both the potential positive and negative societal impacts. For details, please refer to Appendix B.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper does not involve the release of data or models.

    Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All existing assets used in the paper, including code, data, and models, have been properly credited to their original creators.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release any new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The paper does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.