

Generalized Per-Agent Advantage Estimation for Multi-Agent Policy Optimization

Seongmin Kim
KAIST
Daejeon, Korea
seongmin.kim@kaist.ac.kr

Giseung Park
University of Toronto
Toronto, Ontario, Canada
giseung.park@utoronto.ca

Woojun Kim
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
woojunk@andrew.cmu.edu

Jiwon Jeon
KAIST
Daejeon, Korea
jiwon.jeon@kaist.ac.kr

Seungyeol Han*
UNIST
Ulsan, Korea
syhan@unist.ac.kr

Youngchul Sung
KAIST
Daejeon, Korea
yjsung@kaist.ac.kr

ABSTRACT

In this paper, we propose a novel framework for multi-agent reinforcement learning that enhances sample efficiency and coordination through accurate per-agent advantage estimation. The core of our approach is Generalized Per-Agent Advantage Estimator (GPAE), which employs a per-agent value iteration operator to compute precise per-agent advantages. This operator enables stable off-policy learning by indirectly estimating values via action probabilities, eliminating the need for direct Q -function estimation. To further refine estimation, we introduce a double-truncated importance sampling ratio scheme. This scheme improves credit assignment for off-policy trajectories by balancing sensitivity to the agent’s own policy changes with robustness to non-stationarity from other agents. Experiments on benchmarks demonstrate that our approach outperforms existing approaches, excelling in coordination and sample efficiency for complex scenarios.

KEYWORDS

Multi-Agent Reinforcement Learning; Multi-Agent Credit Assignment Problem; Policy Optimization

ACM Reference Format:

Seongmin Kim, Giseung Park, Woojun Kim, Jiwon Jeon, Seungyeol Han, and Youngchul Sung. 2026. Generalized Per-Agent Advantage Estimation for Multi-Agent Policy Optimization. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026)*, Paphos, Cyprus, May 25 – 29, 2026, IFAAMAS, 9 pages. <https://doi.org/10.65109/OBIA4073>

1 INTRODUCTION

Multi-agent reinforcement learning (MARL) addresses cooperative tasks where agents learn policies to maximize shared rewards through environmental interactions [24]. These tasks face challenges from coordination requirements and partial observability.

*Corresponding author



This work is licensed under a Creative Commons Attribution International 4.0 License.

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). <https://doi.org/10.65109/OBIA4073>

While independent learning approaches [37] often suffer from non-stationarity issues, the centralized training and decentralized execution (CTDE) paradigm [15, 19] enables agents to utilize global information during training while maintaining local observation dependency during execution.

Within CTDE, various algorithms have emerged, including value-based methods like VDN and QMIX [28, 34] and multi-agent policy gradient (MAPG)-based methods such as COMA and MAPPO [8, 42]. Among these, MAPPO, an extension of Proximal Policy Optimization (PPO, Schulman et al. [32]), has gained popularity for its strong performance. However, MAPPO’s reliance on Generalized Advantage Estimator (GAE) [31] introduces limitations: (1) reduced sample efficiency due to on-policy constraints, and (2) inadequate credit assignment from estimating identical advantages for all agents. While techniques like V-traces enhance sample efficiency in single-agent settings, their application to multi-agent systems has not been theoretically or experimentally explored. The second problem is that MAPPO updates policies based on centralized critic with GAE, which estimates the same advantage for all individual agents’ actions in the same timestep. This approach is inadequate for solving the multi-agent credit assignment problem, which is a significant challenge in MARL where the contribution of individual agents to the global reward must be accurately assessed.

The *multi-agent credit assignment problem* significantly impacts MARL performance [4, 44]. Value-based methods address this with value function factorization [28, 33], but lack theoretical guarantees and often fail to provide accurate credit assignment in complex tasks. In MAPG area, COMA employ counterfactual baselines but suffer from variance and scalability issues. DAE [18] offers a GAE-based approach but lacks policy-invariance [22], potentially leading to sub-optimal convergence.

Motivated by these limitations, we investigate an advantage estimator for ensuring precise per-agent credit assignment at n -step horizons, remains policy-invariant, and enables stable off-policy reuse under MAPG within CTDE.

The key contributions of our work are summarized as follows:

- We propose *Generalized Per-Agent Advantage Estimator*, which provides explicit per-agent credit signals under CTDE and unifies on-policy learning and off-policy reuse within a single estimator.
- We establish contraction of the per-agent operator and show policy invariance of GPAE theoretically.

- We propose a *double-truncated* importance sampling scheme tailored to multi-agent coupling, improving stability and maintaining credit fidelity compared to single-truncation or individual-only truncation.
- We demonstrate the practical benefits of our method through extensive experiments, showing significant performance improvements in MARL tasks.

2 BACKGROUND

2.1 Dec-POMDP and CTDE Setup

A multi-agent system can be modeled as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) [23], defined by the tuple $\langle \mathcal{S}, \mathcal{A}, P, R, \mathcal{Z}, O, n, \gamma \rangle$, where \mathcal{S} is the global state space, $\mathcal{A} = \prod_{i=1}^n \mathcal{A}^i$ is the joint action space of n agents, with \mathcal{A}^i being the action space of agent i , $P(s'|s, \mathbf{a})$ is the state transition probability given the current state s and joint action $\mathbf{a} = \{a^1, \dots, a^n\}$, $R(s, \mathbf{a})$ is the shared reward function, \mathcal{Z} is the observation space, and $O(o^i|s, i)$ specifies the observation probabilities for each agent i , and $\gamma \in [0, 1)$ is the discount factor.

The goal of MARL is to learn a policy $\pi = \prod_{i=1}^n \pi^i(a^i|o^i)$ for each agent that maximizes the expected cumulative reward $J(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \mathbf{a}_t) \right]$. The CTDE paradigm facilitates MARL by allowing agents to use global information including full state s during training, while restricting each agent’s policy to depend only on local observations o_i during execution. Note that our theoretical analysis assumes full observability during training, which is common in many CTDE-based MARL studies [17, 39] to enable rigorous analysis. Extending theory to partially observable settings remains challenging, as solving Dec-POMDPs is NEXP-complete and requires super-exponential time in the worst case [1, 43].

2.2 Advantage Estimation and Credit Assignment

In MAPG methods, accurate advantage estimation plays a central role in stabilizing policy updates and enabling effective credit assignment. Among existing estimators, GAE [31] is widely adopted for its ability to balance bias and variance through an exponentially-weighted sum of temporal difference (TD) errors with bias-variance trade-off parameter λ .

MAPPO [42], a strong baseline in cooperative MARL, extends PPO to the multi-agent setting under CTDE. It applies GAE as the default advantage estimator for each agent. However, MAPPO treats all agents identically by assuming a shared team advantage:

$$A^i(s, \mathbf{a}) = A^{\text{global}}(s, \mathbf{a}) \quad \forall i, \quad (1)$$

which ignores the agent-specific impact on the joint outcome and thus limits the fidelity of credit assignment. To address this limitation, COMA [8] proposes a per-agent advantage formulation using a counterfactual baseline that marginalizes out the individual agent’s contribution. COMA computes a per-agent advantage using a counterfactual baseline as:

$$A^{i, \text{COMA}}(s, \mathbf{a}) = Q(s, \mathbf{a}) - \sum_{a^i} \pi_i(a^i|o^i) Q(s, (a^i, \mathbf{a}^{-i})), \quad (2)$$

where \mathbf{a}^{-i} represents the actions of all agents except i . This formulation enables per-agent credit assignment by marginalizing over agent i ’s actions.

More recently, Difference Advantage Estimation (DAE) [18] was developed as a GAE-based alternative by incorporating potential-based difference rewards to address multi-step credit assignment:

$$A_t^{i, \text{DAE}} = \sum_{l=0}^{\infty} (\gamma \lambda)^l \delta_{t+l}^i, \quad (3)$$

where $\delta_{t+l}^i = r_{t+l} - \beta^{l+1} \mathbb{E}_{a^i} [r_{t+l}] + \gamma V_{t+l+1} - V_{t+l}$ with bias-credit assignment control parameter β .

3 MOTIVATION

MARL poses unique challenges, among which the multi-agent credit assignment problem is a key bottleneck. Accurately assigning credit to individual agents based on their contributions to the global reward is critical for effective learning. However, existing methods in MAPG domain exhibit fundamental limitations.

Building on the background and Eq. (1), we note that MAPPO’s reliance on a single advantage function for every agent can mask each agent’s distinct contributions. COMA advantage relies on TD(0) estimation, which limits its ability to leverage n -step estimation and n -step credit-assignment. DAE [18] extends GAE with a potential-based difference reward to address credit assignment. While DAE allows a n -step credit assignment, it is not policy-invariant due to the estimated reward bias term, which can lead to suboptimal results. Additionally, DAE suffers from instability in explicit reward estimation.

Beyond credit assignment, sample inefficiency further complicates learning in MAPG. Importance sampling offers a theoretical means to utilize off-policy data, but high variance under MARL often reduces its practical utility. While single-agent methods like V-trace [7] control variance by truncating importance sampling ratios, their effectiveness does not readily transfer to multi-agent systems because each agent’s behavior interdepends on others.

We thus seek an advantage estimator that (i) delivers per-agent, n -step credit signals under CTDE, (ii) remains policy-invariant to avoid bias, and (iii) is compatible with off-policy data. Table 1 summarizes how existing MAPG estimators fall short with respect to these criteria.

Table 1: Comparison of MAPG advantage estimators

Advantage	Credit Assignment	Off-Policy	Additional Bias($\lambda = 1$)
GAE	No	No	No
COMA	1-step	No	No
DAE	n -step	No	Yes
GPAE (Ours)	n-step	Yes	No

Guided by these requirements, we introduce GPAE, which provides n -step, per-agent credit signals while preserving policy invariance and enabling off-policy reuse.

3.1 Credit Assignment in the Presence of Anomalous Behavior

Before detailing the methodology, we provide some preliminary experimental results in this section to reinforce our motivation. Here, we examine how various advantage estimators respond when an agent deviates from cooperation.

For the experiment, we use the SMAX [29] environment, a JAX-implemented version of SMAC [30], which is a widely used benchmark for MARL algorithms. In SMAX, agents collaborate to achieve a common goal, typically involving cooperation in battle against enemy. In the SMAX-3m task, three agents cooperate to win a battle, requiring careful coordination and credit assignment.

To demonstrate how methods handle the multi-agent credit assignment, we modify the SMAX-3m task by introducing 5% probability that one agent performs a “stop” action. This irregularity disrupts coordination, underscoring the need for robust credit assignment and stable advantage estimation.

We focus on whether each advantage estimator can accurately penalize the “misbehaving” agent’s suboptimal action and thus guide the overall team to learn more robust behaviors. In particular, we track the difference in per-agent advantage at the timestep where the anomalous action occurs. Formally, we measure:

$$\Delta A = \frac{1}{N-1} \sum_{j \neq i} \hat{A}_t^j - \hat{A}_t^i,$$

where \hat{A}_t^i is the per-agent advantage for the misbehaving agent i at timestep t , and \hat{A}_t^j are the advantages of the other $N-1$ agents. A higher ΔA indicates that the advantage more strongly penalizes the anomalous agent relative to agents acting optimally, implying better credit assignment.

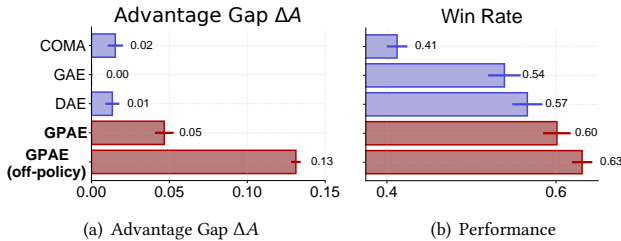


Figure 1: (a) Advantage gap ΔA indicating how strongly each method penalizes the anomalous agent. Higher ΔA means more effective credit assignment against the “stop” action. (b) Average win rates, illustrating overall learning stability and performance.

Figure 1(a) compares the *advantage gap* ΔA of several advantage estimators. Our proposed GPAE yields the highest ΔA (particularly in its off-policy form), demonstrating its ability to measure credit assignment effectively, while COMA and DAE achieves a nonzero gap but are less effective in credit assignment. Note that GAE using common advantage for all agents cannot extract advantage gap. Figure 1(b) shows the average win rate, assessing the overall performance. For this figure, we used the same PPO-style policy optimization only with advantage changed.

Note that COMA performs significantly worse than other advantage estimation methods even though it extracts larger advantage gap than GAE and DAE. The main difference of COMA from other methods is that COMA uses one-step estimation for advantage, which is prone to large errors, whereas all others use n -step advantage estimation. Hence, we recognize that the accuracy of estimation by n -step estimation as well as credit assignment is important for performance. It is seen that under the n -step estimation framework, the advantage gap, i.e., credit assignment indeed translates to the overall performance and its impact saturates as the advantage gap increases, following our intuition.

These observations motivate our approach to unify on-policy and off-policy advantage estimation around n -step signals while preserving per-agent credit assignment. In the following sections, we introduce GPAE and the associated DT-ISR weighting, along with theoretical properties and empirical validation.

4 METHODOLOGY

In this section, we introduce two contributions of our work. First, based on our value iteration operator, we propose a novel advantage estimator, called **Generalized Per-agent Advantage Estimator (GPAE)**. Then we extend the method to the off-policy case with theoretical insights, and propose a *double-truncated importance sampling* for the off-policy correction term in the weighted advantage.

4.1 Generalized Per-Agent Advantage Estimation

To begin, we consider the counterfactual value function $\mathbb{E}_{a^i \sim \pi^i} [Q(s, a^i, \mathbf{a}^{-i})]$, where Q denotes the joint action-value function estimate Q^π , and \mathbf{a}^{-i} represents the actions of all agents except agent i . For convenience, we define:

$$\overline{EQ}^i := \mathbb{E}_{a^i \sim \pi^i} [Q(s, a^i, \mathbf{a}^{-i})].$$

Here, \overline{EQ}^i serves as an estimate of the *per-agent state-value function*, which averages out only the action of agent i under its policy π^i and thus has (s, \mathbf{a}^{-i}) as its argument, in contrast to the full state-value function $V(s) = \mathbb{E}_{a \sim \pi} [Q(s, \mathbf{a})]$ having only s as the argument.

To build an efficient way to update \overline{EQ}^i , we present a new per-agent value iteration operator \mathcal{R}^i , a specifically designed for MARL to facilitate effective credit assignment by focusing on per-agent value estimation. For now, it only considers on-policy learning, thus we denote the operator with $\mathcal{R}_{\text{on}}^i$.

Definition 4.1. The value iteration operator $\mathcal{R}_{\text{on}}^i$ for \overline{EQ}^i is defined as

$$\mathcal{R}_{\text{on}}^i \overline{EQ}^i := \overline{EQ}^i + \mathbb{E}_{a_0^i \sim \pi^i} \left[\mathbb{E}_\pi \left[\sum_{t \geq 0} (\gamma \lambda)^t (r_t + \gamma \overline{EQ}_{t+1}^i - \overline{EQ}_t^i) \right] \middle| (s_0, a_0^i, \mathbf{a}_0^{-i}) = (s, a^i, \mathbf{a}^{-i}) \right], \quad (4)$$

where $\overline{EQ}_t^i := \mathbb{E}_{a_t^i \sim \pi^i} [Q(s_t, a_t^i, \mathbf{a}_t^{-i})]$ and $\lambda \in (0, 1]$ is a bias-variance trade-off parameter.

$\mathcal{R}_{\text{on}}^i$ introduces the ability to compute per-agent state-value estimates efficiently, incorporating partial averaging over the actions

of the agent i while retaining dependencies on the actions of other agents, \mathbf{a}^{-i} . This structure enables n -step credit assignment from the value estimation step, in contrast to existing methods that consider only 1-step credit assignment and value estimation separately. Our proposed method provides a more stable and efficient learning method with estimation by connecting advantage and value in a novel way.

The operator $\mathcal{R}_{\text{on}}^i$ possesses desirable theoretical properties. We establish the contraction property of \mathcal{R}^i , which ensures convergence to a unique fixed point:

Theorem 4.1. $\mathcal{R}_{\text{on}}^i$ is a γ -contraction, which means that \overline{EQ}^i converges to the unique fixed point. If $\lambda = 1$, the fixed point is $\mathbb{E}_{a^i \sim \pi^i} [Q^\pi(s, a^i, \mathbf{a}^{-i})]$.

PROOF SKETCH. Rewriting R_{on}^i by shifting the TD-error indices shows that the difference between two inputs $\overline{EQ}^{i,1}$ and $\overline{EQ}^{i,2}$ is scaled by a geometric factor $(\gamma\lambda)^t$. Taking the sup-norm gives an overall contraction of at most γ , so R_{on}^i is a γ -contraction. For $\lambda = 1$, the TD terms telescope, leaving $[Q^\pi(s, \mathbf{a}) - \mathbb{E}_{a^i \sim \pi^i} [Q(s, a^i, \mathbf{a}^{-i})]]$, and the fixed point becomes $\mathbb{E}_{a^i \sim \pi^i} [Q^\pi(s, a^i, \mathbf{a}^{-i})]$. \square

Theorem 4.1 states that $\mathbb{E}_{a^i \sim \pi^i} [Q^\pi(s, a^i, \mathbf{a}^{-i})]$ can be estimated directly from the policy distribution of agent i and the actions of the remaining agents, without the dual process of estimating the true joint Q value and marginalizing over the policy at Q .

From the definition 4.1 to compute the n -step advantage with per-agent TD error without compromising the structure of the operator $\mathcal{R}_{\text{on}}^i$, the following theorem is presented:

Theorem 4.2. The inner conditional expectation term of \mathcal{R}^i , i.e., $\mathbb{E}_\pi [\sum_{t \geq 0} (\gamma\lambda)^t (r_t + \gamma \overline{EQ}_{t+1}^i - \overline{EQ}_t^i)]$ given $(s_0, \mathbf{a}_0^i, \mathbf{a}_0^{-i}) = (s, a^i, \mathbf{a}^{-i})$ reduces to $Q^\pi(s, \mathbf{a}) - \mathbb{E}_{a^i \sim \pi^i} [Q(s, a^i, \mathbf{a}^{-i})]$, ensuring policy invariance with $\lambda = 1$.

PROOF SKETCH. With $\lambda = 1$, the sum $\sum_t (r_t + \gamma \overline{EQ}_{t+1}^i - \overline{EQ}_t^i)$ collapses to $Q^\pi(s, \mathbf{a}) - \mathbb{E}_{a^i \sim \pi^i} [Q(s, a^i, \mathbf{a}^{-i})]$ by telescoping. Since the baseline term involving \overline{EQ}^i has zero gradient under decentralized execution, this advantage yields the correct policy gradient. \square

Theorem 4.2 suggests that these n -step TD-errors can be utilized to achieve unbiased policy updates. In general, The trade-off between variance and bias can be controlled by adjusting λ .

Using Theorem 4.2, we propose the GPAE for on-policy learning:

$$\hat{A}_t^{i, \text{GPAE, on}} := \sum_{l \geq t} (\gamma\lambda)^{l-t} \delta_l^{i, \text{GPAE}}, \quad (5)$$

where $\delta_l^{i, \text{GPAE}} = r_l + \gamma \overline{EQ}_{l+1}^i - \overline{EQ}_l^i$ is the per-agent TD error.

In the single-agent case, GPAE reduces to GAE(λ). This demonstrates GPAE as a generalization of GAE(λ) for multi-agent scenarios with explicit credit assignment.

4.2 Extension to Off-Policy Estimation

While \mathcal{R}^i and GPAE were originally defined for on-policy learning, adapting them to off-policy scenarios requires addressing the mismatch between the behavior policy μ^i and the target policy π^i . To achieve this, we reformulate $\mathcal{R}_{\text{on}}^i$ to include importance sampling ratio (ISR) weights, allowing the estimation of per-agent value functions and advantages in an off-policy setting.

Definition 4.2. The generalized per-agent operator \mathcal{R}^i is now defined as:

$$\mathcal{R}^i \overline{EQ}^i := \overline{EQ}^i + \mathbb{E}_{a_0^i \sim \mu^i} \left[\rho_0^i \mathbb{E}_\mu \left[\sum_{t \geq 0} \gamma^t \left(\prod_{j=1}^t c_j^i \right) \cdot \left(r_t + \gamma \overline{EQ}_{t+1}^i - \overline{EQ}_t^i \right) \middle| s_0, a_0^i, \mathbf{a}_0^{-i} \right] \right], \quad (6)$$

where $(s_0, \mathbf{a}_0) = (s, \mathbf{a})$, $\rho_0^i = \frac{\pi^i(a_0^i | s_0^i)}{\mu^i(a_0^i | s_0^i)}$, $\boldsymbol{\mu} = \prod_{i=1}^N \mu^i$ is the behavior policy, and c_t^i is the trace weight for off-policy correction. For the case of on-policy learning, c_t^i is substituted for the value of λ , a bias-variance trade-off parameter.

This operator introduces the necessary corrections to the per-agent value updates by weighting the contributions of sampled transitions according to their likelihood under the target policy. The theoretical properties of \mathcal{R}^i , including its contraction property, remain intact under the condition that the behavior policy sufficiently explores the state-action space.

The off-policy GPAE is constructed analogously by incorporating ISR weights into the per-agent TD error. Finally, our off-policy generalized per-agent advantage estimator is given by:

$$\hat{A}_t^{i, \text{GPAE}} := \sum_{l \geq t} \gamma^{l-t} \left(\prod_{j=t+1}^l c_j^i \right) \delta_l^{i, \text{GPAE}}, \quad (7)$$

where $\delta_l^{i, \text{GPAE}} = r_l + \gamma \overline{EQ}_{l+1}^i - \overline{EQ}_l^i$ remains the per-agent TD error.

Note that the per-agent value iteration operator denoted by \mathcal{R}^i and $\hat{A}_t^{i, \text{GPAE}}$ expanded to off-policy learning, are the broader framework that encompass on-policy learning case. That is to say, \mathcal{R}^i and $\hat{A}_t^{i, \text{GPAE}}$ in Eq. 6 and 7 inherently transform into $\mathcal{R}_{\text{on}}^i$ and $\hat{A}_t^{i, \text{GPAE, on}}$ in Eq. 4 and 5 in the on-policy scenario, without the necessity of additional modifications.

To ensure the validity of the off-policy extension, we establish the following theoretical guarantees:

Theorem 4.3. \mathcal{R}^i is a γ -contraction in the general off-policy case. $\hat{A}_t^{i, \text{GPAE}}$ also guarantees policy invariance with $c_t^i = \rho_t$.

PROOF SKETCH. In the off-policy operator R^i , each TD term is weighted by $c_t^i \in [0, 1]$. Bounding these weights shows that the operator still scales differences by at most γ , preserving the γ -contraction. When $c_{i,t} = \rho_t$ (full IS ratios), all bias terms vanish and the expression matches the on-policy case [26], giving the same $Q^\pi - \mathbb{E}_{a_i} Q$ form. Thus the estimator remains unbiased. \square

While Theorem 4.3 establishes the theoretical foundations, practical implementation requires addressing variance explosion due to the use of off-policy samples. This phenomenon, arising from the mismatch between behavioral and target policies, necessitates introducing a bias into the ISR weight to stabilize learning. In single-agent settings, techniques like V-trace have been effective in achieving this balance. However, applying V-trace directly to multi-agent systems introduces new challenges, which we address in the following subsection.

4.3 Choice of Off-Policy Correction Term c^i

To stabilize off-policy estimation in MARL, we extend the idea of truncated importance sampling ratio (ISR). In single-agent methods, the weight is defined as $c_t = \min(1, \rho_t)$, which bounds variance by truncating the ISR. Directly applying this to multi-agent systems, however, fails to account for the interaction between agents: a shared joint ratio controls variance but obscures individual contributions, whereas using only agent-specific ratios preserves credit signals but can destabilize training. To address this, we introduce a new double-truncated ISR weight (DT-ISR) that balance variance control with per-agent credit sensitivity.

We first consider two direct approaches to obtain weights in multi-agent systems. Single truncation (ST) applies a shared truncation on a joint ISR $\rho = \prod_{j \in \mathcal{N}} \pi_t^j / \mu_t^j$, defining $c_t^{i,ST} = \lambda \min(1, \rho)$ for every agent i . ST effectively controls variance under non-stationary team dynamics because all agents receive the same bounded weight, but it makes the update insensitive to an individual agent’s policy change and thus weakens per-agent credit signals.

As opposed to ST, we can only consider the ISR of the own policy: $c_t^{i,IT} = \lambda \min(1, \rho_t^i)$, where $\rho_t^i = \pi_t^i / \mu_t^i$ and IT denotes individual truncation. This can be effective to focus on own individual policy, but it ignores shifts induced by the team. As a result, the training signal is evaluated under a mismatched occupancy. This leads to unstable learning due to biased updates since samples collected under outdated team behavior are not sufficiently discounted.

Therefore, it is essential to propose a novel truncation scheme for MARL that incorporates the strengths of both IT and ST. The scheme should be constrained by a bound of 1 while maintaining a close alignment with both individual ISR ρ_t^i and joint ISR ρ .

In this regard, we propose a novel truncation scheme for the trace coefficient of multi-agent off-policy correction: a **double-truncated importance sampling ratio (DT-ISR) weight**:

$$c_t^{i,DT} = \lambda \min(1, \rho_t^i \min(\eta, \rho_t^{-i})), \quad (8)$$

where $\rho_t^{-i} = \prod_{j \neq i} \pi_t^j / \mu_t^j$, and η is a constant that serves to mitigate the effects of other agents’ policies, while making the unique ISR of agent i less intrusive. This formulation ensures that the unique contribution of agent i is preserved while mitigating variance introduced by the collective policies of other agents. DT-ISR provides a better approximation of the ISR weight, enabling more effective off-policy value and advantage estimation with improved stability.

To motivate and evaluate our proposed method, we present key visualizations in Fig. 2. We choose the environment SMAX-1s1z with $0.5M$ training timestep, in which two agents with different dynamics cooperate to destroy enemies, thus accurate credit assignment can drive rapid learning. Here, we take $\eta = 1.05$ for $c_t^{i,DT}$. The Fig. 2(a), 2(b) illustrate the average distance of each truncation weight from the true individual ISR ρ^i and joint ISR ρ , denoted as $d(\rho^i, c^i)$ and $d(\rho, c^i)$, respectively. Fig. 2(c) highlights the average gap between these two distances, which shows how the weight is balanced between the two true ISRs. The average gap is defined as

$$\Delta c^i := \mathbb{E}_\tau [|d(\rho, c^i) - d(\rho^i, c^i)|], \quad (9)$$

where c^i is given off-policy correction term and τ is a trajectory sample. The smaller this gap, the better the truncation method is at approximating the true joint ISR while maintaining accurate individual ISR estimates. As shown in Fig. 2(c), DT-ISR is effective

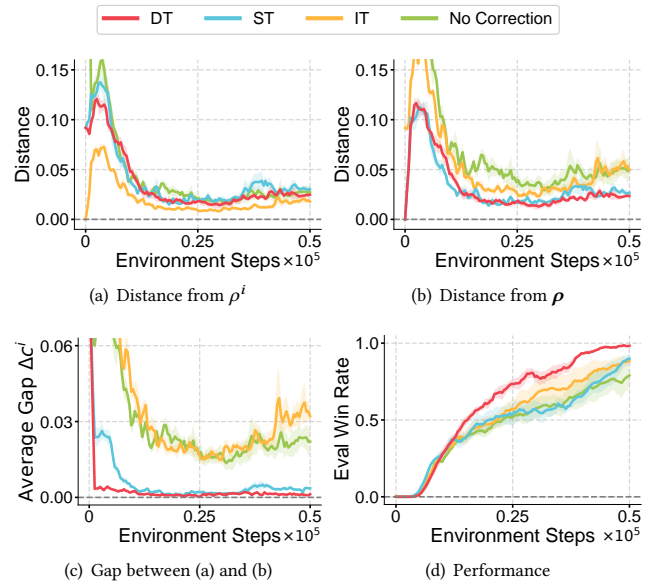


Figure 2: Off-policy correction comparisons in the SMAX-1s1z task. (a) Distance from true ρ^i , (b) Distance from joint true ρ , (c) Gap between (a) and (b), (d) Final performance. Legend is shared across all plots. DT-ISR demonstrates the lowest gap Δc^i and the highest performance.

in reducing the gap Δc^i when compared to other methods. Nonetheless, further validation is required to confirm that the minimization of the gap correlates with stable learning. Fig. 2(d) compares the performance of DT against IT, ST, and without off-policy correction, demonstrating the superiority of DT in achieving higher win rates.

This analysis underscores the advantages of our DT-ISR, laying a strong foundation for further exploration of robust off-policy corrections in MARL.

4.4 Overall Learning Structure and Algorithm

We now summarize the overall training pipeline of GPAE with DT-ISR under CTDE. The procedure integrates the on- and off-policy estimators into a practical optimization loop. At each iteration, (i) trajectories are collected by the joint policy $\pi = \prod_{i=1}^N \pi^i$, (ii) stored into a short replay buffer R , and (iii) gradient updates are performed by sampling mini-batches from R . Actors θ are updated per agent, and a centralized critic head parameterizes $\overline{EQ}^i(s, a^{-i}; \psi)$.

For each sampled trajectories, we compute GPAE by Eq. 7, and the value function is optimized using the following loss function:

$$L(\psi) = \sum_{\tau \in \mathcal{B}} (\overline{EQ}_\psi^i - \overline{EQ}_{targ}^i)^2, \quad (10)$$

where \mathcal{B} represents the replay buffer, and the target value \overline{EQ}_{targ}^i is explicitly given by

$$\overline{EQ}_{targ}^i = \overline{EQ}_\psi^i + \rho_t^i \hat{A}_t^{i,GPAE}, \quad (11)$$

where $\bar{\rho}_t^i = \min(1, \rho_t^i)$ is a bounded individual ISR for stable learning and $\bar{\psi}$ denotes a parameter of target value network. Actors are updated with PPO-style clipped objectives using A_t^i .

The complete procedure is summarized in Algorithm 1, which unifies our theoretical operator into an implementable CTDE algorithm with off-policy sample reuse.

Algorithm 1 GPAE

```

Initialize shared policy parameter  $\theta$ , shared value parameter  $\psi$ 
for  $N$  agents
Initialize trajectory buffer  $R$  with batch reuse length  $M$ 
for each iteration  $m$  do
  Collect a set of trajectories  $B_m$  from the environment by the
  joint policy  $\pi_m$ 
  Store  $B_m$  in the trajectory buffer  $R$ 
  Compute  $\hat{A}_t^{i, \text{GPAE}}$  and  $\overline{EQ}_{targ}^i$  with
  Eq. 7 and Eq. 11 by using the samples stored in  $R$ 
  for each gradient step do
    Update the actor parameter  $\theta$  with PPO actor loss
    Update the critic parameter  $\psi$  with Eq. 10
  end for
end for

```

5 EXPERIMENTAL RESULTS

5.1 Experimental Setup

We evaluated the proposed method in two challenging MARL environments, **SMAX** and **MABrax**, implemented in JAX [2]. These environments are representative of discrete and continuous action domains respectively, and serve as standard testbeds for evaluating GPAE’s credit assignment under partial observability. Figure 3 provides an overview of the two environments used in our evaluation.

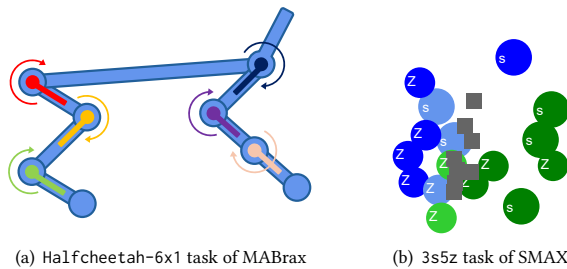


Figure 3: Illustration of the two evaluation environments.

MABrax is a Multi-agent version of Brax [9], where each agent represents an individual joint of a robot tasked with coordinated movement, with continuous action spaces. MABrax focuses on multi-agent continuous control, with each agent corresponding to an individual joint of a robot. proper credit assignment based on centralized values is crucial to ensure synchronized actions among the joints. Limited observations further emphasize the need for precise credit assignment, as effective coordination is necessary to complete tasks successfully. Figure 3(a) illustrates the halfcheetah-6x1 task,

where six agents independently control the hip, thigh, and leg joints of a half-cheetah robot to generate coordinated forward locomotion.

SMAX is a JAX-based implementation of the StarCraft Multi-agent Challenge (SMAC, Samvelyan et al. [30]). SMAX involves discrete action spaces where effective inter-agent cooperation is critical for task success. Agents engage in combat to achieve a common goal, with the win rate serving as the primary evaluation metric. Multi-agent credit assignment plays a pivotal role in enabling agents to coordinate under partial observability. Figure 3(b) includes the 3s5z scenario, where three Stalker units and five Zealot units form a cooperative team against a mirrored adversarial team. The gray squares in the figure visualize attacks between units.

Baselines. Our method is compared against baseline algorithms, focusing on its performance relative to other existing advantage estimating MAPG methods. Specifically, we consider representative baselines including MAPPO [42], which utilizes GAE[31] as an advantage estimator. We also compare against DAE[18], a variant of advantage estimation using reward estimation, and COMA [8], a classical counterfactual baseline for credit assignment. In addition, QMIX [28] and VDN [34] are included as value decomposition approaches for discrete-action tasks. Note that QMIX and VDN are not applicable for continuous-action tasks. Furthermore, we provide an ablation study to assess the impact of the DT-ISR scheme and analyze the influence of hyperparameter η on learning performance.

Including the environment, the implementation of MAPPO, VDN and QMIX is based on Rutherford et al. [29]. For GPAE, we provide our own implementation, where off-policy sample reuse is achieved through a short replay buffer. Since DAE has no existing public implementation, we implemented it from scratch. Similarly, COMA lacks a JAX-based implementation, so we implemented it closely following the original description.

Evaluation Protocol. All methods were trained for 10M timesteps in each environment, and evaluated every 0.5M timesteps. For each task, we conducted 5 independent runs with different random seeds and report the mean and standard deviation across seeds. In SMAX, performance is measured by the win rate, while in MABrax it is measured by episodic return. The results reported in the main tables correspond to the final performance at the end of training. In addition, we provide aggregated learning curves that combine results across all tasks within each environment, where the shaded region indicates the standard error of the mean. Computationally, our algorithm increases wall-clock training time by at most 6% compared to GAE within same MAPPO across all SMAX and MABrax tasks. All experiments were conducted on a server equipped with NVIDIA TITAN Xp GPUs and Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz.

5.2 Comparison with Baselines

Across SMAX tasks, GPAE consistently achieves superior performance over all baseline methods. A notable trend is that MAPPO-based advantage estimators already outperform value decomposition baselines such as QMIX and VDN, underscoring the effectiveness of policy-gradient approaches in cooperative MARL. Building on this, GPAE delivers an additional margin of improvement, demonstrating its ability to stabilize training with precise credit assignment and improve sample efficiency with off-policy sample

Table 2: Performance comparison after learning 1×10^7 timesteps. The upper section shows the average win rate performances in SMAX tasks, with values presented as mean (standard deviation). The lower section shows the episode returns in MABrax tasks, with values presented as mean \pm standard deviation. For each task, we conducted experiments with 5 random seeds. The bold numerals denote the highest performance for each task. Note that QMIX and VDN are not applicable to continuous tasks, and therefore, no results are available for MABrax tasks.

Domain	Task	GPAE (off)	GPAE (on)	MAPPO	DAE	COMA	QMIX	VDN
SMAX	3s5z_vs_3s6z	87.3 \pm 3.9	6.1 \pm 2.0	2.6 \pm 0.7	6.5 \pm 2.0	0.0 \pm 0.0	1.3 \pm 0.8	0.5 \pm 0.4
	5m_vs_6m	93.7 \pm 1.0	8.8 \pm 0.9	3.1 \pm 1.8	4.2 \pm 2.5	0.3 \pm 0.6	3.9 \pm 0.6	3.2 \pm 1.9
	10m_vs_11m	98.5 \pm 2.7	47.5 \pm 2.8	36.4 \pm 2.4	25.0 \pm 8.2	1.3 \pm 0.3	32.8 \pm 0.9	22.6 \pm 8.0
	6h_vs_8z	99.5 \pm 0.1	99.8 \pm 0.2	99.0 \pm 0.2	98.9 \pm 0.1	80.4 \pm 4.9	75.3 \pm 5.3	96.1 \pm 2.6
	smacv2_5_units	81.0 \pm 1.3	80.0 \pm 1.7	75.2 \pm 0.9	77.1 \pm 1.2	54.0 \pm 1.9	59.5 \pm 3.3	39.4 \pm 2.9
	smacv2_10_units	75.0 \pm 1.3	69.0 \pm 1.7	63.5 \pm 2.5	65.7 \pm 0.6	47.4 \pm 0.4	62.1 \pm 1.6	29.5 \pm 2.4
MABrax	halfcheetah-6x1	3463 \pm 68	3138 \pm 32	2965 \pm 45	2983 \pm 26	2017 \pm 39	N/A	N/A
	ant-8x1	3285 \pm 151	1557 \pm 82	1247 \pm 49	1279 \pm 40	901 \pm 96	N/A	N/A
	ant-4x2	3574 \pm 217	1666 \pm 32	1379 \pm 124	1441 \pm 50	950 \pm 27	N/A	N/A
	walker2d-6x1	912 \pm 18	697 \pm 19	489 \pm 77	591 \pm 25	242 \pm 64	N/A	N/A
	hopper-3x1	1572 \pm 110	1356 \pm 66	1088 \pm 65	1050 \pm 63	709 \pm 203	N/A	N/A
	humanoid-9 8	445 \pm 4	286 \pm 4	258 \pm 7	261 \pm 6	214 \pm 45	N/A	N/A

reuse. Importantly, GPAE shows its benefits even when restricted to on-policy samples only, already surpassing MAPPO and DAE. This highlights that the improvement is not solely due to sample reuse but also stems from the design of our estimator.

When extended with off-policy sample reuse, GPAE achieves further gains in both performance and sample efficiency. In Figures 4, which aggregate performance across all SMAX and MABrax tasks respectively, GPAE exhibits clearly steeper learning curves at early timesteps, indicating that it learns effective coordination policies with fewer samples. This advantage is particularly evident in the most challenging SMAX scenarios such as 3s5z_vs_3s6z and 5m_vs_6m, where credit assignment is nontrivial. While MAPPO and DAE degrade under these conditions, GPAE sustains more reliable coordination. By contrast, COMA shows clear weaknesses, with its one-step counterfactual structure failing to capture long-range dependencies, leading to consistently poor performances. QMIX and VDN demonstrate relatively steady but inferior results compared to MAPPO-based estimators, further highlighting that value decomposition is less suitable for these SMAX tasks.

Turning to the continuous-control benchmarks in MABrax, the gap between GPAE and other baselines remains significant. Value decomposition methods such as QMIX and VDN are not applicable in continuous domains, leaving MAPG estimators and COMA as the primary points of comparison. Here, GPAE again surpasses MAPPO and DAE, which, while competitive, struggle to maintain stability across high-dimensional action spaces. COMA proves highly unstable in this setting, similar to results from discrete tasks, further confirming the limitations of traditional counterfactual baselines based on TD(0) estimation.

Taken together, these findings highlight that GPAE consistently outperforms the baselines across diverse domains. Even without off-policy augmentation, GPAE shows improvements over the existing methods, while with sample reuse it achieves unmatched sample efficiency and performance. In both discrete and continuous

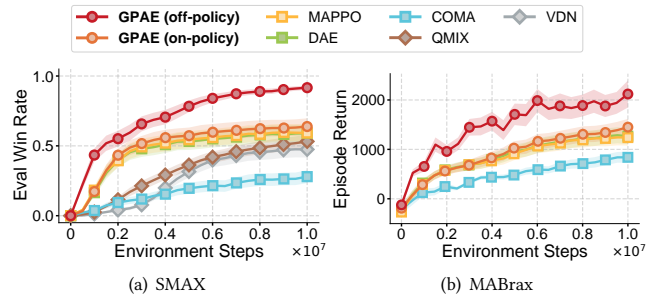


Figure 4: (a) and (b) represent the learning curve of the aggregated performance for all tasks on SMAX and MABrax, respectively.

domains, GPAE establishes itself as a unified framework that not only improves asymptotic performance but also accelerates learning, thereby validating its effectiveness as a solution for multi-agent credit assignment.

5.3 Ablation Study

DT-ISR Weight. We isolate the effect of the truncation scheme on GPAE by fixing the advantage estimator and varying only the off-policy weighting: no correction, single truncation (ST), individual truncation (IT), and our double truncation (DT-ISR). All variants employ the same off-policy sample reuse protocol described in Section 5. We focus on two SMAX scenarios, 5m_vs_6m and 3s5z/3s6z. As summarized in Table 2, these tasks exhibit the largest performance gaps between GPAE and other advantage estimators. The ablation is designed to verify how success is realized on the most demanding settings when off-policy data is reused. In particular, the analysis asks whether stability can be retained while preserving per-agent credit assignment.

Table 3 reports win rate performance after 10M timesteps. DT-ISR consistently outperforms other truncation schemes, achieving higher final performance. These results are consistent with the findings in Section 4.3, where DT-ISR improved off-policy learning stability across a broader set of tasks. The improvement here stems from the same mechanism: balancing stability and credit sensitivity by combining the strengths of joint and individual truncation.

Table 3: Ablation study on DT-ISR.

method	5m_vs_6m	3s5z_vs_3s6z
DT-ISR	93.7	87.3
ST-ISR	44.4	80.8
IT-ISR	58.6	83.7
No Correction	34.5	74.9

Table 4: Impact of η on DT-ISR weights.

η	5m_vs_6m		3s5z_vs_3s6z	
	win rate	Δc^i	win rate	Δc^i
1.0	95.2	-0.00154	84.5	-0.00105
1.056	93.7	-0.00409	87.3	-0.0067
1.1	93.1	-0.00553	83.2	-0.00918
1.15	39.4	-0.00561	80.3	-0.00938

Impact of η and Trace Dynamics. We analyze the impact of varying the double-truncation parameter η on key metrics, including learning performance in SMAX, measured by win rate. We also compute Δc^i as defined in Eq. 9, which measures the difference between the distance of truncation weights from the true joint ISR and individual ISR. The results are presented in Table 4, showing that DT-ISR is robust to changes in η , with performance remaining stable across a range of values. Specifically, performance is consistently good for η values between 1.0 and 1.05.

The ablation suggests that the performance gain arises from applying off-policy weights that integrate per-agent n -step advantages, control variance, and preserve credit assignment. DT-ISR supplies this coupling with a single hyperparameter η .

6 RELATED WORKS

Multi-Agent Policy Gradient. Multi-agent policy gradient is an effective approach under CTDE because it can exploit the global state to estimate the centralized critic. MADDPG [19] uses individual critic for each agent with global information. PS-TRPO [10] proposes parameter sharing for actor-critic to reduce the network size. COMA [8] estimates individual advantage with a counterfactual baseline, which enables explicit credit assignment. DOP [40] and FACMAC [25] propose off-policy policy gradient methods based on value decomposition. IPPO [5] and MAPPO [42] show that PPO performs strongly in the SMAC environment. HAPPO [16] proposed a sequential update structure of MAPPO, with individual

actor networks. Kuba et al. [17] analyzed the variance of MAPG, and proposed a baseline that minimizes the variance [17].

Multi-Agent Credit Assignment Problem. Credit assignment methods are commonly categorized into two approaches: *implicit* and *explicit* credit assignment. Implicit approaches such as VDN [34], QMIX [28], QTRAN [33], and several subsequent studies [14, 38] rely on value decomposition to implicitly infer individual contributions from the global Q-function. Explicit methods aim to assign credit directly, often through reward shaping or advantage decomposition. LIIR [6] learns intrinsic rewards based on individual state-action pairs. COMA [8] introduces a counterfactual baseline for computing per-agent advantages, while DAE [18] extends this idea by incorporating potential-based difference rewards into GAE [31]. Closely related, Castellini et al. [3] propose a principled formulation of policy gradients that leverage marginal contribution signals. Beyond these primary categories, methods such as MAAC [12] and intention sharing [13] employ attention mechanisms to enhance inter-agent coordination through structured representation learning.

Off-Policy Value Estimation. n -step estimation can provide more accurate value estimation for policy gradient methods. TD(λ) [35] and GAE [31] has been used widely for on-policy methods. In single-agent RL, n -step off-policy value estimation has been developed for improved sample efficiency. Retrace(λ) [21] and V-trace [7] propose low-variance value estimation by using truncated importance sampling trace. DISC [11] proposed using GAE-V (i.e., GAE with V-trace) for off-policy advantage estimation. GePPO [27] and Meng et al. [20] analyzes the reuse of off-policy samples with PPO. For the multi-agent value-based method, SMIX(λ) [41] extends QMIX by estimating the joint action value function based on n -step return and expected SARSA [36], where credit assignment is still implicit by value factorization.

7 CONCLUSION

One of the most important challenges in MARL is addressing the multi-agent credit assignment problem. While various approaches have attempted to tackle this issue, explicit study and analysis of credit assignment within policy gradient-based methods have been relatively underexplored. Our work directly addresses this gap by introducing a novel framework that accurately estimates per-agent advantages, enabling more effective policy updates. By leveraging our new value operator and integrating advanced techniques like double-truncated importance sampling weight, we provide a principled approach to credit assignment that enhances both on-policy and off-policy learning. With its solid theoretical foundation, GPAE paves the way for the development of more efficient, robust, and scalable multi-agent systems.

ACKNOWLEDGMENTS

This work was supported by Center for Applied Research in Artificial Intelligence (CARAI) Grant funded by Defense Acquisition Program Administration (DAPA) and Agency for Defense Development (ADD) of Republic of Korea (UD230017TD, 50%), and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2025-00557589, Generative Model Based Efficient Reinforcement Learning Algorithms for Multi-modal Expansion in Generalized Environments, 50%).

REFERENCES

- [1] Daniel S Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of operations research* 27, 4 (2002), 819–840.
- [2] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. 2018. JAX: composable transformations of Python+NumPy programs. (2018). <http://github.com/jax-ml/jax>
- [3] Jacopo Castellini, Sam Devlin, Frans A. Oliehoek, and Rahul Savani. 2022. Difference Rewards Policy Gradients. *Neural Computing and Applications Special Issue on Adaptive and Learning Agents 2021* (2022), 24.
- [4] Wubing Chen, Wenbin Li, Xiao Liu, Shangdong Yang, and Yang Gao. 2023. Learning Explicit Credit Assignment for Cooperative Multi-Agent Reinforcement Learning via Polarization Policy Gradient. *Proceedings of the AAAI Conference on Artificial Intelligence* 37 (2023), 11542–11550.
- [5] Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviy-chuk, Philip HS Torr, Mingfei Sun, and Shimon Whiteson. 2020. Is independent learning all you need in the starcraft multi-agent challenge? *arXiv preprint arXiv:2011.09533* (2020).
- [6] Yali Du, Lei Han, Meng Fang, Ji Liu, Tianhong Dai, and Dacheng Tao. 2019. LIIR: Learning individual intrinsic reward in multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 32 (2019).
- [7] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. 2018. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International Conference on Machine Learning*. PMLR, 1407–1416.
- [8] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
- [9] C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. 2021. *Brax - A Differentiable Physics Engine for Large Scale Rigid Body Simulation*. <http://github.com/google/brax>
- [10] Jayesh K Gupta, Maxim Egorov, and Mykel Kochenderfer. 2017. Cooperative multi-agent control using deep reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*. Springer, 66–83.
- [11] Seungyul Han and Youngchul Sung. 2019. Dimension-wise importance sampling weight clipping for sample-efficient reinforcement learning. In *International Conference on Machine Learning*. PMLR, 2586–2595.
- [12] Shariq Iqbal and Fei Sha. 2019. Actor-attention-critic for multi-agent reinforcement learning. In *International Conference on Machine Learning*. PMLR, 2961–2970.
- [13] Woojun Kim, Jongeui Park, and Youngchul Sung. 2020. Communication in multi-agent reinforcement learning: Intention sharing. In *International Conference on Learning Representations*.
- [14] Woojun Kim and Youngchul Sung. 2023. An Adaptive Entropy-Regularization Framework for Multi-Agent Reinforcement Learning. In *Proceedings of the 40th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 202)*. PMLR, 16829–16852.
- [15] Landon Kraemer and Bikramjit Banerjee. 2016. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing* 190 (2016), 82–94.
- [16] Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. 2022. Trust Region Policy Optimisation in Multi-Agent Reinforcement Learning. In *International Conference on Learning Representations*.
- [17] Jakub Grudzien Kuba, Muning Wen, Linghui Meng, Haifeng Zhang, David Mguni, Jun Wang, Yaodong Yang, et al. 2021. Settling the variance of multi-agent policy gradients. *Advances in Neural Information Processing Systems* 34 (2021), 13458–13470.
- [18] Yueheng Li, Guangming Xie, and Zongqing Lu. 2022. Difference Advantage Estimation for Multi-Agent Policy Gradients. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, 13066–13085.
- [19] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 6382–6393.
- [20] Wenjia Meng, Qian Zheng, Gang Pan, and Yilong Yin. 2023. Off-policy proximal policy optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 9162–9170.
- [21] Rémi Munos, Tom Stepleton, Anna Harutyunyan, and Marc Bellemare. 2016. Safe and efficient off-policy reinforcement learning. *Advances in Neural Information Processing Systems* 29 (2016).
- [22] Andrew Y. Ng, Daishi Harada, and Stuart J. Russell. 1999. Policy Invariance Under Reward Transformations: Theory and Application to Reward Shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*.
- [23] Frans A Oliehoek. 2012. Decentralized POMDPs. In *Reinforcement Learning*. Springer, 471–503.
- [24] Afshin Oroojlooy and Davood Hajinezhad. 2023. A review of cooperative multi-agent deep reinforcement learning. *Applied Intelligence* 53, 11 (2023), 13677–13722.
- [25] Bei Peng, Tabish Rashid, Christian Schroeder de Witt, Pierre-Alexandre Kamieny, 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.
- [26] Doina Precup, Richard S. Sutton, and Satinder P. Singh. 2000. Eligibility Traces for Off-Policy Policy Evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*.
- [27] James Queeney, Yannis Paschalidis, and Christos G Cassandras. 2021. Generalized proximal policy optimization with sample reuse. *Advances in Neural Information Processing Systems* 34 (2021), 11909–11919.
- [28] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2018. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*. PMLR, 4295–4304.
- [29] Alexander Rutherford, Benjamin Ellis, Matteo Gallici, Jonathan Cook, Andrei Lupu, Gardhar Ingvarsson, Timon Willi, Ravi Hammond, Akbir Khan, Christian Schroeder de Witt, Alexandra Souly, Saptarashmi Bandyopadhyay, Mikayel Samvelyan, Minqi Jiang, Robert Tjarko Lange, Shimon Whiteson, Bruno Lacerda, Nick Hawes, Tim Rocktäschel, Chris Lu, and Jakob Nicolaus Foerster. 2024. Jax-MARL: Multi-Agent RL Environments and Algorithms in JAX. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- [30] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philip H. S. Torr, Jakob Foerster, and Shimon Whiteson. 2019. The StarCraft Multi-Agent Challenge. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*.
- [31] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2016. High-Dimensional Continuous Control Using Generalized Advantage Estimation. In *International Conference on Learning Representations*.
- [32] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).
- [33] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. 2019. QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*. PMLR, 5887–5896.
- [34] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. 2018. Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward. 2085–2087.
- [35] Richard S Sutton. 1988. Learning to predict by the methods of temporal differences. *Machine Learning* 3, 1 (1988), 9–44.
- [36] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT Press.
- [37] Ming Tan. 1993. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the Tenth International Conference on Machine Learning*. 330–337.
- [38] Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. 2021. QPLEX: Duplex Dueling Multi-Agent Q-Learning. In *International Conference on Learning Representations*.
- [39] Jiangxing Wang, Deheng Ye, and Zongqing Lu. 2023. More Centralized Training, Still Decentralized Execution: Multi-Agent Conditional Policy Factorization. In *The Eleventh International Conference on Learning Representations*.
- [40] Yihan Wang, Beining Han, Tonghan Wang, Heng Dong, and Chongjie Zhang. 2020. DOP: Off-policy multi-agent decomposed policy gradients. In *International Conference on Learning Representations*.
- [41] Xinghu Yao, Chao Wen, Yuhui Wang, and Xiaoyang Tan. 2021. Smix (λ): Enhancing centralized value functions for cooperative multiagent reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems* (2021).
- [42] Chao Yu, Akash Velu, Eugene Vinitzky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. 2022. The surprising effectiveness of PPO in cooperative multi-agent games. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*. Article 1787.
- [43] Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. 2021. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control* (2021), 321–384.
- [44] Meng Zhou, Ziyu Liu, Pengwei Sui, Yixuan Li, and Yuk Ying Chung. 2020. Learning implicit credit assignment for cooperative multi-agent reinforcement learning. *Advances in neural information processing systems* 33 (2020), 11853–11864.