Offline Multi-Agent Preference-Based Reinforcement Learning with Agent-aware Direct Preference Optimization

Qian Kou[1]
Xi'an Jiaotong University[2]
Xi'an, China
xjtukouqian@stu.xjtu.edu.cn

Mingyang Li[1]
Xi'an Jiaotong University[2]
Xi'an, China
limingyang@stu.xjtu.edu.cn

Zeyang liu[1]
Xi'an Jiaotong University[2]
Xi'an, China
zeyang.liu@xjtu.edu.cn

Long Qian
Xi'an Jiaotong University[2]
Xi'an, China
qianlongym@stu.xjtu.edu.cn

Zhuoran Chen
Xi'an Jiaotong University[2]
Xi'an, China
zhuoran.chen@xjtu.edu.cn

Lipeng Wan
Xi'an Jiaotong University[2]
Xi'an, China
wanlipeng77@xjtu.edu.cn

Xingyu Chen[3]
Xi'an Jiaotong University[2]
Xi'an, China
chenxingyu_1990@xjtu.edu.cn

Xuguang Lan[3]
Xi'an Jiaotong University[2]
Xi'an, China
xglan@mail.xjtu.edu.cn

ABSTRACT

Multi-agent Preference-Based Reinforcement Learning (MAPbRL) is promising in offline policy learning by leveraging human preferences to replace complex manual reward designing. Current MAPbRL methods use complicated structures to realize better reward modeling with off-the-shelf MARL algorithms and obtain the joint policy based on it. However, it faces a severe preference-behavior mismatch problem stemming from the instability of RL training and global-local preference inconsistency datasets in offline MARL, resulting in potential suboptimal policy convergence. To address this problem, we propose Agent-aware Multi-Agent Direct Preference Optimization (AMADPO) by utilizing a multi-agent preference predictor to guide agent-aware direct optimization from imbalanced preference labels, which can learn coordination policy from both positive and negative segments. Experimental results in SMAC environment show substantial improvements in global-local preference inconsistency datasets, demonstrating the effectiveness of AMADPO in solving the preference-behavior mismatch problem.

KEYWORDS

Multi-agent reinforcement learning, Preference-based reinforcement learning, Offline reinforcement learning

ACM Reference Format:
Qian Kou[1], Mingyang Li[1], Zeyang liu[1], Long Qian, Zhuoran Chen, Lipeng Wan, Xingyu Chen[3], and Xuguang Lan[3]. 2025. Offline Multi-Agent Preference-Based Reinforcement Learning with Agent-aware Direct Preference Optimization. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 10 pages.

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), A. El Fallah Seghrouchni, Y. Vorobeychik, S. Das, A. Nowe (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

1 INTRODUCTION

Recent advances in Reinforcement Learning (RL) have shown strong performance in solving complex decision-making problems across various domains [3, 28, 31, 35, 40]. However, it can be very challenging to construct an efficient and suitable reward function for specific tasks, which costs huge human expert efforts. To solve this problem, offline Preference-based Reinforcement Learning (PbRL) enables agents to learn from preference labels between trajectory segments [7, 21, 30] through a fixed offline preference dataset, avoiding unsafe physical interactions with the environment[37]. Consequently, agents' behaviors can be aligned with human desires through relative comparison over pairs of segments, which is much easier available than directly providing rewards. Recent works in offline PbRL have showcased its effectiveness in addressing single-agent RL tasks across various domains[6, 18].

However, it is difficult to directly extend PbRL into multi-agent RL because the cooperative tasks require a fine-grained reward design. MAPT [49], building on previous work [18], uses transformers to decode global preference in both agent-wise and temporal-wise, achieving a more effective reward structure for multi-agent cooperation. Meanwhile, DPM [16] leverages the Large Language Model (GPT-4o [1]) to provide additional rank preferences across agents' actions at every timestep within a single trajectory segment. Unfortunately, these methods face a crucial challenge: the *preference-behavior mismatch*, i.e., the learned behavior of the agents may not align with the human preference.

This mismatch stems from: *First*, the global-local preference inconsistency, where the mis-coordinated agents may hinder the learning of good agents when the joint trajectory is less preferred from the perspective of the global view. For example, in a multi-robot coordination task, each human operator has a different level of skills, resulting in global-local preference inconsistency datasets. As

[1]Both authors contributed equally to this research.
[2]National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Application Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China.
[3]Corresponding authors.

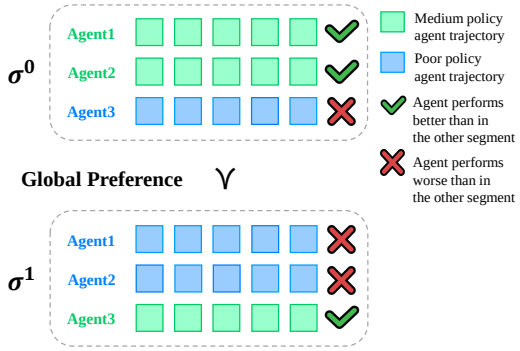


Figure 1: An example of global-local preference inconsistency preference datasets. Joint multi-agent trajectory Segment 0 is preferred over Segment 1 according to global preference while Agent3 actually performs better in Segment 1.

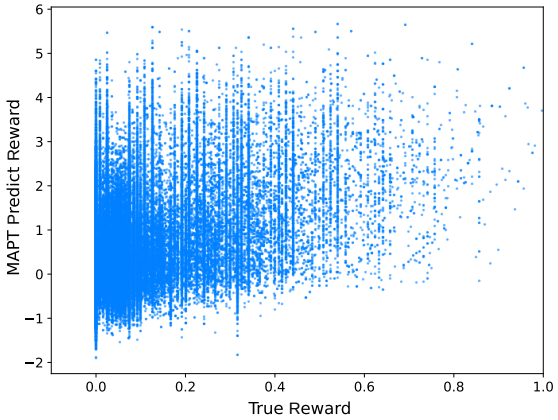


Figure 2: Predict reward and true reward in task MMM2 of SMAC environment using MAPT as reward model.

illustrated in Figure 1, although σ_0 is preferred over σ_1 from a global perspective, several agent trajectories in σ_1 still outperform those in σ_0. If we only rely on global preferences, well-performed agents' policies in non-preferred segments cannot be utilized, thus decreasing sample efficiency and final performance. In addition, each agent trajectory in an offline multi-agent joint trajectory usually comes from different sources and has a different data quality [38]. *Second*, the learning instability in the two-stage training paradigm of PbRL, where the reward and the policy models are trained separately. As shown in Figure 2, we visualize the ground truth reward and predicted reward using MAPT, which models human preferences as a weighted summarization of agent-wise and temporal-wise rewards. It fails to reconstruct the underlying global rewards. Therefore, there arises an open research problem:

How to identify well-preformed agents between compared offline segments and construct a direct preference optimization method, solving the problem of preference-behavior mismatch and thus boosting multi-agent coordination?

To solve this challenge, we propose Agent-aware Multi-Agent Direct Preference Optimization (AMADPO). AMADPO involves training a multi-agent preference predictor and directly optimizing policy based on the provided preference datasets, where the good agents' behaviors from preferred and less-preferred trajectories are fully utilized. AMADPO begins with introducing additional preference labels over the same agents between compared trajectory segments. Along with the global preference labels, these labels help train the multi-agent preference predictor to identify preferred agent trajectories in both preferred and less preferred segments. Inspired by previous works [2, 12] in single agent PbRL, we design a multi-agent policy-related metric to replace the reward in the preference model, which increases the probability of preferred segments and decrease those closed to less preferred ones. Furthermore, we leverage the intermediate values of preference predictor to form agent-aware importance weights, which indicates the probability of an agent being preferred compared to others within the same segment. Finally, we enhance policy learning by combining weights and the proposed metric, allowing learning from good agents in both compared segments and distancing itself from less preferred agents. Our contributions are summarized as follows:

- We first analyze the problem of preference-behavior mismatch in the current multi-agent PbRL methods and eliminate this mismatch by bridging the relation between direct preference optimization and multi-agent PbRL.
- We present a novel paradigm named AMADPO which incorporates the generalization of preference predictor and the stability of direct preference optimization. Leveraging additional agent-wise preference labels, it can produce agent-aware importance weights and directly learn from good agents in both preferred and less preferred multi-agent joint trajectories.
- Experimental results showcase that our approach achieves better results in complex offline datasets in Starcraft Multi-agent Challenge [35], even outperforming offline MARL algorithms (i.e., ICQ [44]) with ground truth reward in some scenarios.

2 RELATED WORKS

In this section, we review the most relevant works from PbRL and offline MARL.

Online PbRL. Designing suitable reward metrics or collecting expert demonstrations can be costly in many real-world tasks. In contrast, human preferences over compared pairs of agent trajectories are much easier to obtain as the signals for policy optimization. Deep RL from human feedback (RLHF) [7] first utilizes Bradley-Terry model [5] to implicitly learn the reward function from human preferences and then use online deep RL algorithms to solve complex control tasks. Following this paradigm, several works improve training efficiency by leveraging techniques in query selection, data enhancement, pre-training, meta-learning, and more powerful RL algorithms [14, 15, 21, 22, 25, 30]. All these works require online interactions with environments and they all adopt a two-stage training paradigm that needs to obtain a reward function first. However, both online learning and two-stage PbRL have limitations. Online

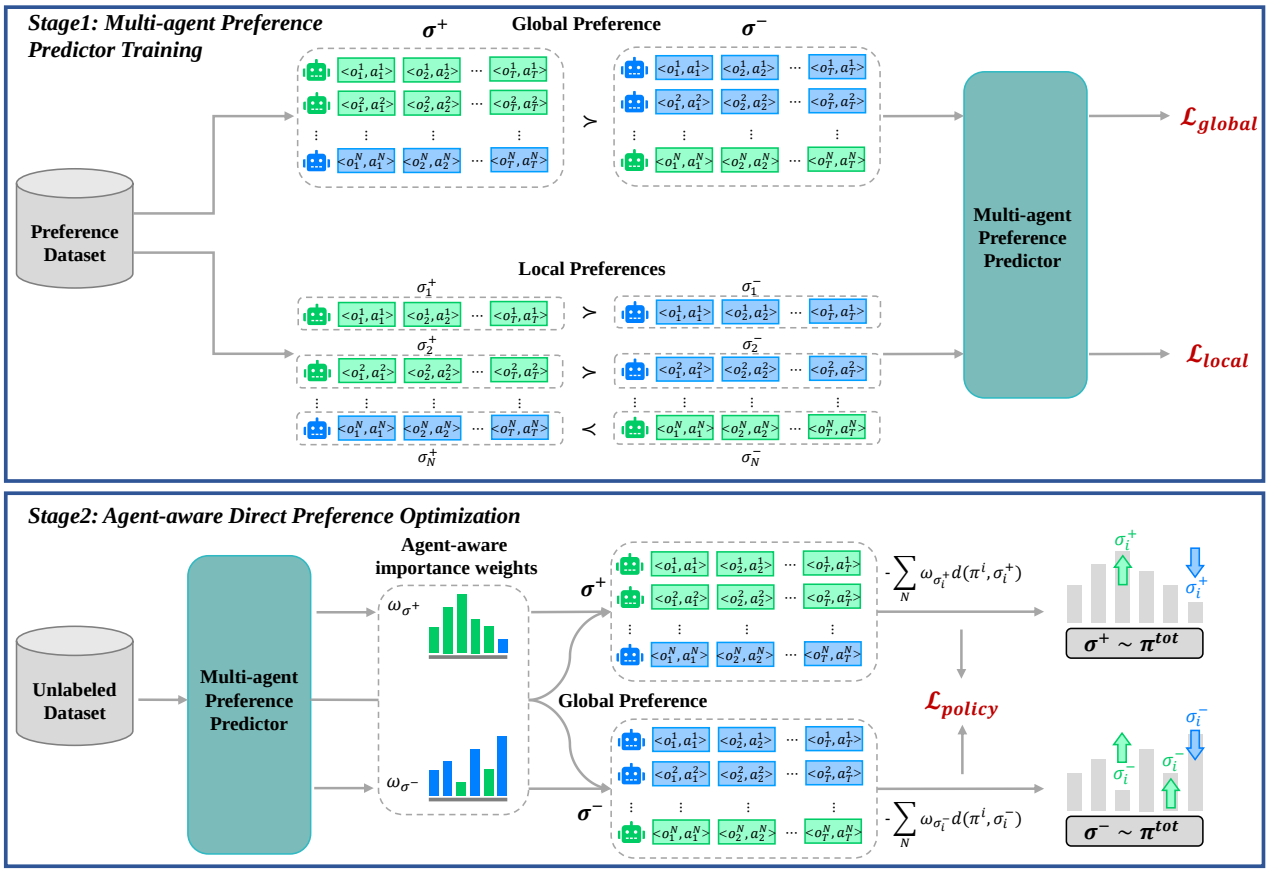


Figure 3: The overview of Agent-aware Multi-Agent Direct Preference Optimization.

RL suffers from expensive and dangerous data collection in real-world settings (e.g., autonomous driving) [24]. As for two-stage PbRL, scaler rewards may create information bottleneck in policy optimization, resulting in suboptimal policy[17, 39]. Besides, separately learning policy based on a reward function that may not have been adequately and correctly trained will lead to undesirable behaviors.

Offline PbRL. OPAL [36] first extends PbRL to offline manners using off-the-shelf PbRL methods thus avoiding unsafe real rollouts with environments. OPRL [37] further improves offline PbRL with pool-based active learning. Preference Transformer (PT) [18] introduces a transformer-based reward structure to model human preferences as a weighted sum of non-Markovian rewards [8]. Other works integrate hindsight information [10], list-wise preferences [6], and multimodal inputs [46] with offline PbRL. Despite these methods eliminating the need for online interactions, they still carry out a two-stage training strategy. On the other hand, led by Direct Preference Optimization (DPO) [32] in LLM post-training stage [47], some works bypass the need for reward model and RL by directly inducing policy from preferences datasets. OPPO [17] uses hindsight information matching in compact latent space for preference learning, while FTB [45] utilizes Diffusion Model to generate higher-preference trajectories. IPL [13], CPL [12] and OPPO [2] replace the reward in Bradley-Terry model with inverse soft-Bellman operator, optimal advantage function, and policy-segment distance respectively, all of which can directly optimize the policy by relating preferences to these policy-related metrics. However, most of them are restricted with exist preferences datasets and can not apply to new datasets without labels.

Offline MARL and MAPbRL. Recently, many works have attempted to apply offline RL in multi-agent settings. ICQ [44] first explores offline MARL using implicitly constrained policy learning. OMAR [29] combines first-order policy gradients and zeroth-order optimization methods to prevent falling into local optima. MADT [26] leverages transformer's ability for sequence modeling in offline pre-training. OMIGA [42] bridges multi-agent value decomposition and policy learning with offline regularizations by converting global-level value regularization into equivalent implicit local value regularizations. There are also multi-agent versions of single-agent offline RL algorithms, such as BCQ [9], CQL [20] and IQL [19]. However, these approaches need relatively high-quality offline datasets and elaborate reward signals. Although demonstrated in single-agent RL tasks, PbRL has been studied in a few works when it comes to multi-agent contexts. MAPT [49] designs a multi-agent version preference transformer reward model structure to decode the global preferences implicitly in both agent-wise and

temporal-wise, achieving better preference modeling. Yet it still has the disadvantages of two-stage PbRL. DPM [16] introduces additional local preferences as a format of rankings of agents' actions at each timestep within a joint trajectory, provided by GPT-4o [1]. Except for the two-stage strategy, DPM also requires detailed textual descriptions of every environment and a substantial amount of ranking preferences among agents in each timestep, which are very costly for both LLM and human annotators.

Relation to CPL and DPPO. CPL, DPPO, and our method are all direct preference optimization approaches. However, AMADPO is different from them in two ways. First, AMADPO aims to address the preference-behavior mismatch problem in multi-agent scenarios while CPL and DPPO focus on single-agent tasks. Second, AMADPO defines the policy-related metric as the divergence between policy and trajectory segment, which can leverage multi-form divergences to optimize the policy instead of single-form in CPL and DPPO. We implement the multi-agent version of CPL and DPPO as baselines.

3 PRELIMINARIES

3.1 Cooperative MARL

We focus on fully cooperative multi-agent tasks which can be formulated as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) [27]. A Dec-POMDP is characterized by the tuple $\langle S, A, P, R, O, \Omega, N, \gamma \rangle$, where $N = \{1, ..., n\}$ is the set of agents. $s \in S$ is the global environment state and A denotes the action set of n agents. At each time step, agent $i \in N$ obtains an observation $o_i \in O$ via the observation function $\Omega(s, i) : S \times N \rightarrow O$, and selects an action $a_i \in A$ which forms a joint action $\mathbf{a} = \{a_1, ..., a_n\} \in A^n$. $P(s'|s, a) : S \times A^n \rightarrow [0, 1]$ is the transition dynamics to the next state s'. $\gamma \in [0, 1)$ is the discount factor. Agents receive the same global reward function $r(\mathbf{o}, \mathbf{a}) : O \times A^n \rightarrow R$, where we denote $\mathbf{o} = (o_1, ...o_n)$, and aim to learn a set of policies $\boldsymbol{\pi}_{tot} = \{\pi_1, ..., \pi_n\}$ that jointly maximize the expected discounted returns $\mathbb{E}_{\mathbf{o},\mathbf{a}\sim\pi_{tot}}[\sum_{t=0}^{\infty} \gamma^t r(\mathbf{o}_t, \mathbf{a}_t)]$. In the offline setting, the policy is learned within a fixed dataset D sampled by the behavior policy $\boldsymbol{\mu}_{tot} = \{\mu_1, ..., \mu_n\}$ without any environment interactions.

3.2 Preference-based RL

Consistent with previous studies [7, 18, 21, 30, 49], we consider assigning preferences over pairs of multi-agent trajectory segments. A segment of length T can be formulated as $\boldsymbol{\sigma} = \{\sigma_1, \ldots, \sigma_n\}$, where $\sigma_j = \{o_{1,j}, a_{1,j}, \ldots, o_{T,j}, a_{T,j}\}$, $j \in N$. Given a segment pair $\{\boldsymbol{\sigma^0}, \boldsymbol{\sigma^1}\}$, preference labels are indicated by $y \in \{0, 1, 0.5\}$ where $y = 0$ and $y = 1$ denotes $\boldsymbol{\sigma^0}$ and $\boldsymbol{\sigma^1}$ is preferred over the other, respectively, and $y = 0.5$ for equally preferred.

We construct the preference predictor based on the Bradley-Terry model [5]:

$$P[\boldsymbol{\sigma^0} \succ \boldsymbol{\sigma^1}] = \frac{\exp \sum_{t=1}^{T} \hat{r}(\mathbf{o}_t^0, \mathbf{a}_t^0)}{\exp \sum_{t=1}^{T} \hat{r}(\mathbf{o}_t^0, \mathbf{a}_t^0) + \exp \sum_{t=1}^{T} \hat{r}(\mathbf{o}_t^1, \mathbf{a}_t^1)} \tag{1}$$

where $\boldsymbol{\sigma^0} \succ \boldsymbol{\sigma^1}$ indicates $\boldsymbol{\sigma^0}$ is preferred than $\boldsymbol{\sigma^1}$, and $\hat{r}$ is the reward function. Then given a preference dataset $\mathcal{D}_{\text{pref}} = \{(\boldsymbol{\sigma^0}, \boldsymbol{\sigma^1}, y)\}$, the reward function $\hat{r}$ can be obtained by minimizing the cross-entropy loss between predictor and preference labels:

$$\begin{aligned} \mathcal{L}_{CE} = -\mathbb{E}_{(\boldsymbol{\sigma^0},\boldsymbol{\sigma^1},y)\sim\mathcal{D}_{\text{pref}}} \Big[& (1-y) \log P\left(\boldsymbol{\sigma^0} \succ \boldsymbol{\sigma^1}\right) \\ & + y \log P\left(\boldsymbol{\sigma^1} \succ \boldsymbol{\sigma^0}\right) \Big] \end{aligned} \tag{2}$$

In this work, we focus on offline MARL accessing to a pre-collected preference dataset $\mathcal{D}_{\text{pref}}$ along with a massive unlabelled dataset $\mathcal{D}_{\text{unlabel}}$, both lacking explicit rewards[2, 10]. Preference dataset $\mathcal{D}_{\text{pref}}$ is utilized to train the reward function which is applied to label $\mathcal{D}_{\text{unlabel}}$. Then any offline MARL algorithm, such as ICQ [44], can be used for policy learning with the resulting dataset.

4 METHOD

This section presents Agent-aware Multi-Agent Direct Preference Optimization (AMADPO) to solve the preference-behavior mismatch problem in offline MAPbRL. AMADPO consists of three components: 1) **Multi-agent Preference Predictor**, which takes global and local labels as inputs and predicts global preferences with identifications of good agents, 2) **Multi-agent Preference Metric**, which defines a preference metric via multi-agent policy-segment divergence, 3) **Agent-aware Importance Weights**, which provide importance weights for each agent to rectify the learning gradients of the joint policy based on the preference metric and intermediate results from the Multi-agent Preference Predictor. Finally, AMADPO directly optimizes the joint policy by prioritizing good agents' behaviors in preferred and less preferred global trajectories.

4.1 Multi-agent Preference Predictor

Prior PbRL approaches often adopt a two-stage procedure including training a reward model from preference labels. These works assume that human preferences are distributed according to the sum or weighted sum of underlying rewards. Applying such rewards to offline MARL training may result in sub-optimal policy or undesirable behaviors. Meanwhile, using the reward model only as a preference predictor trained with supervised learning has no reward-preference gaps since we have access to exact labels.

First of all, we define the concept of Preference Score (PS) in place of reward to form the multi-agent preference predictor model. In concrete, we slightly modify Eq.1 as follows:

$$P_{MA}[\boldsymbol{\sigma^0} \succ \boldsymbol{\sigma^1}] = \frac{\exp \sum_{t=1}^{T} w_t^0 AGG\{\rho(o_{t,i}^0, a_{t,i}^0)\}_{i=1}^{N}}{\sum_{m\in\{0,1\}} \exp \sum_{t=1}^{T} w_t^m AGG\{\rho(o_{t,i}^m, a_{t,i}^m)\}_{i=1}^{N}} \tag{3}$$

where $\rho(o, a)$ represents the PS of an observation-action pair, AGG denotes an agent-wise aggregation operator and w_t is the temporal-wise weight adapting to different predictor structures. The aggregation operator is the average operation for MLP [7], LSTM [8] and PT [18] predictor and is a transformer for MAPT [49] predictor. w_t is trained in LSTM, PT, and MAPT predictors to consider sequential information. w_t of MLP predictor directly equals to $1/T$.

Moreover, only accessing global preferences between two joint multi-agent trajectory segments to train a multi-agent preference predictor is confused about which agents are better in global-local preference inconsistency situations.

To induce agent-aware importance weights for subsequent policy optimization, we introduce additional local preference labels and the agent preference model. Given a multi-agent trajectory segment $\{\boldsymbol{\sigma^0}, \boldsymbol{\sigma^1}\}$, local preferences are provided in following format:

$$y_i \coloneqq \begin{cases} 0 & \sigma_i^0 \succ \sigma_i^1 \\ 1 & \sigma_i^1 \succ \sigma_i^2, \quad i \in N \\ 0.5 & \sigma_i^0 = \sigma_i^1 \end{cases} \tag{4}$$

Agent preference model is similar to Eq.3:

$$P_A[\sigma_i^0 \succ \sigma_i^1] = \frac{\exp \sum_{t=0}^{T-1} w_t^0 \rho(o_{t,i}^0, a_{t,i}^0)}{\sum_{m\in\{0,1\}} \exp \sum_{t=0}^{T-1} w_t^m \rho(o_{t,i}^m, a_{t,i}^m)} \tag{5}$$

Compared to DPM [16], our local preference labels only compare the same agent's whole trajectory between segments while DPM needs $2T \times \binom{N}{2}$ local labels for every segment pair which can be extremely numerous as the number of agents grows. With additional local preference labels, we can train a multi-agent preference predictor with any structure using the following loss function:

$$\begin{aligned} \mathcal{L}_{global} = -\mathbb{E}_{(\boldsymbol{\sigma^0},\boldsymbol{\sigma^1},y)\sim\mathcal{D}_{\text{pref}}} \Big[& (1-y)\log P_{MA}\left(\boldsymbol{\sigma^0} \succ \boldsymbol{\sigma^1}\right) \\ & + y \log P_{MA}\left(\boldsymbol{\sigma^1} \succ \boldsymbol{\sigma^0}\right)\Big] \end{aligned} \tag{6}$$

$$\begin{aligned} \mathcal{L}_{local} = -\mathbb{E}_{(\boldsymbol{\sigma^0},\boldsymbol{\sigma^1},\{y_i\})\sim\mathcal{D}_{\text{pref}}} \Bigg[& \sum_{i\in N} (1-y_i)\log P_A\left(\sigma_i^0 \succ \sigma_i^1\right) \\ & + y_i \log P_A\left(\sigma_i^1 \succ \sigma_i^0\right)\Bigg] \end{aligned} \tag{7}$$

$$\mathcal{L}_{pref} = \mathcal{L}_{global} + \lambda_l \mathcal{L}_{local} \tag{8}$$

where $\lambda_l \in [0,1]$ controls the impacts of local preferences.

After training on a small pre-collected preference dataset D_{pref} that consists of quadruple $(\boldsymbol{\sigma^0}, \boldsymbol{\sigma^1}, y, \{y_i\}_{i=1}^n)$, the multi-agent preference predictor can be used to label new datasets with global preferences and produce agent-aware importance weights.

4.2 Multi-agent Preference Metric

A simple yet efficient way in DPO methods is to replace the reward in preference model with policy-related metrics and directly optimize policy from preferences. We design a multi-agent policy-related preference metric as the divergence between joint policy and multi-agent joint trajectory segments, which is the average sum of the divergence between agent policy and each single transition:

$$d(\pi_i, \sigma_i) = \frac{1}{T}\sum_{t=0}^{T-1} d_{oa}(\pi_i, o_{t,i}, a_{t,i}) \tag{9}$$

$$d(\boldsymbol{\pi_{tot}}, \boldsymbol{\sigma}) = \frac{1}{N}\sum_{i=1}^{N} d(\pi_i, \sigma_i) \tag{10}$$

where d_{oa} denotes the distance between agent policy and single-agent divergence. The specific form of d_{oa} can be varied, so we chose two simple yet meaningful divergences. One is the expected $L2$ distance between the agent action and the trajectory action:

$$d_{action}(\pi_i, o_{t,i}, a_{t,i}) = \mathbb{E}_{\hat{a}_i\sim\pi_i(\cdot|o_{t,i})}\left[\|a_{t,i} - \hat{a}_i\|_2\right] \tag{11}$$

The other is to directly measure the policy-transition divergence using the KL-constrained term used in offline RL objective [11, 23]:

$$d_{policy}(\pi_i, o_{ti}, a_{ti}) = -\alpha \log \frac{\pi_i(o_{t,i}, a_{t,i})}{\mu_i(o_{t,i}, a_{t,i})} \tag{12}$$

where α is a temperature parameter and μ^i is behavior policy of dataset.

Replacing reward with policy-segment divergence needs a slight alteration. Since the preferred segments should have higher rewards, we use the negative version of policy-segment distance to assign higher values for policies closed to the preferred segments. Now we can directly optimize multi-agent policy by combining policy-segment divergence with BT model and cross-entropy loss:

$$\begin{aligned} &\mathcal{L}_{policy} = \\ &-\mathbb{E}_{(\boldsymbol{\sigma^+},\boldsymbol{\sigma^-})\sim\mathcal{D}_{\text{pref}}}\left[\log \frac{\exp\left(-d(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^+})\right)}{\exp\left(-d(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^+})\right)+\exp\left(-d(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^-})\right)}\right] \end{aligned} \tag{13}$$

where $\boldsymbol{\sigma^+}, \boldsymbol{\sigma^-}$ represent the preferred and less preferred segment for convenience (also referred as the positive and negative segments). Following previous works [2, 12, 13], we also add a regular parameter $\lambda_p \in [0,1]$ to penal the policy deviating from the preferred trajectory segment:

$$\begin{aligned} &\mathcal{L}_{policy} = \\ &-\mathbb{E}_{(\boldsymbol{\sigma^+},\boldsymbol{\sigma^-})\sim\mathcal{D}_{\text{pref}}}\left[\log \frac{\exp\left(-d(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^+})\right)}{\exp\left(-d(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^+})\right)+\lambda_p \exp\left(-d(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^-})\right)}\right] \end{aligned} \tag{14}$$

4.3 Agent-aware Importance Weights

We can directly optimize the policy through the loss function in Eq.14 with preference labels provided by the preference predictor on unlabeled dataset $\mathcal{D}_{\text{unlabel}}$. However, this objective equally encourages each agent's policy to approach the corresponding agent trajectory in the global preferred joint segment, which leads to suboptimal policy in global-local preference inconsistency datasets for some agents may align with uneven behaviors. Similar to those MARL algorithms that work on the multi-agent credit assignment problem [33, 34, 41, 43], we make use of trained multi-agent preference predictor to produce Agent-aware importance weights to highlight good agents contribution in both compared segments.

After trained on preference dataset $\mathcal{D}_{\text{pref}}$ with global and local preference labels, the resulting preference predictor gives a preference score for each agent trajectory:

$$\rho(\sigma_i) = w_i \sum_{t=0}^{T-1} w_t \rho(o_t^i, a_t^i) \tag{15}$$

Agent's contribution can be further evaluated by applying the softmax function on preference scores resulting in normalized agent-aware importance weights:

$$\omega_{\sigma_i} = \frac{\exp \rho(\sigma_i)/\tau}{\sum_{j=1}^{N} \exp \rho(\sigma_j)/\tau} \tag{16}$$

where τ is the temperature coefficient. Combining the agent-aware weights with policy-segment divergence formulates the agent-aware direct preference optimization objective:

$$d_A(\boldsymbol{\pi_{tot}}, \boldsymbol{\sigma}) = \textstyle\sum_N \omega_{\sigma_i} d(\pi^i, \sigma_i) \tag{17}$$

$$\begin{aligned} &\mathcal{L}_{policy} = \\ &-\mathbb{E}_{(\boldsymbol{\sigma^+},\boldsymbol{\sigma^-})\sim\mathcal{D}_{\text{pref}}}\left[\log \frac{\exp -d_A(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^+})}{\exp -d_A(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^+})+\lambda_p \exp -d_A(\boldsymbol{\pi_{tot}},\boldsymbol{\sigma^-})}\right] \end{aligned} \tag{18}$$

Table 1: Evaluation results on different global-local preference inconsistency SAMC datasets across baselines.

Difficulty	Task	Dataset	BC	ICQ	ICQ+MLP	ICQ+LSTM	ICQ+PT	ICQ+MAPT	CPL	DPPO	**Ours**
Easy	3m	Low	3.74±1.31	5.79±1.19	0.14±0.06	4.16±0.72	5.02±0.25	4.29±1.03	5.12±1.38	5.72±3.26	**8.79±1.47**
		Medium	4.79±0.57	7.68±1.26	0.16±0.09	10.6±1.49	9.72±1.68	**11.1±2.6**	6.67±0.9	6.11±3.87	8.56±0.93
	8m	Low	3.31±0.15	5.00±1.66	0.17±0.03	0.17±0.06	0.17±0.09	0.17±0.07	3.64±0.26	2.94±0.24	**9.83±0.51**
		Medium	4.43±0.33	11.54±1.46	2.7±0.23	2.7±0.67	2.7±0.54	2.7±0.05	7.41±1.45	10.28±2.82	**14.97±1.46**
	2s3z	Low	4.56±0.78	8.13±0.61	5.26±0.57	8.44±0.95	6.89±0.75	0.52±0.38	6.52±2.12	3.11±1.74	**9.07±0.33**
		Medium	5.77±0.95	**13.9±0.78**	0.51±0.39	11.13±0.29	12.45±0.46	0.5±0.37	9.5±1.14	3.57±1.11	**13.04±0.32**
	1c3s5z	Low	6.78±1.29	7.88±0.16	2.32±0.21	2.82±0.15	3.3±0.8	3.68±0.35	**9.44±0.84**	7.74±0.26	**8.47±0.93**
		Medium	8.76±0.41	**17.7±1.11**	4.69±0.18	6.79±4.31	3.07±0.34	8.08±5.25	12.7±0.35	11.04±0.13	**12.62±0.26**
Hard	3s_vs_5z	Low	5.4±0.33	6.66±0.65	2.74±0.48	2.26±0.63	4.7±0.21	2.69±0.63	**6.67±1.31**	5.52±0.24	**5.95±1.23**
		Medium	8.79±1.61	11.5±1.34	1.68±0.65	1.82±0.66	1.03±0.22	1.65±1.45	8.96±4.62	9.47±2.67	**12.53±1.60**
	5m_vs_6m	Low	5.38±1.29	5.28±0.41	0.32±0.21	4.02±0.15	5.3±0.8	4.68±0.35	3.71±0.23	2.68±0.32	**6.22±0.24**
		Medium	4.83±0.31	5.36±0.63	0.71±0.24	2.87±0.24	3.07±0.34	5.78±0.35	4.5±0.34	3.69±0.16	**6.42±0.42**
	8m_vs_9m	Low	5.38±1.29	6.39±1.18	0.32±0.21	8.02±1.39	5.3±0.8	6.9±1.08	7.02±1.38	7.46±1.33	**8.55±1.14**
		Medium	4.83±0.31	9.57±0.58	0.35±0.24	3.21±0.25	8.75±0.26	9.54±0.48	7.52±0.37	3.04±0.48	**11.88±0.58**
	10m_vs_11m	Low	4.77±0.47	8.92±0.16	0.14±0.06	9.47±0.99	7.62±0.29	**9.71±0.18**	7.82±0.19	6.54±0.34	**8.94±0.26**
		Medium	5.06±0.37	8.95±0.36	0.13±0.09	9.63±0.13	9.6±0.65	**10.1±0.32**	8.13±0.23	10.06±0.32	**9.13±0.16**
Super Hard	MMM2	Low	2.66±0.27	6.37±0.36	0.25±0.20	4.16±0.15	6.48±0.45	6.10±0.59	4.55±0.24	3.67±0.22	**6.58±0.47**
		Medium	3.21±0.25	9.14±0.32	0.21±0.19	3.68±0.58	6.74±0.86	5.80±0.43	7.42±1.45	8.36±1.66	**9.54±0.88**
	6h_vs_8z	Low	5.79±0.13	**7.22±0.64**	5.31±0.59	5.59±0.58	7.02±0.79	5.65±0.63	6.72±0.63	6.88±0.45	**6.97±0.25**
		Medium	3.21±0.25	**11.3±0.75**	1.21±0.56	2.63±0.52	5.56±0.45	1.30±0.69	8.69±0.67	6.62±0.26	**9.34±0.70**
	corridor	Low	5.33±0.22	6.69±0.50	1.44±0.52	1.66±0.62	1.62±0.84	6.05±0.42	6.42±0.53	4.52±0.18	**7.06±0.49**
		Medium	6.66±0.57	10.4±0.45	1.33±0.46	1.11±0.42	2.19±0.55	0.63±0.43	7.75±0.22	5.42±0.56	**13.7±0.33**
	3s5z_vs_3s6z	Low	5.62±0.33	7.74±0.19	1.49±0.26	0.56±0.22	8.02±0.23	2.69±2.53	6.56±0.19	4.92±0.43	**8.09±0.99**
		Medium	7.24±0.14	13.89±0.47	0.74±0.26	10.1±0.36	6.12±0.35	0.61±0.29	7.8±0.15	6.29±0.46	**14.14±0.47**

There is still a small problem with this objective. Agent-aware weights assign higher ω_{σ_i} for better agents which make good agent trajectories dominate the policy-segment divergence. This is undesirable for $\boldsymbol{\sigma^-}$ since the objective aims to keep policy away from it, leading to the isolation from good agents policy in $\boldsymbol{\sigma^-}$. To fix this problem, we reverse the agent-aware weights for $\boldsymbol{\sigma^-}$:

$$\omega_{\sigma_i^-} = \frac{\exp -\rho(\sigma_i^-)/\tau}{\sum_{j=1}^{N} \exp -\rho(\sigma_j^-)/\tau} \tag{19}$$

Finally, we can utilize Eq. 18 to directly optimize policy in global-local preference inconsistency datasets and learn from good agents in both preferred and less preferred segments.

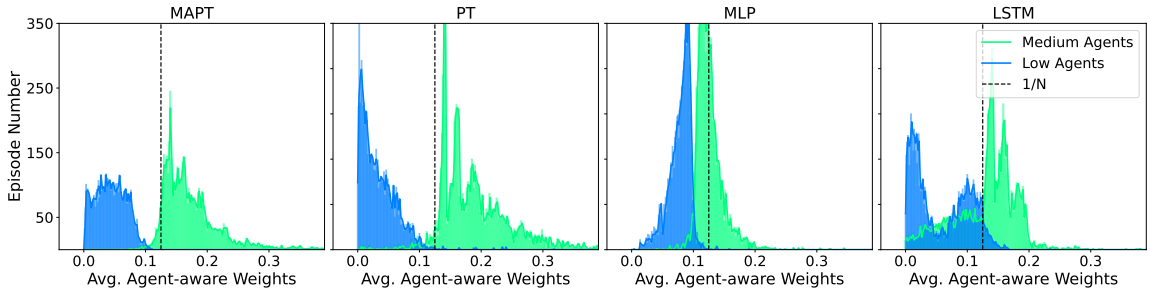
5 EXPERIMENTS

In this section, we first briefly introduce the StarCraft Multi-Agent Challenge (SMAC) [35] environment and then present the pipeline for global-local preference inconsistency multi-agent datasets generation in SMAC. Then we evaluate AMADPO against baselines on these datasets with further analysis experiments to demonstrate the outperformance of our method.

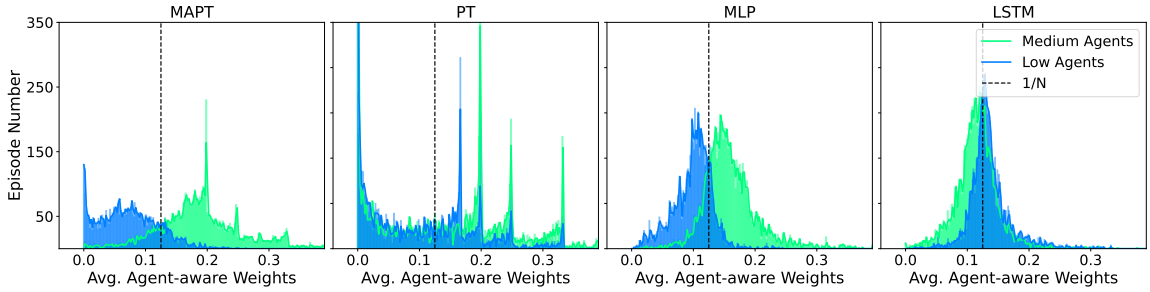
5.1 Experiment Settings

Benchmark Datasets. Experiments are mainly conducted on the SMAC[35] environment which needs accurate control of individual units to complete various cooperation tasks. We select twelve maps that include both homogeneous and heterogeneous tasks across three difficulty levels. In order to construct global-local inconsistency preference datasets, similar to previous work [38], we collect diverse policies by first training joint policies using online algorithms QMIX [34] and EMC [48] and store them at fixed intervals. Next, these policies are divided into poor, medium and good policy pools according to episode returns. Finally, we can sample policies from different policy pools for individual agents to generate global-local preference inconsistency trajectories. For preference labels, we use synthetic labels provided by scripted teachers [7, 18, 21]. Script teacher generates global preference labels based on the ground truth task rewards while generates local preferences according to the level of agents' controlled policy. Specifically, we construct two quality types of preference datasets for each map.

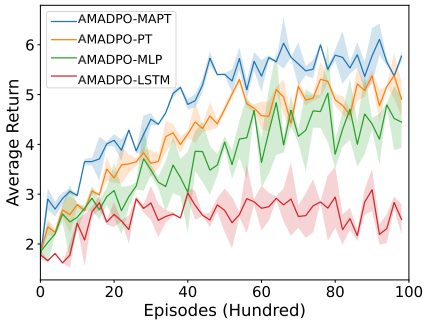
To further illustrate the datasets, we take task ***2s3z*** for example. First, we randomly select 1~3 agents to be controlled by policies sampled from medium policy pools while rest are controlled by poor policies to rollout trajectories. Then we sample pairs of segments to label global and local preference labels. A possible preference pair could be ($\boldsymbol{\sigma^0} = \{s_1^m, s_2^p, z_1^m, z_2^m, z_3^p\}, \boldsymbol{\sigma^1} = \{s_1^p, s_2^m, z_1^p, z_2^p, z_3^m\}, y = 0, \{y_i\} = \{0, 1, 0, 0, 1\}$), where s, z are agent types and superscript m, p indicate medium and poor policies. This results in the low quality dataset while medium quality dataset needs good and poor policies (We mainly focus on non-expert data). We generate 500 quadruples of $(\boldsymbol{\sigma^0}, \boldsymbol{\sigma^1}, y, \{y_i\})$ as preference dataset $\mathcal{D}_{\text{pref}}$ and 3000 pairs of $(\boldsymbol{\sigma^0}, \boldsymbol{\sigma^1})$ as unlabeled dataset $\mathcal{D}_{\text{unlabel}}$ for each map with two qualities.



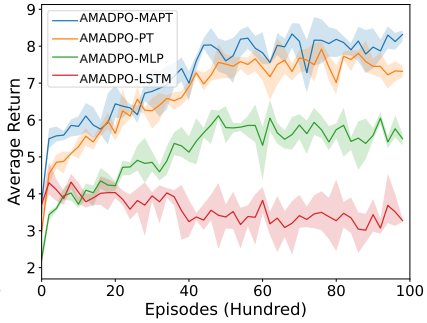
(a) Average agent-aware weights of Preference Predictors in Low quality MMM2 task.



(b) Average agent-aware weights of Preference Predictors in Low quality 3s5z_vs_3s6z task.



(c) Learning curves of AMADPO in Low quality MMM2 task.



(d) Learning curves of AMADPO in Low quality 3s5z_vs_3s6z task.

Figure 4: Analysis on Preference Predictor in super hard tasks.

Baselines. We compare our method against offline MARL, PbRL, and DPO methods. For offline MARL, we choose the state-of-the-art algorithm ICQ [44] as the performance baseline trained with Ground Truth (GT) rewards. Behavior Clone (BC) can extract the behavior policy of the dataset. Following previous work [49], we also consider PbRL methods with different reward models: MLP [7], LSTM [8], Preference Transformer (PT) [18], Multi-agent Preference Transformer (MAPT) [49]. Notably, we also apply additional local preferences to train the reward models for fair competition. We select CPL [12] and DPPO [2] for DPO methods and adapt them in multi-agent form with Eq.9. As for our method AMADPO, we choose MAPT as the structure of multi-agent preference predictor and d_{policy} as d_{oa} (relevant ablations are discussed in). We first train the multi-agent preference predictor with the ensembling method on $\mathcal{D}_{\text{pref}}$ and then use it to label $\mathcal{D}_{\text{unlabel}}$ with global preference and produce agent-aware importance weights to directly optimize joint policy from preference labels.

5.2 Comparative Evaluation Results

We report the mean and variance of episode returns for each algorithm in Table 1. Every algorithm is evaluated for 32 episodes and 5 random seeds. The comparative evaluation results demonstrate that AMADPO significantly outperforms other baselines in most tasks. ICQ with different reward models achieving less competitive results compared to AMADPO shows the negative impact of the reward-preference mismatch problem due to the isolation of reward model training and instability of RL learning in global-local inconsistency preference datasets. Our method also surpasses the multi-agent version of CPL and DPPO which reveals the necessity of multi-agent preference predictor and agent-aware weights. These results emphasize the effectiveness of agent-aware importance weights guided direct preference optimization in the imbalanced situation. Remarkably, AMADPO enables learning from good agents of both compared segments in new unseen datasets without unstable MARL training process.

Table 2: Ablation study on reverse agent-aware weights.

Task	Dataset	**AMADPO-Reverse**	AMADPO-Origin
8m_vs_9m	Low	**9.46 ± 0.07**	6.82 ±0.34
	Medium	**11.3 ± 0.23**	7.86 ± 0.18
MMM2	Low	**6.85 ±0.17**	4.78 ± 0.31
	Medium	**8.92 ± 0.43**	7.01 ± 0.21

Table 3: Ablation study on divergence selection.

Task	Dataset	**AMADPO-d_{policy}**	AMADPO-d_{action}
8m_vs_9m	Low	**8.37 ± 0.07**	6.82 ± 0.34
	Medium	**11.02 ± 0.23**	7.86 ± 0.18
MMM2	Low	**3.56 ± 0.43**	7.01 ± 0.31
	Medium	**9.12 ± 0.17**	4.78 ± 0.21

5.3 Preference Predictor Analysis

To obtain explicit agent-aware importance weights, it is extremely important to correctly model the preference predictor. We evaluate the capabilities of four different model structures to indicate good agents in global-local inconsistency preference datasets. Each model is trained on $\mathcal{D}_{\text{pref}}$ with both global and local preference labels and then used to predict global preferences and produce agent-aware importance weights on $\mathcal{D}_{\text{unlabel}}$. Figure 4a,4b show the average agent-aware weights of medium and poor agents on low quality datasets of two super hard tasks ***MMM2*** and ***3s5z_vs_3s6z***. Ideally, preference predictor should assign higher weights for the preferred agents in both segments across cooperation-wise. However, MLP and LSTM fail to extract the explicit agent-aware weights for many poor agents receiving importance weights over the average line $1/N$, even trained with additional local preference labels. PT and MAPT achieve reasonable results and MAPT performs better due to both temporal-wise and agent-wise preference modeling. Learning curves are illustrated in Figure 4c,4d and it can be found that using MAPT as preference predictor results in best policy optimization.

5.4 Ablation Studies

In this section, we conduct ablations studies on designed components and hyperparameters selection of AMADPO.

Reverse agent-aware importance weights. During applying agent-aware importance weights to direct preference optimization, we reverse the weights of less preferred segment σ^-. Due to the distancing effect of the objective in Eq.18 on negative segments, the aggregation of original agent-aware weights and policy-segment distance can keep joint policy away from good agents' behaviors in negative segments. Reverse agent-aware importance weights can effectively address this issue by turning the major part of negative segments to poor agents, thus avoiding pushing away policy from good agents. Ablation results are shown in Table2. AMADPO's performance degrades without reverse agent-aware weights.

Divergence selection. We absorb previous DPO studies [2, 12] and uniformly define the policy-related metric as policy-segment divergence. Policy-segment divergence has many forms, but we mainly implement two types of divergence and extend them to the multi-agent setting as mentioned in Section 4.2. Experiments using two forms are shown in Figure 5 which indicates that d_{action} gives less guidance for policy optimization. This is most likely due to the discrete action setting in SMAC environment which suffers from $L2$ loss optimization [4]. According to the results, we choose d_{policy} for direct preference optimization. In order to utilize d_{policy}, we first obtain behavior policy $\boldsymbol{\mu^{tot}}$ through BC before applying AMADPO.

Hyperparameters selection. There are two main hyperparameters need to conduct ablations: regulariser λ_p and temperature τ. λ_p is a regularizer used to control the divergence between joint policy and negative segments. Intuitively, the smaller λ_p is, the farther policy is away from negative segments. Actually, our method is insensitive to λ_p, probably due to the agent-aware importance weights have already implicitly regularised less preferred behaviors in negative segments. From Table 4, our method is also robust to τ in a narrow range.

Table 4: Ablation study on hyperparameter selection.

Task	Dataset	$\lambda_p = 0.1$	$\lambda_p = 0.5$	$\lambda_p = 1$
MMM2	Low	5.91 ± 0.66	6.37 ± 0.17	6.66 ± 0.49
	Medium	8.13 ± 0.46	7.95 ± 0.25	7.65 ± 0.24
8m_vs_9m	Low	8.63 ± 0.61	8.52 ± 0.31	6.89 ± 0.25
	Medium	10.61 ± 0.78	10.25 ± 0.37	9.16± 0.46
		$\tau = 0.5$	$\tau = 1$	$\tau = 2$
MMM2	Low	6.37± 0.17	6.34 ± 0.34	5.78 ± 0.14
	Medium	8.37 ± 0.07	8.32 ± 0.34	8.31 ± 0.24
8m_vs_9m	Low	5.44 ± 0.61	7.47 ± 0.31	8.58 ± 0.75
	Medium	10.16 ± 0.17	9.91 ± 0.21	9.16 ± 0.12

6 CONCLUSION

In this paper, we analyze the preference-behavior mismatch problem in current Multi-agent Preference-Based Reinforcement Learning (MAPbRL) methods, stemming from the global-local preference inconsistency and learning instability in the two-stage training paradigm of PbRL. To address this problem, we propose a novel offline MAPbRL framework named Agent-aware Multi-agent Direct Preference (AMADPO). AMADPO introduces agent-wise preference labels to identify and utilize the good agents' experiences in preferred and less-preferred trajectories sampled from the offline training datasets. In addition, we also design a policy-related metric and agent-aware importance weights to guide and rectify the learning gradients for good agents' policies during direct policy optimization. We compared AMADPO against various baselines on MAPbRL datasets collected from the Starcraft Multi-agent Challenge benchmark. The experimental results demonstrate the superior performance of the proposed method, especially in tasks that involve the problem of global-local preference inconsistency. Future work includes evaluating our method on continuous environments with extra offline MARL algorithms and studying an end-to-end training paradigm of agent-aware direct joint policy optimization.

ACKNOWLEDGMENTS

This work was supported in part by National Key RD Program of China under grant No. 2021ZD0112700, NSFC under grant No. 62125305, No. U23A20339, No. 62088102, No. 62203348.

REFERENCES

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[2] Gaon An, Junhyeok Lee, Xingdong Zuo, Norio Kosaka, Kyung-Min Kim, and Hyun Oh Song. 2023. Direct preference-based policy optimization without reward modeling. *Advances in Neural Information Processing Systems* 36 (2023), 70247–70266.

[3] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. 2019. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680* (2019).

[4] Christopher M Bishop and Nasser M Nasrabadi. 2006. *Pattern recognition and machine learning*. Vol. 4. Springer.

[5] Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika* 39, 3/4 (1952), 324–345.

[6] Heewoong Choi, Sangwon Jung, Hongjoon Ahn, and Taesup Moon. 2024. Listwise Reward Estimation for Offline Preference-based Reinforcement Learning. *arXiv preprint arXiv:2408.04190* (2024).

[7] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[8] Joseph Early, Tom Bewley, Christine Evers, and Sarvapali Ramchurn. 2022. Non-markovian reward modelling from trajectory labels via interpretable multiple instance learning. *Advances in Neural Information Processing Systems* 35 (2022), 27652–27663.

[9] Scott Fujimoto, David Meger, and Doina Precup. 2019. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*. PMLR, 2052–2062.

[10] Chen-Xiao Gao, Shengjun Fang, Chenjun Xiao, Yang Yu, and Zongzhang Zhang. 2024. Hindsight Preference Learning for Offline Preference-based Reinforcement Learning. *arXiv preprint arXiv:2407.04451* (2024).

[11] Divyansh Garg, Joey Hejna, Matthieu Geist, and Stefano Ermon. 2023. Extreme q-learning: Maxent rl without entropy. *arXiv preprint arXiv:2301.02328* (2023).

[12] Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W Bradley Knox, and Dorsa Sadigh. 2023. Contrastive prefence learning: Learning from human feedback without rl. *arXiv preprint arXiv:2310.13639* (2023).

[13] Joey Hejna and Dorsa Sadigh. 2024. Inverse preference learning: Preference-based rl without a reward function. *Advances in Neural Information Processing Systems* 36 (2024).

[14] Donald Joseph Hejna III and Dorsa Sadigh. 2023. Few-shot preference learning for human-in-the-loop rl. In *Conference on Robot Learning*. PMLR, 2014–2025.

[15] Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. 2018. Reward learning from human preferences and demonstrations in atari. *Advances in neural information processing systems* 31 (2018).

[16] Sehyeok Kang, Yongsik Lee, and Se-Young Yun. 2024. DPM: Dual Preferences-based Multi-Agent Reinforcement Learning. In *ICML 2024 Workshop on Models of Human Feedback for AI Alignment*.

[17] Yachen Kang, Diyuan Shi, Jinxin Liu, Li He, and Donglin Wang. 2023. Beyond reward: Offline preference-guided policy optimization. *arXiv preprint arXiv:2305.16217* (2023).

[18] Changyeon Kim, Jongjin Park, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. 2023. Preference transformer: Modeling human preferences using transformers for rl. *arXiv preprint arXiv:2303.00957* (2023).

[19] Ilya Kostrikov, Ashvin Nair, and Sergey Levine. 2021. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169* (2021).

[20] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 1179–1191.

[21] Kimin Lee, Laura Smith, and Pieter Abbeel. 2021. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. *arXiv preprint arXiv:2106.05091* (2021).

[22] Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. 2021. B-pref: Benchmarking preference-based reinforcement learning. *arXiv preprint arXiv:2111.03026* (2021).

[23] Sergey Levine and Vladlen Koltun. 2013. Guided policy search. In *International conference on machine learning*. PMLR, 1–9.

[24] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643* (2020).

[25] Runze Liu, Fengshuo Bai, Yali Du, and Yaodong Yang. 2022. Meta-reward-net: Implicitly differentiable reward learning for preference-based reinforcement learning. *Advances in Neural Information Processing Systems* 35 (2022), 22270–22284.

[26] Linghui Meng, Muning Wen, Chenyang Le, Xiyun Li, Dengpeng Xing, Weinan Zhang, Ying Wen, Haifeng Zhang, Jun Wang, Yaodong Yang, et al. 2023. Offline pre-trained multi-agent decision transformer. *Machine Intelligence Research* 20, 2 (2023), 233–248.

[27] Frans A Oliehoek, Christopher Amato, et al. 2016. *A concise introduction to decentralized POMDPs*. Vol. 1. Springer.

[28] Afshin Oroojlooy and Davood Hajinezhad. 2023. A review of cooperative multi-agent deep reinforcement learning. *Applied Intelligence* 53, 11 (2023), 13677–13722.

[29] Ling Pan, Longbo Huang, Tengyu Ma, and Huazhe Xu. 2022. Plan better amid conservatism: Offline multi-agent reinforcement learning with actor rectification. In *International conference on machine learning*. PMLR, 17221–17237.

[30] Jongjin Park, Younggyo Seo, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. 2022. SURF: Semi-supervised reward learning with data augmentation for feedback-efficient preference-based reinforcement learning. *arXiv preprint arXiv:2203.10050* (2022).

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

[32] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2024).

[33] Tabish Rashid, Gregory Farquhar, Bei Peng, and Shimon Whiteson. 2020. Weighted qmix: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning. *Advances in neural information processing systems* 33 (2020), 10199–10210.

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

[35] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder De Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. 2019. The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043* (2019).

[36] Daniel Shin, Daniel S Brown, and Anca D Dragan. 2021. Offline preference-based apprenticeship learning. *arXiv preprint arXiv:2107.09251* (2021).

[37] Daniel Shin, Anca D Dragan, and Daniel S Brown. 2023. Benchmarks and algorithms for offline preference-based reward learning. *arXiv preprint arXiv:2301.01392* (2023).

[38] Qi Tian, Kun Kuang, Furui Liu, and Baoxiang Wang. 2023. Learning from good trajectories in offline multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 11672–11680.

[39] Peter Vamplew, Benjamin J Smith, Johan Källström, Gabriel Ramos, Roxana Rădulescu, Diederik M Roijers, Conor F Hayes, Fredrik Heintz, Patrick Mannion, Pieter JK Libin, et al. 2022. Scalar reward is not enough: A response to silver, singh, precup and sutton (2021). *Autonomous Agents and Multi-Agent Systems* 36, 2 (2022), 41.

[40] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *nature* 575, 7782 (2019), 350–354.

[41] Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. 2020. Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062* (2020).

[42] Xiangsen Wang, Haoran Xu, Yinan Zheng, and Xianyuan Zhan. 2024. Offline multi-agent reinforcement learning with implicit global-to-local value regularization. *Advances in Neural Information Processing Systems* 36 (2024).

[43] Yaodong Yang, Jianye Hao, Ben Liao, Kun Shao, Guangyong Chen, Wulong Liu, and Hongyao Tang. 2020. Qatten: A general framework for cooperative multiagent reinforcement learning. *arXiv preprint arXiv:2002.03939* (2020).

[44] Yiqin Yang, Xiaoteng Ma, Chenghao Li, Zewu Zheng, Qiyuan Zhang, Gao Huang, Jun Yang, and Qianchuan Zhao. 2021. Believe what you see: Implicit constraint approach for offline multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 34 (2021), 10299–10312.

[45] Zhilong Zhang, Yihao Sun, Junyin Ye, Tian-Shuo Liu, Jiaji Zhang, and Yang Yu. 2023. Flow to better: Offline preference-based reinforcement learning via preferred trajectory generation. In *The Twelfth International Conference on Learning Representations*.

[46] Dezhong Zhao, Ruiqi Wang, Dayoon Suh, Taehyeon Kim, Ziqin Yuan, Byung-Cheol Min, and Guohua Chen. 2024. PrefMMT: Modeling Human Preferences in Preference-based Reinforcement Learning with Multimodal Transformers. *arXiv preprint arXiv:2409.13683* (2024).

[47] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223* (2023).

[48] Lulu Zheng, Jiarui Chen, Jianhao Wang, Jiamin He, Yujing Hu, Yingfeng Chen, Changjie Fan, Yang Gao, and Chongjie Zhang. 2021. Episodic multi-agent reinforcement learning with curiosity-driven exploration. *Advances in Neural*

Information Processing Systems 34 (2021), 3757–3769.
[49] Tianchen Zhu, Yue Qiu, Haoyi Zhou, and Jianxin Li. 2024. Decoding Global Preferences: Temporal and Cooperative Dependency Modeling in Multi-Agent Preference-Based Reinforcement Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 17202–17210.