An Axiomatic Framework for Reinforcement Learning in Open Multi-Agent Systems

Jianhong Wang*
INFORMED-AI Hub
University of Bristol

Yang Li*
School of Computer Science
Shanghai Jiao Tong University

Samuel Kaski†
ELLIS Institute Finland

Jonathan Lawry
School of Engineering Mathematics and Technology
University of Bristol

Abstract

Open multi-agent systems, where only a subset of agents are controllable, are central to real-world domains such as smart grids, yet lack a principled foundation for reinforcement learning algorithm design. We study this setting through n-agent ad hoc teamwork (NAHT), a recent framework for learning in open multi-agent environments. We propose an axiomatic framework that derives value function learning from the Shapley axioms (Efficiency, Symmetry and Linearity), treating them as structural constraints rather than heuristic design choices. This formulation builds on a novel NAHT decomposition, which characterises the fundamental difference between open and fully controllable multi-agent systems, and establishes a cooperative game-theoretic foundation for learning. We further show that enforcing these axioms on learning individual value functions recovers the Shapley value for Dec-POMDPs, providing a principled bridge between reinforcement learning and classical cooperative game theory. Building on this framework, we introduce Shapley Machines and Banzhaf Machines, two generic algorithmic templates that can be instantiated with standard reinforcement learning methods. When applied to strong base algorithms such as IPPO and POAM, our methods consistently improve learning efficiency and generalisation to unseen agent types and conventions across NAHT benchmark tasks. The code of experiments is published on `https://github.com/hsvgbkhgbv/axiomatic-rl-naht`.

1 Introduction

Multi-agent systems (MAS) have become a central paradigm for modelling real-world domains such as smart grids [Wang et al., 2021], railway networks [Zhang et al., 2024], and swarm robotics [Nayak et al., 2023, Li et al., 2025], where multiple agents must coordinate under uncertainty. Reinforcement learning (RL) [Sutton and Barto, 2018], and in particular cooperative multi-agent reinforcement learning (CMARL), has shown promise in solving such problems. However, existing CMARL methods typically assume that all agents are controllable and that team composition is fixed, which we call the closed multi-agent systems; these assumptions rarely hold in practice.

In many real-world scenarios, agents must operate in open multi-agent systems, where only a subset of agents is controllable and the remaining agents may be unknown or dynamically changing. This setting introduces two fundamental challenges: (i) generalisation to previously unseen teammate

*Correspondence to `jianhong.wang@bristol.ac.uk` and `yang.li.cs@sjtu.edu.cn`. †Samuel Kaski is also with Aalto University and University of Manchester.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

behaviours, and (ii) openness to varying team compositions. To address these challenges, recent work has introduced the n-agent ad hoc teamwork (NAHT) [Wang et al., 2025], which extends CMARL to settings with partially controllable and dynamically formed teams.

Although the NAHT objective is to maximise the team value over varying team compositions, this objective does not specify how agents can form cooperation in an open system with both controlled and uncontrolled agents. From the perspective of CMARL, an agent is represented by its individual value. This raises a fundamental question:

How should individual value functions be structured and learned to support cooperation in open multi-agent systems?

In this paper, we address this question from the perspective of cooperative game theory. Specifically, we introduce an NAHT decomposition that represents a team value by two parts: internal cooperation among controlled agents and external teamwork involving uncontrolled agents. This decomposition reveals a structural distinction between open and closed multi-agent systems, which is not innately captured by the NAHT objective.

Building on this structure, we propose an axiomatic framework to derive individual value functions, grounded in the axioms of the Shapley value [Shapley, 1953, Dubey, 1975]. Rather than constructing individual value functions directly by the explicit form of the Shapley value, we derive the individual value function learning by enforcing its axioms (Efficiency, Symmetry and Linearity) as structural constraints. This yields a principled approach to designing learning algorithms, where individual value functions are shaped to satisfy some properties, such as cooperative game-theoretic properties applied in this paper. Notably, we prove in theory that enforcing these axioms on learning individual value functions recovers the Shapley value for Dec-POMDPs [Oliehoek et al., 2016], while relaxing specific axioms leads to alternative payoff allocations, such as the Banzhaf index.

More importantly, our axiomatic framework also reveals a novel connection between cooperative game theory and reinforcement learning. In particular, we show that the Linearity axiom induces a structure equivalent to the truncated TD(λ) [Cichosz, 1994], where the number of n-step return components can be estimated by the number of non-empty coalition basis games in cooperative game theory. This provides a theoretical explanation for the role of n-step returns and suggests a new perspective on the design of temporal-difference learning methods in the NAHT (see Section 5.2).

NAHT includes two extreme cases: single-agent RL with an environment of other uncontrolled agents, and CMARL with fully controllable agents. The above result therefore extends to both settings.

Based on this axiomatic framework, we introduce two generic algorithmic templates: Shapley Machines, which enforce all axioms, and Banzhaf Machines, which relax the Efficiency axiom. These templates can be instantiated with existing RL algorithms, such as IPPO [De Witt et al., 2020] and POAM, to yield practical methods for solving the NAHT.

We evaluate the proposed methods on modified MPE and SMAC benchmarks adapted to the NAHT setting [Wang et al., 2025]. Our results show that the proposed framework consistently improves learning efficiency and generalisation to unseen agent types and conventions. Furthermore, we demonstrate that relaxing specific axioms can lead to improved performance in certain regimes, highlighting the flexibility of the proposed axiomatic perspective. Due to the limit of space, we leave the related work section to Appendix A.

2 Background

2.1 N-Agent Ad Hoc Teamwork

In this section, we describe a problem setting for open multi-agent systems, referred to as n-agent ad hoc teamwork (NAHT) [Wang et al., 2025]. The main challenges of NAHT are as follows: (1) cooperating with potentially unknown types of teammates (generalization), and (2) coping with a varying number of uncontrolled teammates (openness). A decentralized partially observable Markov decision process (Dec-POMDP) [Oliehoek et al., 2016] is considered to formalize the problem. There is a team of agents $\mathcal{M}$ whose size is denoted as M, a state space $\mathcal{S}$, a joint action space $\mathcal{A} = \times_{i \in \mathcal{M}} \mathcal{A}_i$, a per-agent observation space $\mathcal{O}_i$, a transition function $P_T : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, a team reward function $R_t : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, a discount factor $\gamma \in [0, 1]$ and an episode length $T \in \mathbb{Z}^+$. Each agent receives

observations via an observation function $O_i : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{O}_i)$. $\mathcal{H}_i$ is defined as an agent i's space of observation and action histories. Each agent is equipped with a policy $\pi_i : \mathcal{H}_i \to \Delta(\mathcal{A}_i)$.

In this problem, there are two groups of agents: a set of controlled agents denoted as $\mathcal{C}$ whose size is denoted as N, and a set of uncontrolled agents denoted as $\mathcal{U}$ whose size is denoted as $M - N$. As NAHT is an open multi-agent system, at the beginning of a multi-agent interaction process, a team of agents $\mathcal{M}$ are randomly drawn from the controlled agent pool and the uncontrolled agent pool. For conciseness, each controlled agent is characterised by its policy, and the set of controlled agents can be expressed as $\mathcal{C}(\theta) = \{\pi_i^\theta\}_{i=1}^N$. The above sampling procedure is denoted as $X(\mathcal{U}, \mathcal{C}(\theta))$. The aim of NAHT is learning parameters θ to solve the following optimization problem:

$$\max_\theta \mathbb{E}_{\pi^{(\mathcal{M})} \sim X(\mathcal{U}, \mathcal{C}(\theta))} \left[\sum_{t=0}^{T} \gamma^t R_t \right], \tag{1}$$

where $\pi^{(\mathcal{M})}$ denotes the joint policy of a team of agents $\mathcal{M}$. Note that different number of controlled agents N would lead to various teams of agents $\mathcal{M}$; the expectation in Eq. 1 is usually approximated by a team value which we will specify in the following section.

2.2 Cooperative Games and Shapley Value

Representation of Cooperative Games. A cooperative game is specified by a pair $(\mathcal{M}, v)$, where $\mathcal{M}$ is the set of agents and $v : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ is its characteristic function. When the agent set is fixed, we identify the game with its characteristic function v for brevity. An arbitrary cooperative game v can be uniquely represented by a set of basis games $\{v_C^1 \mid \emptyset \neq C \subseteq \mathcal{M}\}$ [Dubey, 1975], as delineated in Definition 2.1. In this paper, we focus on investigating how cooperative games are approximated over a team of agents $\mathcal{M}$. Therefore, we only consider the function values $v(\mathcal{M})$ and $v_C^1(\mathcal{M})$ for a team of agents $\mathcal{M}$ in the presentation of this work.[2] As per previous work in the MARL [Wang et al., 2020], we consider the superadditive game as the game class of $\mathcal{G}$, which is suitable for a cooperative multi-agent task, such as NAHT (see Appendix B.2 for more details).

Definition 2.1 (**Cooperative Games** [Dubey, 1975])**.** *We define a function $v_C^z(D)$ for an arbitrary $z \in \mathbb{R}$ such that if a coalition $C \subseteq D$, $v_C^z(D) = z$, otherwise, $v_C^z(D) = 0$. For a cooperative game described by a characteristic function $v : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$, we can describe a set of such games over a team of agents $\mathcal{M}$ by a set of basis games $\{v_C^1 \mid \emptyset \neq C \subseteq \mathcal{M}\}$, such that $\mathcal{G} = \{w \mid w = \sum k_C v_C^1, k_C \in \mathbb{R}, \emptyset \neq C \subseteq \mathcal{M}\}$.*[3]

Shapley Value. Shapley value is a multidimensional linear transformation $\phi : \mathcal{G} \to \mathbb{R}^M$ defined on the set of cooperative game $\mathcal{G}$, given that there are M agents in total. Each dimension of ϕ indicates the payoff allocation to an agent. It has been proved that Shapley value is a unique payoff allocation function on the set of cooperative games $\mathcal{G}$ which satisfies the following axioms: Efficiency, Symmetry and Additivity [Dubey, 1975],[4] highlighted in the following Theorem 2.2:

Theorem 2.2 (**Axioms of Shapley Value** [Dubey, 1975])**.** *Shapley value is a unique payoff allocation function on the cooperative game space $\mathcal{G}$, satisfying Efficiency, Symmetry and Additivity.*

3 N-Agent Ad Hoc Teamwork Decomposition

To understand how learning differs in open multi-agent systems, we first characterise the cooperative-game structure of team values of the NAHT. Recall that in the NAHT a team consists of two groups of agents: a set of controlled agents $\mathcal{C}$, whose policies are optimized, and a set of uncontrolled agents $\mathcal{U}$, whose behaviours are unknown and may vary across episodes. Unlike the closed multi-agent systems, the performance of a team in the NAHT depends not only on coordination among controlled agents, but also on their ability to adapt to diverse and potentially unseen teammates. To make this distinction explicit, we represent a team value at a given state in the NAHT as a state-specific cooperative game

[2] In theory, $v(D)$ and $v_C^1(D)$ for any coalition $D \subset \mathcal{M}$ exist. As they are irrelevant to the main result that we would like to show, we ignore them in this paper.

[3] The analytic form of k_C is represented as $k_C = \sum_{T \subseteq C} (-1)^{|C|-|T|} v(T)$.

[4] For an arbitrary characteristic function game, Shapley value is uniquely determined by the Efficiency, Additivity, Symmetry, and Dummy Player axioms [Chalkiadakis et al., 2011][Chap. 2]. Since the Dummy Player axiom is embedded in the basis game construction of Dubey [1975], it has been naturally satisfied.

value that is the weighted sum of non-empty coalition basis game values, from the perspective of cooperative game theory, as follows:

Definition 3.1 (**State-Specific Cooperative Games**). *We define a function $D \mapsto v^z_{C,s}(D)$, for any real value $z \in \mathbb{R}$ and state $s \in \mathcal{S}$, such that if a coalition $C \subseteq D$, $v^z_{C,s}(D) = z$, otherwise, $v^z_{C,s}(D) = 0$. It is not difficult to observe that we have $v^z_{C,s} = z v^1_{C,s}$, where $v^1_{C,s}$ is a state-specific basis game for state s. Similarly, a set of generalised cooperative games to state s, referred to as state-specific cooperative games, is represented as $\mathcal{G}(s) = \{w_s \mid w_s = \sum k_C v^1_{C,s}, k_C \in \mathbb{R}, \emptyset \neq C \subseteq \mathcal{M}\}$. In other words, an arbitrary state-specific game $v_s \in \mathcal{G}(s)$ can be represented as $v_s = \sum k_C v^1_{C,s}$.*

Definition 3.2 (**Game Structure for NAHT**). *$\mathcal{G}_{NAHT} = \times_{s \in \mathcal{S}} \mathcal{G}(s)$ denotes the set of all NAHT processes, where a team value of an NAHT process at state s is represented as a state-specific cooperative game value $v_s(\mathcal{M})$, where $v_s \in \mathcal{G}(s)$. Each element belonging to $\mathcal{G}_{NAHT}$ is a tuple of state-specific cooperative games, defined by a team of agents $\mathcal{M}$, a state space $\mathcal{S}$, and weightings k_C associated with the state-specific basis games of all non-empty coalitions $\emptyset \neq C \subseteq \mathcal{M}$.*

Based on generalisation of the cooperative game representation from the classic cooperative game as shown in Definition 3.1 and the game structure for the NAHT in Definition 3.2, we define a state-specific cooperative game value $v_s(\mathcal{M})$, as the team value of the NAHT at state s, as follows:

$$v_s(\mathcal{M}) = \underbrace{\sum_{C_I \in \mathcal{P}_+(\mathcal{C})} k_{C_I} v^1_{C_I,s}(\mathcal{M})}_{\textbf{Internal Cooperation}} + \underbrace{\sum_{C_E \in \mathcal{P}_+(\mathcal{M}) \setminus \mathcal{P}_+(\mathcal{C})} k_{C_E} v^1_{C_E,s}(\mathcal{M})}_{\textbf{External Teamwork}}, \tag{2}$$

where $\mathcal{M} = \mathcal{C} \cup \mathcal{U}$ denotes the full team, $\mathcal{P}_+(\cdot)$ indicates the powerset of a set excluding the empty set, and each term corresponds to a contribution from a subset (coalition) of agents. The first term captures internal cooperation, i.e., interactions solely among controlled agents $\mathcal{C}$. This corresponds to the standard task in the CMARL, where all agents are optimised jointly. The second term captures external teamwork, i.e., interactions between controlled agents $\mathcal{C}$ and uncontrolled agents $\mathcal{U}$, along with interactions among uncontrolled agents $\mathcal{U}$. The second term is unique to open multi-agent systems, and reflects the need to coordinate with agents whose behaviours are not controlled and may vary across episodes. When all agents are controlled, i.e., $\mathcal{C} = \mathcal{M}$, the external teamwork part vanishes, and the decomposition reduces to internal cooperation, recovering the fully controllable setting of the CMARL.

Example 3.3. *As shown in Figure 1, suppose we have a team of agents $\mathcal{M} = \{1, 2\}$ for a state s, where $\mathcal{C} = \{1\}$ and $\mathcal{U} = \{2\}$. θ is the angle between two vectors representing two situations, such that $w_s \in W = span\{v^1_{\{1\},s}\}$ and $u_s \in \mathcal{G}(s) = span\{v^1_{\{1\},s}, v^1_{\{2\},s}, v^1_{\{1,2\},s}\}$. It is obvious that $w_s(\mathcal{M})$, as a projection of $u_s(\mathcal{M})$, loses information about the external teamwork.*

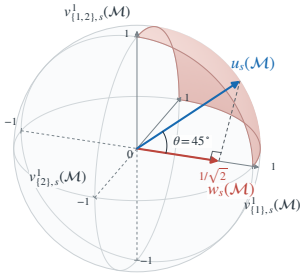


Figure 1: Geometric illustration of the NAHT decomposition. The red and blue vectors are two games represented by different basis game sets, denoted by $w_s(\mathcal{M})$ and $u_s(\mathcal{M})$. Their difference is measured by the θ. The shading area in red indicates the range of generated cooperative games with that $k_C \geq 0$ and $\sum (k_C)^2 = 1$.

Implication. This decomposition highlights a fundamental difference between NAHT and CMARL: In the CMARL, all agents are controllable, and learning focuses exclusively on internal cooperation. In the NAHT, learning must additionally account for external teamwork with uncontrolled agents. Ignoring the external component effectively reduces the problem to a closed multi-agent system, which introduces bias and limits generalisation to unseen team compositions. The NAHT decomposition therefore provides a structural characterisation of openness, and it will serve as the foundation for the axiomatic framework introduced as follows.

4 From Shapley Axioms to Learning Algorithms

Building on the NAHT decomposition, we now derive an axiomatic framework for learning in open multi-agent systems. Our approach is based on the axioms of the Shapley value, which we interpret as structural constraints on individual value functions. In detail, we aim to show that how the individual values, as the output of a multidimensional linear transformation ϕ applied to a state-specific cooperative game v_s, satisfy the

Shapley value's axioms in learning. When we enforce all these axioms together, the Shapley-form individual values are uniquely shaped (see Theorem 4.7). Moreover, the axioms can be directly translated into algorithmic design principles: (1) The Efficiency axiom corresponds to reward shaping; (2) The Linearity axiom corresponds to truncated TD(λ)-style updates; (3) The Symmetry axiom corresponds to individual value function structures. These principles will later lead to two generic algorithmic templates, referred to as Shapley Machine and Banzhaf Machine.

4.1 From Game Structure to Dec-POMDPs

The game structure for the NAHT in Definition 3.2 can be used to represent any sequential decision-making process evaluated in states. However, it does not necessarily imply that the process embedded in the game structure strictly follows the problem setting of the NAHT — Dec-POMDPs [Oliehoek et al., 2016]. To bridge this gap, we augment the game structure with Markov dynamics and per-step team rewards to link the values of different states. When the state-specific cooperative game value for each state s_t is represented as $v_{s_t}(\mathcal{M}) = \mathbb{E}_\pi[\sum_{\tau=0}^{T-t-1} \gamma^\tau R_{t+\tau} \mid s_t]$, we can derive the expected n-step return in RL [Sutton and Barto, 2018][Chap. 7], with a given horizon n to link $v_{s_t}(\mathcal{M})$ and $v_{s_{t+n}}(\mathcal{M})$, such that

$$v_{s_t}(\mathcal{M}) = \mathbb{E}_\pi \left[\sum_{\tau=0}^{n-1} \gamma^\tau R_{t+\tau} + \gamma^n v_{s_{t+n}}(\mathcal{M}) \,\middle|\, s_t \right] := \mathbb{E}_\pi \left[G_{t:t+n} \mid s_t \right]. \tag{3}$$

For ease of presentation, we denote the state-specific basis game value $v_{C,s}^{v_s(\mathcal{M})}(\mathcal{M})$ by $V(C, s)$.

Proposition 4.1. *By Definition D.2 and a horizon assignment to match coalition and n-step return orderings, for every state $s \in \mathcal{S}$,*

$$\big(V(C_1, s), \ldots, V(C_m, s)\big) \;=\; \big(\mathbb{E}_\pi[G_{t:t+1} \mid s], \ldots, \mathbb{E}_\pi[G_{t:t+m} \mid s]\big),$$

and, each $V(C_i, s)$ is equal to $v_s(\mathcal{M})$ in value.

By Proposition 4.1 (see the proof in Appendix D), we obtain the following formula that aligns the values of state-specific basis games of various coalitions to the expected n-step returns with the corresponding horizons $n \in \{1, ..., m\}$ as their representation, for all $\emptyset \neq C_n \subseteq C_{n+1}, C_m = \mathcal{M}$, such that

$$V(C_n, s_t) = \mathbb{E}_\pi \left[\sum_{\tau=0}^{n-1} \gamma^\tau R_{t+\tau} + \gamma^n v_{s_{t+n}}(\mathcal{M}) \,\middle|\, s_t \right]. \tag{4}$$

Specifically, the sampled n-step return $G_{t:t+n}$ is an unbiased estimator of $V(C_n, s)$ conditional on $s_t = s$. Under the assumptions in Appendix E, the variance of the state-specific basis game value estimators is non-decreasing in coalition size, while n-step return variance is non-decreasing in horizon. This correspondence motivates the horizon assignment in Eq. (4).

4.2 Linearity Axiom and Temporal-Difference Learning

The Linearity axiom plays a central role in connecting cooperative game theory to temporal-difference learning. We show this in the following Proposition 4.2 (see the proof in Appendix F):

Proposition 4.2. *Given the expression of state-specific basis game values in Eq.* (4)*, applying the Linearity axiom to state-specific cooperative games of Eq.* (2) *can lead to the following equation:*

$$\phi(v_{s_t}) = \mathbb{E}_\pi \left[\sum_{n=1}^{m} k'_{C_n} \cdot \sum_{\tau=0}^{n-1} \gamma^\tau \phi(R_{t+\tau}) + \gamma^n \phi(v_{s_{t+n}}) \,\middle|\, s_t \right]. \tag{5}$$

In Proposition 4.2, the term within the $\mathbb{E}_\pi[\cdot]$ takes the form of truncated λ-return [Sutton and Barto, 2018][Chap. 12], when all k'_{C_n} are seen as weightings for the n-step return components.

In practice, we only need to learn $\phi_i(v_{s_t})$ directly for each controlled agent $i \in \mathcal{C}$. For brevity, we express $\phi_i(v_{s_t})$ as $V_i(s_t)$ in the following paragraphs. Based on Eq. (5), we derive the TD error $\delta_{t,i}$ of truncated TD(λ) (TTD(λ)) [Sutton and Barto, 2018][Chap. 12] for each agent i purely from the game structure for the NAHT, such that

$$\delta_{t,i} := \sum_{n=1}^{m} k'_{C_n} \cdot \sum_{\tau=0}^{n-1} \left(\gamma^\tau \phi_i(R_{t+\tau}) + \gamma^n V_i(s_{t+n}) \right) - V_i(s_t). \tag{6}$$

Remark 4.3. *The above result provides a theoretical explanation for the role of the n-step return in the NAHT; it gives an insight that the truncated TD(λ) learning arises naturally from the Linearity axiom and the NAHT decomposition to capture both internal cooperation and external teamwork, given an appropriate horizon.*

4.3 Efficiency Axiom and Reward Shaping

Note that $\phi(R_{t+\tau})$ in Eq. (6) has not yet been defined so far, because $R_{t+\tau}$ is not belonging to the state-specific cooperative game space $\mathcal{G}(s)$. As a result, it is necessary to represent $\phi(R_{t+\tau})$ in the form of $\phi(v_s)$, where $v_s \in \mathcal{G}(s)$ is the underlying state-specific cooperative game. The Efficiency axiom can address this problem. To satisfy the Efficiency axiom such that $\sum_{i=1}^{M} \phi_i(v_{s_t}) = v_{s_t}(\mathcal{M})$, we reasonably presume that its sufficient condition holds: $\sum_{i=1}^{M} \phi_i(R_t) = R_t$. Then, we demonstrate that how $\phi_i(R_t)$ is represented by the Efficiency axiom in the following Proposition 4.4 (see the proof in Appendix G):

Proposition 4.4. *Given the condition $\sum_{i=1}^{M} \phi_i(R_t) = R_t$, each agent's payoff allocation defined on rewards R_t, i.e., $\phi_i(R_t)$, can be expressed as*

$$\phi_i(R_t) := R_t - \sum_{j \neq i} \left(\phi_j(v_{s_t}) - \gamma \phi_j(v_{s_{t+1}}) \right). \tag{7}$$

By replacing $\phi_j(v_{s_t})$ with $V_j(s_t)$ as there is in Eq. (6), we get that

$$\phi_i(R_t) = R_t - \sum_{j \neq i} \left(V_j(s_t) - \gamma V_j(s_{t+1}) \right). \tag{8}$$

The condition $\sum_{i=1}^{M} \phi_i(R_t) = R_t$ for deriving the Efficiency axiom is also referred to as additively factored immediate reward function. The following Theorem 4.5 shows that the Efficiency axiom always exists in the Dec-POMDP, given a factorization scheme of the transition function.

Theorem 4.5 (**Principle of Decomposing Value Functions** [Oliehoek et al., 2016])**.** *Given an additively factored reward function, for any timestep t there is a factorization of the transition function, such that the value of a finite-horizon factored Dec-POMDP is decomposable across agents.*

Since NAHT is considered in Dec-POMDPs (see Section 2.1), to guarantee the feasibility of the Efficiency axiom as shown in Theorem 4.5, another constraint is added to implicitly facilitate searching for a factorization of the transition function, such that

$$\sum_{i=1}^{M} V_i(s_t) = v_{s_t}(\mathcal{M}). \tag{9}$$

4.4 Symmetry Axiom and Value Function Structure

The Symmetry axiom here means that two agents who contribute equally to every possible coalition (excluding the ones involving themselves) should receive equal payoff allocations [Chalkiadakis et al., 2011][Chap. 2]. In the literature from Dubey [1975], it was described as the permutation-equivariance of an multidimensional operator ϕ defined on a cooperative game, as shown in the following Theorem 4.6 (see the proof in Appendix G):

Theorem 4.6. *The payoff allocation operator satisfying permutation-equivariance is a sufficient condition for the Symmetry axiom.*

Next, we aim to show how to construct such an operator to fulfil the permutation-equivariance, such that $\phi_{\sigma(i)}(\sigma(v_s)) = \phi_i(v_s)$ can be constructed in practice, where σ means relabelling agent IDs to another ordering, e.g., $(1, 2) \mapsto (2, 1)$. Recall that $\phi_i(v_s)$ is expressed as $V_i(s)$. In plain words, this means that relabelling the agents should simply relabel their individual values, which can be implemented by the following two conditions: (1) The operator ϕ's output is changeable consistently with reordering agent IDs, e.g., $(1, 2) \mapsto (2, 1) \Rightarrow (V_1(s), V_2(s)) \mapsto (V_2(s), V_1(s))$; (2) The operator's output with respect to each agent should be invariant to reordering agent IDs on its input `inp` in practice (e.g., the world information s and/or other agents' information), e.g., $V_i(\sigma(\texttt{inp})) = V_i(\texttt{inp})$. Note that the second condition usually holds when other agents' information is not directly included in the input to an agent's individual value function, such as POAM [Wang et al., 2025] and IPPO [De Witt et al., 2020].

4.5 From Shapley Axioms to Learning Algorithms

The algorithm that implements V_i as agents' individual values to fulfil the Efficiency, Linearity and Symmetry axioms through learning is named as **Shapley Machine**: (1) Shape individual rewards $\phi_i(R)$ following Eq. (8) and set Eq. (9) as a regularization term, to fulfil the Efficiency axiom. (2) Implement the TTD(λ) following Eq. (6), to fulfil the Linearity axiom. The $k'_C > 0$[5] is implemented as geometric distribution following the convention of TTD(λ) [Sutton and Barto, 2018][Chap. 12]. (3) Structure the policy and individual value networks to fulfil the Symmetry axiom.

Since the Linearity axiom is a sufficient condition for the Additivity axiom (see Appendix B.3), the individual values satisfying all those axioms in implementation realizes Shapley values for the Dec-POMDPs, as highlighted in the following Theorem 4.7 (see the proof in Appendix G):

Theorem 4.7. *Shapley Machine is an algorithm enforcing V_i to fulfil Efficiency, Symmetry and Linearity, so the V_i is the Shapley value for Dec-POMDPs.*

To realize the Shapley Machine, we use two base algorithms structurally satisfying the Symmetry axiom: POAM [Wang et al., 2025] and IPPO [De Witt et al., 2020]. We then shape the objective functions following conditions imposed by other two axioms, to obtain the algorithms **SM-POAM** and **SM-IPPO**, respectively. The details of implementation are left to Appendix H.1.

Following the same principle of designing algorithms via axiomatic characterization, we also propose a variant that enforces the Linearity and Symmetry axioms while omitting Efficiency.[6] Since the resulting V_i aligns with the Banzhaf index, we call this algorithm **Banzhaf Machine**. Analogous to the implementation of Shapley Machine, its instantiation on POAM and IPPO are named as **BM-POAM** and **BM-IPPO**, respectively.

Remark 4.8. *Banzhaf Machine is an algorithm generating V_i, fulfilling Symmetry and Linearity.*

5 Experiments

In experiments, we evaluate the proposed SM-POAM, SM-IPPO, BM-POAM and BM-IPPO on the modified MPE-PP and SMAC benchmark tasks tailored to the NAHT settings [Wang et al., 2025], against the counterpart baseline algorithms POAM and IPPO. The uncontrolled agents are taken from five pre-trained policies, which were trained in fully controllable CMARL settings using the algorithms IPPO, MAPPO, VDN, QMIX, and IQL. To avoid confusion between the standard IPPO algorithm (trained under the CMARL settings) and the version trained within the NAHT process (where only a subset of agents is controllable), we refer to the latter as IPPO-NAHT.

We validate the game structure for the NAHT, a core concept in our theory, by setting the number of n-step return components considered in TTD(λ) using the number of basis games m. Since the number of basis games is far more than the preset episode length (511 vs. 150) in 10v11, we approximate the number of basis games to the episode length as a trade-off. About the 8v9 scenario, the number of basis games needed in theory is close to the episode length (127 vs. 120), so the number of basis games can be naturally approximated by the episode length. As a result, for 8v9 and 10v11 Banzhaf Machine is equivalent to the baseline counterparts in implementation. The details of baselines, experimental settings and evaluation metrics are provided in Appendix H. All results are obtained by first computing each metric with 128 episodes and then averaging these per-seed metrics across 5 random seeds, with 95% confidence intervals reported.

Research Questions. We focus on the following research questions in experimentation: (1) Is the number of coalitions m used to decide the horizon of TTD(λ) (see Section 4.2) a reasonable choice? (2) Are all Shapley value's axioms equally appropriate across scenarios? (3) Which axiom is key to generalization to unseen conventions or unseen agent types?

5.1 Main Results and Analysis

Answer to Question 1. We conduct experiments with varying numbers of basis games m using SM-POAM as the candidate algorithm. As shown in Figure 2, learning performance in MPE-PP and 8v9 is sensitive to the choice of m, while in 5v6 it is comparatively robust. Overall, these results

[5] See Remark C.3 for why $k'_C > 0$ is chosen.

[6] Without the Efficiency axiom as a constraint on $\phi(R_t)$, Banzhaf Machine uses $\phi_i(R_t) = R_t$ as one implementation.

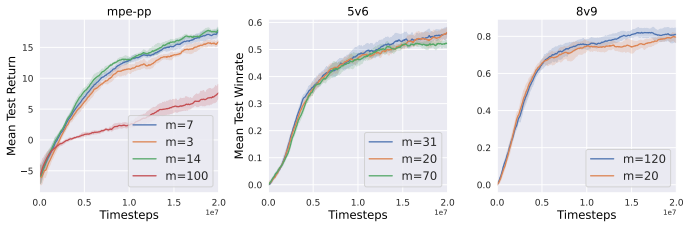


Figure 2: Experiments verifying that the number of n-step return components in TTD(λ) can be determined by the number of non-empty coalition basis games. In MPE-PP the number of basis games is $m = 7$, in 5v6 it is $m = 31$, and in 8v9 it is $m = 120$.

indicate that setting the number of n-step return components as the number of non-empty coalition basis games can nearly maximize learning performance, thereby providing an empirical evidence to support the mathematical rationale behind the Linearity axiom that derives this setting for the NAHT. Although we do not deny that there may exist other preferred choices of m (as observed in the results of MPE-PP and 5v6), this does not undermine the effectiveness of using the number of non-empty coalition basis games, as a criterion, to decide the horizon for TTD(λ) in practice.

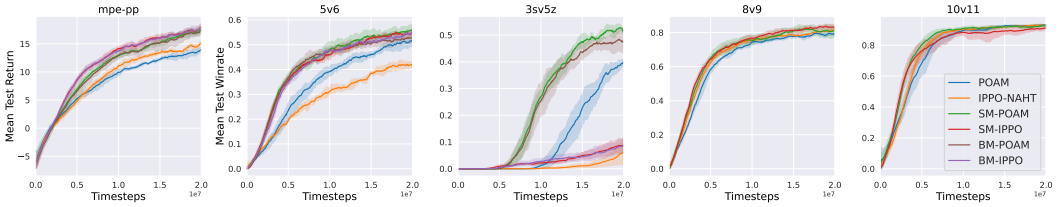
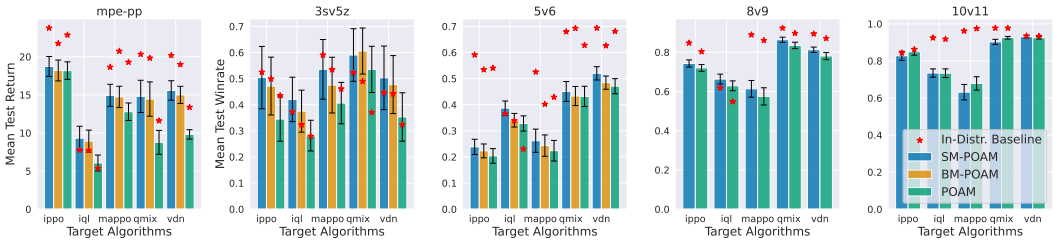


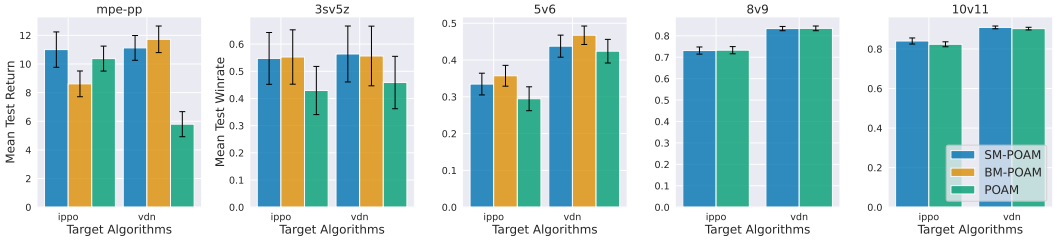
Figure 3: Evaluation curves during training in NAHT.

Answer to Question 2. We evaluate all baselines alongside our proposed algorithms. For the large-scale tasks 8v9 and 10v11, the base algorithms IPPO-NAHT and POAM are effectively equivalent to BM-IPPO and BM-POAM under the approximation of m due to limited episode lengths, and are therefore omitted in implementation. Figure 3 shows that, for either IPPO-NAHT or POAM, at least one of their Shapley Machine or Banzhaf Machine variants achieves the best performance. Recall that Banzhaf Machine corresponds to dropping the Efficiency axiom. In MPE-PP and 3sv5z with IPPO-NAHT as the base algorithm, Banzhaf Machine performs on par with Shapley Machine. In most other scenarios (except 10v11 with IPPO-NAHT), Shapley Machine outperforms Banzhaf Machine, although sometimes by a narrow margin. Moreover, comparisons between the base algorithms and their Banzhaf counterparts in MPE-PP, 5v6, and 3sv5z underscore the importance of the Linearity axiom. Overall, the results suggest that all three Shapley axioms generally enhance learning performance (in possibly unseen team compositions), but the Efficiency axiom can, in some cases, reduce it. This justifies the value of our proposed axiomatic framework, which provides a novel perspective on analysing performance through the falsifiable axioms. We believe that this axiomatic framework, as a paradigm, can help design new algorithms in the future.

Answer to Question 3. We evaluate all baselines alongside our proposed algorithms after training, to test their capability of generalization to unseen conventions and unseen agent types, respectively. The evaluation is conducted by the best model saved during training for each random seed of running experiments. Note that the wide range of confidence intervals is due to that the result averages performance across various number of controlled agents, and we further demonstrate the result per the number of controlled agents in Appendix I.5. It can be seen from Figure 4a that SM-POAM outperforms both BM-POAM and POAM on all scenarios, except for the 10v11. This implies that in general all Shapley axioms may be instrumental in tackling unseen conventions. As seen from Figure 4b, the performance of BM-POAM is comparable to and even better than SM-POAM in most cases. This implies that the Efficiency axiom could weaken the capability of generalization to unseen agent types. This phenomenon recurs, and becomes even more pronounced, in the evaluation of IPPO-NAHT related algorithms, as shown in Figure 11b in Appendix I.3.



(a) Evaluation on unseen conventions (identical algorithms with different random seeds). The star denotes the in-convention baseline, and the histograms show performance on unseen conventions.

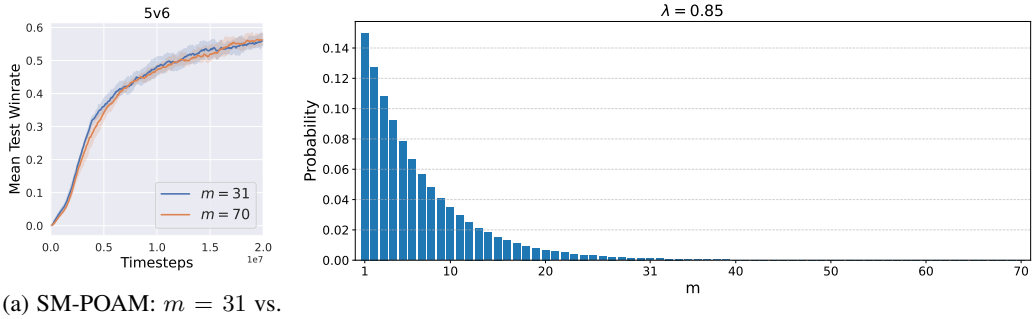


(b) Evaluation on unseen agent types. Train set: MAPPO, QMIX, IQL; test set: IPPO, VDN.

Figure 4: Out-of-distribution (OOD) evaluation after training, assessing unseen conventions and unseen agent types during training, between learned agents (POAM, SM-POAM and BM-POAM) and uncontrolled agent types (IPPO, VDN, IQL, QMIX and MAPPO). The performance is evaluated by averaging all pairings of N controlled and $M - N$ uncontrolled agents and their corresponding random seeds, called M–N score [Wang et al., 2025].

5.2 Number and Weightings of Basis Games

The weightings of basis games (k'_C in Eq. (6)) are set by the probability values of a geometric distribution governed by λ, respecting the convention of RL. λ will influence the shape of the geometric distribution. As λ increases, the geometric distribution becomes a longer tail with the probability decaying slowly as the number of basis games m increases. In other words, controlling m can be achieved by simply changing the weightings of basis games, or equivalently, the λ, under the assumption of the geometric distribution. To verify this result in the NAHT, we conduct a case study, as shown in Figure 5. It can be observed that the number of effective basis games (the probability values over which are non-zero) for $m = 31$ and $m = 70$ is the same under $\lambda = 0.85$, resulting in nearly identical learning performance. This aligns with the conventional view of λ's impact on TTD(λ) in RL, primarily as a mechanism for controlling the tail length of the return [Sutton and Barto, 2018][Chap. 12.3].

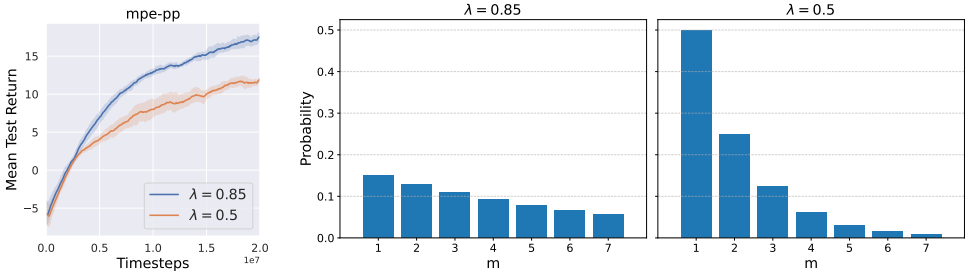


(a) SM-POAM: $m = 31$ vs. $m = 70$ for $\lambda = 0.85$.

(b) Demonstration of the shape of weightings of basis games with $\lambda = 0.85$.

Figure 5: Controlling the number of effective basis games can be implemented by changing the parameters to control the shape of weightings of basis games. Note that the probability values for $m > 31$ are nearly zero.

In contrast, we demonstrate that even with the same number of basis games, the values of their corresponding weightings significantly influence the learning performance. As shown in Figure 6, given the same number of active basis games as $m = 7$, the learning processes with $\lambda = 0.5$ and $\lambda = 0.85$ result in very different performance. This provides an alternative insight into how weightings influence the TTD(λ), different from the traditional RL perspective.



(a) SM-POAM: $\lambda = 0.5$ vs. $\lambda = 0.85$ for $m = 7$.

(b) Demonstration of weighting shapes of the basis games with $m = 7$, for $\lambda = 0.85$ and $\lambda = 0.5$, respectively.

Figure 6: Influence of different values of basis games given the same number of effective basis games m. Note that the shapes of geometric distributions vary significantly with different values of λ, under the same effective number of basis games.

6 Discussion, Limitations and Future Work

We studied reinforcement learning (RL) in open multi-agent systems through the lens of the n-agent ad hoc teamwork (NAHT), and introduced a cooperative game-theoretic framework that makes explicit how openness is distinguished from the cooperative multi-agent reinforcement learning (CMARL). Central to our approach is the NAHT decomposition, which separates internal cooperation among controlled agents from external teamwork with uncontrolled agents. Building on this structure, we proposed an axiomatic framework that derives value function learning from the Shapley axioms (Efficiency, Symmetry, and Linearity) by enforcing them as structural constraints. This perspective shifts the design of RL algorithms from heuristic constructions to principled designs in the NAHT. Since CMARL is an extreme case of the NAHT, our framework also works on it.

Our axiomatic framework reveals a novel connection between cooperative game theory and reinforcement learning: The Linearity axiom provides a theoretical explanation for the n-step return methods and the TTD(λ) in the NAHT. This result is coincidentally related to the latest finding in single-agent offline RL that the n-step return methods with selected horizons can improve scalability to the increasing size of datasets [Park et al., 2026]. As the NAHT encompasses the case of single-agent RL with other unknown agents, our framework suggests that, through the TTD(λ), the increasing size of datasets could be a promising way to address the NAHT, which is left for future work.

More broadly, the axiomatic perspective suggests new opportunities for exploring richer classes of weighting schemes beyond the standard geometric forms in temporal-difference learning for the NAHT (as demonstrated in Section 5.2). For example, weightings k_C satisfying $\sum(k_C)^2 = 1$ as shown in Figure 1 could define alternative weighting schedules. Because the NAHT problem can be reduced to the special case of single-agent RL with other uncontrolled agents, with an appropriate connection between the world model (transition function) and the concept of other uncontrolled agents, this result can be relaxed to the broader single-agent RL.

A key limitation of our work is that it is still unclear about how to deal with the scenario where the predefined largest horizon (the episode length) is far less than the number of non-empty coalition basis games. Our current horizon assignment is $n_i = i$, as clarified in Appendix D. Our tentative strategy to mitigate this shortcoming is approximating the number of basis games as the episode length. In future work, this can be further investigated by changing our current horizon assignment to another one, e.g., $n_i = f(i, s)$, where $f(i, s) \in \mathbb{N}$ is a function of states s and coalition indices i, based on Definition D.2. Another limitation is that our current work only considers the full agent team $\mathcal{M}$ as the instantiation of state-specific cooperative games, to match the purpose of our work. Exploring the physical implication of the instances of other coalitions $D \subset \mathcal{M}$ is left for future work.

Acknowledgements

Jianhong Wang and Jonathan Lawry are supported by the Engineering and Physical Sciences Research Council (EPSRC) [Grant Ref: EP/Y028732/1]. Samuel Kaski is supported by UKRI Turing AI World-Leading Researcher Fellowship, EP/W002973/1 and EU, ERC ODD-ML 101201120.

References

Noa Agmon and Peter Stone. Leading ad hoc agents in joint action settings with multiple teammates. In *AAMAS*, pages 341–348, 2012.

Noa Agmon, Samuel Barrett, and Peter Stone. Modeling uncertainty in leading ad hoc teams. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 397–404, 2014.

Stefano V Albrecht and Subramanian Ramamoorthy. A game-theoretic model and best-response learning method for ad hoc coordination in multiagent systems. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1155–1156, 2013.

Ronen I Brafman and Moshe Tennenholtz. On partially controlled multi-agent systems. *Journal of Artificial Intelligence Research*, 4:477–507, 1996.

Jiajun Chai, Yuqian Fu, Dongbin Zhao, and Yuanheng Zhu. Aligning credit for multi-agent cooperation via model-based counterfactual imagination. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pages 281–289, 2024.

Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge. *Computational aspects of cooperative game theory*. Morgan & Claypool Publishers, 2011.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Pawel Cichosz. Truncating temporal differences: On the efficient implementation of td (lambda) for reinforcement learning. *Journal of Artificial Intelligence Research*, 2:287–318, 1994.

Peter Dayan. The convergence of td (λ) for general λ. *Machine learning*, 8:341–362, 1992.

Peter Dayan and Terrence J Sejnowski. Td (λ) converges with probability 1. *Machine Learning*, 14: 295–301, 1994.

Christian Schroeder De Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip HS Torr, Mingfei Sun, and Shimon Whiteson. Is independent learning all you need in the starcraft multi-agent challenge? *arXiv preprint arXiv:2011.09533*, 2020.

Pradeep Dubey. On the uniqueness of the shapley value. *International Journal of Game Theory*, 4(3): 131–139, 1975.

Gianluigi Greco, Enrico Malizia, Luigi Palopoli, Francesco Scarcello, et al. On the complexity of the core over coalition structures. In *IJCAI*, volume 11, pages 216–221, 2011.

Dongge Han, Chris Xiaoxuan Lu, Tomasz P. Michalak, and Michael J. Wooldridge. Multiagent model-based credit assignment for continuous control. In *21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022*, pages 571–579. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 2022.

Jiahui Li, Kun Kuang, Baoxiang Wang, Furui Liu, Long Chen, Fei Wu, and Jun Xiao. Shapley counterfactual credits for multi-agent reinforcement learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 934–942, 2021.

Yang Li, Junfan Chen, Feng Xue, Jiabin Qiu, Wenbin Li, Qingrui Zhang, Ying Wen, and Wei Pan. At-drone: Benchmarking adaptive teaming in multi-drone pursuit. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 1144–1161. PMLR, 2025.

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

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

Seohong Park, Kevin Frans, Deepinder Mann, Benjamin Eysenbach, Aviral Kumar, and Sergey Levine. Horizon reduction makes rl scalable. *Advances in Neural Information Processing Systems*, 38:8350–8389, 2026.

Jing Peng and Ronald J Williams. Incremental multi-step q-learning. In *Machine Learning Proceedings 1994*, pages 226–232. Elsevier, 1994.

Muhammad A Rahman, Niklas Hopner, Filippos Christianos, and Stefano V Albrecht. Towards open ad hoc teamwork using graph-based policy learning. In *International Conference on Machine Learning*, pages 8776–8786. PMLR, 2021.

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

Harm Seijen and Rich Sutton. True online td (lambda). In *International Conference on Machine Learning*, pages 692–700. PMLR, 2014.

LS Shapley. A value for n-person games. *Annals of Mathematics Studies*, 28:307–318, 1953.

Peter Stone and Sarit Kraus. To teach or not to teach? decision making under uncertainty in ad hoc teams. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 117–124, 2010.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3: 9–44, 1988.

Richard S Sutton and Andrew G Barto. Reinforcement learning: An introduction second edition. *Adaptive computation and machine learning: The MIT Press, Cambridge, MA and London*, 2018.

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

Caroline Wang, Muhammad Arrasy Rahman, Ishan Durugkar, Elad Liebman, and Peter Stone. N-agent ad hoc teamwork. *Advances in Neural Information Processing Systems*, 37:111832–111862, 2025.

Jianhong Wang, Yuan Zhang, Tae-Kyun Kim, and Yunjie Gu. Shapley q-value: A local reward approach to solve global reward games. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7285–7292, 2020.

Jianhong Wang, Wangkun Xu, Yunjie Gu, Wenbin Song, and Tim C Green. Multi-agent reinforcement learning for active voltage control on power distribution networks. *Advances in Neural Information Processing Systems*, 34:3271–3284, 2021.

Jianhong Wang, Yuan Zhang, Yunjie Gu, and Tae-Kyun Kim. Shaq: Incorporating shapley value theory into multi-agent q-learning. *Advances in Neural Information Processing Systems*, 35: 5941–5954, 2022.

Jianhong Wang, Yang Li, Yuan Zhang, Wei Pan, and Samuel Kaski. Open ad hoc teamwork with cooperative game theory. In *International Conference on Machine Learning*, pages 50902–50930. PMLR, 2024.

Christopher John Cornish Hellaby Watkins et al. Learning from delayed rewards. *PhD Thesis*, 1989.

David Williams. *Probability with martingales*. Cambridge university press, 1991.

Annie Xie, Dylan Losey, Ryan Tolsma, Chelsea Finn, and Dorsa Sadigh. Learning latent representations to influence multi-agent interaction. In *Conference on robot learning*, pages 575–588. PMLR, 2021.

Di Xue, Lei Yuan, Zongzhang Zhang, and Yang Yu. Efficient multi-agent communication via shapley message value. In *IJCAI*, pages 578–584, 2022.

H Peyton Young. Monotonic solutions of cooperative games. *International Journal of Game Theory*, 14(2):65–72, 1985.

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

Yuan Zhang, Umashankar Deekshith, Jianhong Wang, and Joschka Boedecker. Improving the efficiency and efficacy of multi-agent reinforcement learning on complex railway networks with a local-critic approach. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 34, pages 698–706, 2024.

Luisa M. Zintgraf, Sam Devlin, Kamil Ciosek, Shimon Whiteson, and Katja Hofmann. Deep interactive bayesian reinforcement learning via meta-learning. In Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé, editors, *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, pages 1712–1714. ACM, 2021.

A Related Work

A.1 TD(λ) in Reinforcement Learning and Multi-Agent Reinforcement Learning

Related work on temporal-difference learning with eligibility traces centers on the seminal TD(λ) algorithm proposed by Sutton [1988], which unified one-step TD and Monte Carlo methods via a trace-decay parameter λ and introduced the forward- and backward-view equivalence for efficient multi-step credit assignment. Subsequent theoretical analyses [Dayan, 1992, Dayan and Sejnowski, 1994, Tsitsiklis and Van Roy, 1996] established convergence guarantees and characterized the bias-variance trade-off inherent in λ-returns, while extensions by Watkins et al. [1989], Peng and Williams [1994] adapted eligibility traces for off-policy control, Q(λ). Advances in function approximation, including true online TD(λ) [Seijen and Sutton, 2014] and gradient-TD methods, further broadened applicability to large-scale and nonlinear settings, inspiring n-step return techniques such as generalized advantage estimation (GAE) [Schulman et al., 2015] for policy optimization. In the multi-agent reinforcement learning (MARL), algorithms of PPO family such as MAPPO [Yu et al., 2022] and IPPO [De Witt et al., 2020], applied GAE to optimize policies and thus used the TD(λ) as the target values to train critics. Our work highlights the theoretical link between the truncated TD(λ), denoted as TTD(λ) [Cichosz, 1994], in the NAHT and the Linearity axiom of the Shapley value. It further shows that the number of n-step return components in the TTD(λ) can be estimated by the number of non-empty coalition basis games. This insight provides a theoretical foundation that can inform the design of a broader class of MARL algorithms.

A.2 Theoretical Models for Ad Hoc Teamwork

We now discuss theoretical models describing the ad hoc teamwork (AHT). Brafman and Tennenholtz [1996] pioneered research into the AHT by investigating repeated matrix games involving a single teammate. Subsequent studies expanded this framework to scenarios with multiple teammates, notably by Agmon and Stone [2012]. Later, Agmon et al. [2014] further relaxed earlier assumptions by allowing teammates' policies to be selected from a known set. Stone and Kraus [2010] initially formalized the AHT through collaborative multi-armed bandits, albeit under notable assumptions such as prior knowledge of teammates' policies and environmental conditions. Albrecht and Ramamoorthy [2013] advanced this field significantly by introducing the stochastic Bayesian game (SBG), the first comprehensive theoretical framework accommodating dynamic environments and unknown teammate behaviours in the AHT. Building on the SBG, Rahman et al. [2021] proposed the open stochastic Bayesian game (OSBG), addressing the open ad hoc teamwork (OAHT). Zintgraf et al. [2021] modelled the AHT through interactive Bayesian reinforcement learning (IBRL) within Markov games, specifically targeting non-stationary teammate policies within single episodes. Xie et al. [2021] introduced the hidden parameter Markov decision process (HiP-MDP) to handle situations where teammates' policies vary across episodes but remain stationary during individual episodes. Most recently, Wang et al. [2024] extended the OSBG by incorporating principles from cooperative game theory, introducing the open stochastic Bayesian coalitional affinity game (OSBG-CAG), which theoretically justifies a graph-based representation for joint Q-value functions and includes rigorous convergence proofs for Q-learning algorithms in open team settings. Our work adopts the perspective of cooperative game theory, introducing an NAHT decomposition based on state-specific cooperative games to address the NAHT. This well captures the structural difference between the CMARL and the NAHT. Based on the NAHT decomposition, we derive an axiomatic framework to address the NAHT, which introduces a novel perspective for solving this problem. The key difference in assumptions between the work by Wang et al. [2024] and our work is that the work by Wang et al. [2024] was built on the non-transferable utility games, while our work is built on the transferable utility games. For more details about the difference between these two cooperative game models, please refer to Chalkiadakis et al. [2011].

A.3 Shapley Value in Multi-Agent Reinforcement Learning

Early studies [Wang et al., 2020] incorporated the Shapley value into credit assignment scheme in a principled way and proposed an algorithm named as SQDDPG, underpinned by the equivalence between cooperative-game theoretical models in dynamic scenarios and the team reward Markov game. Wang et al. [2022] further improved the theory by proving the existence of the Shapley value under the team reward Markov game through construction, and proposed an algorithm, relying on

the construction, named as SHAQ — an algorithm with promising convergence to find an optimal joint policy. It also shed light on the relationship between the Shapley value and the relevant value decomposition and credit assignment approaches for the CMARL. Li et al. [2021] improved the stability of SQDDPG and proposed an algorithm named Shapley Counterfactual Credits. Han et al. [2022] incorporated the Shapley value into the model-based PPO as each agent's advantage value, through the estimated coalition values by trained transition and reward models. Chai et al. [2024] continued the idea from Han et al. [2022], by replacing the transition and reward models with more powerful world models. On the other hand, Xue et al. [2022] incorporated the Shapley value into multi-agent communication, as a criterion for forming communication between agents. One of our work's aims is learning the Shapley value to address the NAHT, a generalized problem to the CMARL. More importantly, the Shapley value is learned by satisfying its axioms, rather than by constructing their explicit formulas as implemented in the previous work. This may help overcome the notorious learning instability issues caused by learning substantial "coalitional values" in the existing algorithms based on the Shapley value. Furthermore, the insight of axiomatic characterization can facilitate improving the flexibility of the Shapley value approaches in a wide variety of tasks. Specifically, one can safely drop any part of a learning algorithm, if it is justified that the targeting task does not align with the corresponding axiom.

B Extra Background

B.1 λ-Return and TD(λ)

We now introduce an extension of return named λ-return. Mathematically, a λ-return G_t^λ for infinite-horizon cases can be expressed as follows:

$$\begin{aligned} G_{t:t+n} &= R_t + \gamma R_{t+1} + \cdots + \gamma^{n-1} R_{t+n-1} + \gamma^n v_\pi(s_{t+n}), \\ G_t^\lambda &= (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} G_{t:t+n}. \end{aligned} \tag{10}$$

Here, v_π denotes the state-value function under policy π. Similarly, the λ-return for finite-horizon cases can be expressed as follows:

$$G_t^\lambda = (1-\lambda) \sum_{n=1}^{T-t} \lambda^{n-1} G_{t:t+n} + \lambda^{T-t} G_{t:T}. \tag{11}$$

Note that if $\lambda = 1$, updating according to the λ-return is a Monte Carlo algorithm. In contrast, if $\lambda = 0$, then the λ-return reduces to $G_{t:t+1}$, the one-step return. The Monte Carlo algorithm is known as its high variance but low bias, while one-step return is known as its low variance but high bias. For this reason, the change of λ can be seen as a tradeoff between bias and variance. The TD prediction using λ-return as the target value is referred to as TD(λ) [Sutton, 1988].

A variant of TD(λ) by shaping G_t^λ with a designated horizon $H \in \{t+1, ..., T\}$ is referred to as truncated TD(λ), shortened as TTD(λ) [Cichosz, 1994]. Mathematically, G_t^λ with a designated horizon H can be formulated as follows:

$$G_t^{\lambda,H} = (1-\lambda) \sum_{n=1}^{H-t} \lambda^{n-1} G_{t:t+n} + \lambda^{H-t} G_{t:H}. \tag{12}$$

Note that in the convention of RL, the horizon H is defined as the time difference from the starting point of an episode, denoted by $t = 0$.

Remark B.1. *In this paper, to simplify the presentation of our work, we redefine the horizon as the time difference from each timestep t, denoted as m, such that $m = H - t$. With the new definition of*

horizon, we can rewrite the above formula of $G_t^{\lambda,H}$ as $G_t^{\lambda,m}$, as follows:

$$\begin{aligned} G_t^{\lambda,m} &= (1-\lambda)\sum_{n=1}^{m}\lambda^{n-1}G_{t:t+n} + \lambda^m G_{t:t+m} \\ &= (1-\lambda)\sum_{n=1}^{m-1}\lambda^{n-1}G_{t:t+n} + \big[(1-\lambda)\lambda^{m-1} + \lambda^m\big]G_{t:t+m} \\ &= (1-\lambda)\sum_{n=1}^{m-1}\lambda^{n-1}G_{t:t+n} + \lambda^{m-1}G_{t:t+m}. \end{aligned} \tag{13}$$

B.2 Superadditive Game and Cooperative Multi-Agent Reinforcement Learning

The superadditive game is a subclass of characteristic games that satisfies an additional condition: $v(C \cup D) \geq v(C) + v(D)$, for any two distinct coalitions $C, D \subseteq \mathcal{M}$ with $C \cap D = \emptyset$. Previous work in the CMARL [Wang et al., 2020] has shown the equivalence on the task objective function between the state-based superadditive games with action spaces and the team reward Markov games with coalition structures.[7] Intuitively, the Shapley value is a solution to the stability of a team formation, resembling cooperation. Since the NAHT is a generalization of the CMARL, this conclusion is still valid in principle. It has been already proved that the Shapley value exists in superadditive games [Shapley, 1953] and each game instance in superadditive games can be uniquely represented as $\sum k_C v_C^1$, where $k_C \geq 0$ [Dubey, 1975]. In cooperative game theory, any game can be transformed to a superadditive game if its value v is transformed by a *superadditive cover* v^* such that $v^*(C) = \max_{CS(C)} \sum_{D \in CS(C)} v(D)$ [Greco et al., 2011]. As a result, it will not lose generality if we consider the cooperative game space as superadditive games in our work.

B.3 Axioms of Shapley Values

Efficiency and Symmetry have been well investigated in the literature of multi-agent reinforcement learning [Wang et al., 2020, Li et al., 2021, Han et al., 2022, Wang et al., 2022, Chai et al., 2024], but Additivity still lacks attention. In detail, Additivity means that $\phi(w_1 + w_2) = \phi(w_1) + \phi(w_2)$, for any cooperative games $w_1, w_2 \in \mathcal{G}$. If we consider m possible games, then we have the following expression: $\phi(\sum_{i=1}^m w_i) = \sum_{i=1}^m \phi(w_i)$, where $w_1, w_2, ..., w_m \in \mathcal{G}$ and the sum $\sum_{i=1}^m w_i$ can be seen as a game reproduced by the above games. Linearity is a stronger condition than Additivity [Dubey, 1975, Young, 1985]. Specifically, Linearity requires $\phi(\sum_{i=1}^m \alpha_i w_i) = \sum_{i=1}^m \alpha_i \phi(w_i)$, for all $\alpha_i \in \mathbb{R}$. In contrast, Additivity only requires to satisfy the above equation when $\alpha_i = 1$. In our work, we use the Linearity axiom to establish the Additivity axiom, to adapt to our derivation.

B.4 Baseline Algorithms: POAM and IPPO

POAM. In general, POAM is an algorithm built upon IPPO [De Witt et al., 2020]. The primary difference between POAM and IPPO is as follows: For each agent's policy and value, in addition to its observation used in IPPO, POAM takes as input an embedding vector to predict other agents' potential behaviours. More specifically, this embedding vector is trained by an encoder-decoder structure, wherein the encoder is modeled as RNNs and decoder is modeled as MLPs. For brevity, we use $-i$ to denote the set of all agents excluding agent i. Let $h_{t,i} = \{o_{k,i}, a_{k-1,i}\}_{k=1}^t$ denote agent i's history of observations and actions up to timestep t and $e_{t,i} \in \mathbb{R}^n$ denote the resulting embedding vector of n dimensions. The encoder parameterized by θ^e is defined as $f_{\theta^e}^{enc} : \mathcal{H}_i \to \mathbb{R}^n$. The embedding vector is decoded by two decoder networks: the observation decoder $f_{\theta^o}^{dec} : \mathbb{R}^n \to \mathcal{O}_{-i}$, and the action decoder $f_{\theta^a}^{dec} : \mathbb{R}^n \to \Delta(\mathcal{A}_{-i})$. The decoder networks are respectively trained to predict the observations and actions of all other agents in the team at timestep t, $o_{t,-i}$ and $a_{t,-i}$, to encourage $e_{t,i}$ to contain information about collective behaviors corresponding to $h_{t,i}$. While the observation decoder directly predicts the observed $-i$'s observations, the action decoder predicts the parameters of a probability distribution over the $-i$'s actions $\pi_{-i}(a_{t,-i}; f_{\theta^a}^{dec}(f_{\theta^e}^{enc}(h_{t,i})))$. As we consider continuous observations and discrete actions, the loss function with using categorical

[7] In Wang et al. [2020], state-based convex games with action spaces were proved to be equivalent to team reward Markov games. However, setting $C \cap D = \emptyset$ can relax a convex game to a superadditive game.

distribution to model π_{-i} is as follows:

$$L_{\theta^e,\theta^o,\theta^a}(h_{t,i}, o_{t,-i}, a_{t,-i}) = ||f^{dec}_{\theta^o}(f^{enc}_{\theta^e}(h_{t,i})) - o_{t,-i}||^2 - \log \pi_{-i}(a_{t,-i}; f^{dec}_{\theta^a}(f^{enc}_{\theta^e}(h_{t,i}))). \quad (14)$$

For brevity, we define $g(h) := (h, f^{enc}_{\theta^e}(h))$ in the following description.

POAM employed generalized advantage estimation (GAE) [Schulman et al., 2015] to form the policy gradient, and the value function used to form GAE is trained by the following loss function:

$$L_{\theta^c}(h_{t,i}) = \frac{1}{2}\left(V_i^{\theta^c}(g(h_{t,i})) - \hat{V}_{t,i}\right)^2, \quad (15)$$

where $\hat{V}_{t,i}$ is the finite-horizon TD(λ) return, for which the horizon is up to the episode length T. Each agent i's GAE estimation is as follows:

$$A_{t,i} = \sum_{l=0}^{T} (\gamma\lambda)^l \delta_{t+l,i}, \quad (16)$$

where $\delta_{t,i} = R_t + \gamma V_i(g(h_{t,i})) - V_i^{\theta^c}(g(h_{t,i}))$ and $V_i(g(h_{t,i}))$ denotes the non-parameterized target individual value. Based on the GAE defined above, the policy optimization loss is defined as

$$L_\theta(h_{t,i}, a_{t,i}) = \min\left\{\frac{\pi_\theta(a_{t,i}|g(h_{t,i}))}{\pi_{\theta_{old}}(a_{t,i}|g(h_{t,i}))}A_{t,i}, \text{clip}\left(\frac{\pi_\theta(a_{t,i}|g(h_{t,i}))}{\pi_{\theta_{old}}(a_{t,i}|g(h_{t,i}))}, 1-\epsilon, 1+\epsilon\right)A_{t,i}\right\}. \quad (17)$$

IPPO. By removing the loss for the encoder-decoder structure as shown in Eq. (14) and replacing the input $g(h_{t,i})$ by $h_{t,i}$ in the above set of equations, we get the loss functions of IPPO.

C Preliminary Results

Lemma C.1. *Given a fixed state $s \in \mathcal{S}$, each state-specific cooperative game $v_s \in \mathcal{G}(s)$ can be uniquely represented by $v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k'_C \cdot v^z_{C,s}$, where $k'_C = \frac{k_C}{z}$ and $z \neq 0$.*

Proof. We can represent each k_C in an equivalent form as $\frac{k_C \cdot z}{z}$, where $z \neq 0$. Substituting this term into the formula $v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k_C v^1_{C,s}$ in Definition 3.1, we can get the following formula such that

$$v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} \frac{k_C \cdot z}{z} \cdot v^1_{C,s}.$$

Since $v^z_{C,s} = z \cdot v^1_{C,s}$ by Definition 3.1, we have the following formula such that

$$v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} \frac{k_C}{z} \cdot v^z_{C,s}.$$

□

Proposition C.2. *For the class of superadditive games formed by a set of state-specific basis games $\{v^z_{C,s} | \emptyset \neq C \subseteq \mathcal{M}\}$, such that $v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k'_C \cdot v^z_{C,s}$, it holds that $k'_C \geq 0$, for all $\emptyset \neq C \subseteq \mathcal{M}$.*

Proof. We first consider a cooperative game space $\mathcal{G}$ (a broader space of superadditive games). As per Dubey [1975], a game $v : \mathcal{G} \to \mathbb{R}_{\geq 0}$ belonging to $\mathcal{G}$ can be uniquely represented as follows:

$$v = \sum_{\emptyset \neq C \subseteq \tilde{N}} k_C v^1_C.$$

The analytic form of k_C under $\mathcal{G}$ is represented as follows:

$$k_C = \sum_{T \subseteq C} (-1)^{|C|-|T|} v(T).$$

The condition for v to be a superadditive game is as follows:

$$v(T_1 \cup T_2) \geq v(T_1) + v(T_2), \qquad T_1, T_2 \subseteq C.$$

Following the result from Dubey [1975], if the above condition holds, $k_C \geq 0$ has to be satisfied.[8]

We now extend the $\mathcal{G}$ to a state-specific cooperative game space $\mathcal{G}(s)$, for a fixed state $s \in \mathcal{S}$. Since the $\mathcal{G}(s)$ is isomorphic to a game space $\mathcal{G}$, the characteristics satisfied in the $\mathcal{G}$ also holds in the $\mathcal{G}(s)$. Therefore, $v_s : \mathcal{G}(s) \to \mathbb{R}_{\geq 0}$ can be uniquely represented as $v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k_C v^1_{C,s}$. As we consider $\mathcal{G}(s)$ as a space of superadditive games, $k_C \geq 0$ holds.

By Lemma C.1, we have $v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k'_C \cdot v^z_{C,s}$, where $k'_C = \frac{k_C}{z}$ and $z \neq 0$. This implies that any set of basis games in the form $\{v^z_{C,s} | \emptyset \neq C \subseteq \mathcal{M}\}$ can form a state-specific cooperative game belonging to superadditive games. The proof is completed. □

Remark C.3. *Following discussion in Section 2.2, we aim to restrict the state-specific cooperative game class to the superadditive game. By Proposition C.2, we can conclude that $k'_C \geq 0$ is a sufficient condition for v_s belonging to the superadditive game class. In other words, $k'_C \geq 0$ is a key condition for reaching cooperation within a team of agents. However, $k'_C = 0$ is a singularity condition, which leads to 0 for all state-specific cooperative games in values, with no actual meaning in practice. For this reason, we only choose $k'_C > 0$ as the condition in our work.*

D Introducing Markovian Dynamics and Rewards to Game Structures

Based on the game structure for the NAHT in Definition 3.2, we now introduce Markovian dynamics to establish the link between the state-specific cooperative game values of consecutive states, aligned with the environmental setting in Dec-POMDPs [Oliehoek et al., 2016].

The only postulate is that the state-specific cooperative game value for each state s_t can be represented in the form of rewards: $v_{s_t}(\mathcal{M}) = \mathbb{E}_\pi[\sum_{\tau=0}^{T-t-1} \gamma^\tau R_{t+\tau} \mid s_t]$. By the convention of RL [Sutton and Barto, 2018][Chap. 7], it is feasible to use the expected n-step return to express the $v_{s_t}(\mathcal{M})$ under a given horizon n to define the transformation, such that

$$v_{s_t}(\mathcal{M}) = \mathbb{E}_\pi \left[\sum_{\tau=0}^{n-1} \gamma^\tau R_{t+\tau} + \gamma^n v_{s_{t+n}}(\mathcal{M}) \,\middle|\, s_t \right] := \mathbb{E}_\pi \left[G_{t:t+n} \mid s_t \right]. \tag{18}$$

For brevity, we denote the state-specific basis game value $v^{v_s(\mathcal{M})}_{C,s}(\mathcal{M})$ by $V(C, s)$ for all $C \subseteq \mathcal{M}$. Then, we get the following fact:

Fact D.1. *$V(C_1, s) = V(C_2, s) = \cdots = V(C_m, s) = v_s(\mathcal{M})$ for all $C_i \in \mathcal{P}_+(\mathcal{M})$, as per Definition 2.1 and $\mathcal{M} \supseteq C_i$, where $\mathcal{P}_+(\mathcal{M})$ indicates the powerset of $\mathcal{M}$ excluding the empty set.*

Definition D.2 (Coalition Ordering by Size and Horizon Assignment). *We first set $m := 2^M - 1$. Given a coalition enumeration $(C_1, \ldots, C_m)$ satisfying (1) $|C_i| \leq |C_{i+1}|$ for all i (non-decreasing in size); (2) within each size, ties are broken by a fixed total order (e.g., lexicographic ordering over agent indices), we can define a sequence of non-decreasing horizons $(n_1, \ldots, n_m)$ such that $n_i \geq n_{i-1}$ for all $i \geq 2$ and $n_1 := 1$. Accordingly, we represent the basis game values $V(C_i, s)$ as their corresponding expected n-step returns:*

$$V(C_i, s) \;:=\; \mathbb{E}_\pi\big[G_{t:t+n_i} \mid s_t = s\big], \qquad (i = 1, \ldots, m).$$

We motivate the horizon assignment in Definition D.2 using the variance comparison in Appendix E. Under the stated assumptions and for a fixed state s, the basis game value estimators $\widehat{V}(C_i, s)$ and the sampled n_i-step returns $G_{t:t+n_i}$ have variances that are both non-decreasing in i. This definition concerns the following situation of coalition ordering: When the number of agents increases (i.e., C_i becomes larger in size), more diverse agent types and behaviours emerge. As a result, the estimator of the corresponding $V(C_i, s)$ has a higher variance.

[8] For example, if $C = \{1, 2\}$, then $T = \emptyset, \{1\}, \{2\}, \{1, 2\}$. As a result, $k_C = v(\emptyset) - v(\{1\}) - v(\{2\}) + v(\{1, 2\})$, where $v(\emptyset) = 0$ by default. If v is assumed to be a superadditive game, $k_C \geq 0$ has to hold.

Proposition D.3. *By Definition D.2, with the horizon assignment $n_i = i$, for every state $s \in \mathcal{S}$,*

$$\big(V(C_1, s), \ldots, V(C_m, s)\big) \;=\; \big(\mathbb{E}_\pi[G_{t:t+1} \mid s], \ldots, \mathbb{E}_\pi[G_{t:t+m} \mid s]\big),$$

and, each $V(C_i, s)$ is equal to $v_s(\mathcal{M})$ in value.

Proof. The result can be immediately obtained from the definition of $V(C_i, s)$ in Definition D.2, and the statement that each $V(C_i, s)$ is equal to $v_s(\mathcal{M})$, directly following Fact D.1. □

E Justification of Coalition Ordering by Size and Horizon Assignment

E.1 Sketch of Justification Procedure

In this section, we will justify the insight behind the coalition ordering by size and horizon assignment, as shown in Definition D.2. We fix a policy π and use its state-value function $v_\pi : \mathcal{S} \to \mathbb{R}$ as the bootstrap value function, as in Section B.1. The function v_π is held fixed during trajectory sampling. We then define a discrete-time stochastic process of TD error terms $(Z_{n+1})_{n \in \mathbb{N}_0}$:

$$Z_{n+1} \;:=\; R_{t+n} + \gamma v_\pi(s_{t+n+1}) - v_\pi(s_{t+n}).$$

As a result, the n-step return $G_{t:t+n}$, as an estimator of the expected n-step return, has the following recursive relationship:

$$G_{t:t+n+1} = G_{t:t+n} + \gamma^n Z_{n+1}.$$

We first show in Lemma E.1 that, given $s_t = s$, the variance of the n-step return $\mathrm{Var}(G_{t:t+n} \mid s)$ is non-decreasing in a monotonically increasing horizon assignment when the TD-error sequence (Z_n) has non-negative autocovariances. Specifically, extending the horizon from n to $n+1$ adds the discounted TD-error term $\gamma^n Z_{n+1}$, whose covariance with the existing return is non-negative. The return variance therefore increases strictly when this additional term has positive conditional variance and remains unchanged when it has zero conditional variance. Then, we make the coalition-variance monotonicity assumption in Assumption E.2, which can be justified by Remark E.3 with evidence from the definition of the game structure for the NAHT and Remark E.4 with evidence from a verification experiment. Briefly, Assumption E.2 concerns the following situation: When the number of agents increases (i.e., C_i becomes larger in size), more diverse agent types and behaviours emerge. As a result, the estimator of the corresponding $V(C_i, s)$ has a higher variance.

For the state-specific basis-game representation, we identify the bootstrap value with the team value, such that $v_\pi(s) = v_s(\mathcal{M})$. By Lemma E.1 and Assumption E.2, we prove in Proposition E.5 that there exist non-decreasing sequences of variances for both the sequence of estimators of $V(C_i, s)$ under non-decreasing coalitions in size and the sequence of n-step returns $G_{t:t+n_i}$ under a non-decreasing horizon assignment. This motivates us to define the following term under this monotone-variance regularity:

$$V(C_i, s) \;:=\; \mathbb{E}_\pi\big[G_{t:t+n_i} \mid s_t = s\big], \qquad (i = 1, \ldots, m).$$

In other words, the n-step return $G_{t:t+n_i}$ is defined as an unbiased estimator for the basis game value $V(C_i, s)$ for each coalition C_i. The justification of Definition D.2 is completed.

E.2 Theoretical and Empirical Justification

Lemma E.1. *If (Z_n) has non-negative autocovariances given $s_t = s$, then $n \mapsto \mathrm{Var}(G_{t:t+n} \mid s)$ is non-decreasing. Also, $\mathrm{Var}(G_{t:t+n+1} \mid s) = \mathrm{Var}(G_{t:t+n} \mid s)$ if and only if $\mathrm{Var}(Z_{n+1} \mid s) = 0$.*

Proof. Use $G_{t:t+n+1} = G_{t:t+n} + \gamma^n Z_{n+1}$ and expand the variance conditioning on $s_t = s$, we can get the following formula:

$$\mathrm{Var}(G_{t:t+n+1} \mid s) = \mathrm{Var}(G_{t:t+n} \mid s) + 2\gamma^n \mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) + \gamma^{2n} \mathrm{Var}(Z_{n+1} \mid s).$$

We first prove that if (Z_n) has non-negative autocovariances given $s_t = s$, then $n \mapsto \mathrm{Var}(G_{t:t+n} \mid s)$ is non-decreasing, as follows:

Using the telescoping sum of TD errors [Sutton and Barto, 2018][Chap. 6], we have the following formula:

$$G_{t:t+n} - v_\pi(s_t) = \sum_{k=0}^{n-1} \gamma^k Z_{k+1}.$$

Since $v_\pi(s_t) = v_\pi(s)$ is constant given $s_t = s$, linearity of covariance yields the following equation:

$$\mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) = \sum_{k=0}^{n-1} \gamma^k \mathrm{Cov}(Z_{k+1}, Z_{n+1} \mid s).$$

Therefore, if all autocovariances $\mathrm{Cov}(Z_i, Z_j \mid s) \geq 0$, then $\mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) \geq 0$.

Since $\mathrm{Var}(Z_{n+1} \mid s) \geq 0$ must hold, we obtain the following inequality:

$$\mathrm{Var}(G_{t:t+n+1} \mid s) \geq \mathrm{Var}(G_{t:t+n} \mid s).$$

That is, $n \mapsto \mathrm{Var}(G_{t:t+n} \mid s)$ is non-decreasing.

We next prove $\mathbf{Var}(\boldsymbol{G_{t:t+n+1}} \mid \boldsymbol{s}) = \mathbf{Var}(\boldsymbol{G_{t:t+n}} \mid \boldsymbol{s})$ if and only if $\mathbf{Var}(\boldsymbol{Z_{n+1}} \mid \boldsymbol{s}) = \mathbf{0}$, as follows:

Given a fact that $\mathrm{Var}(G_{t:t+n+1} \mid s) = \mathrm{Var}(G_{t:t+n} \mid s)$ if and only if the following condition holds:

$$\mathrm{Var}(Z_{n+1} \mid s) = 0, \qquad \mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) = 0,$$

we now aim to prove that $\mathrm{Var}(Z_{n+1} \mid s) = 0$ is sufficient for $\mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) = 0$, and the proof is completed:

If a non-negative random variable (r.v.) $W \geq 0$ satisfies $\mathbb{E}[W \mid s] = 0$, then $W = 0$ almost surely (a.s.) given s [Williams, 1991][Chap. 9].

Apply this with $W = (Z_{n+1} - \mathbb{E}[Z_{n+1} \mid s])^2$. If $\mathrm{Var}(Z_{n+1} \mid s) = 0$, then

$$\mathbb{E}\left[(Z_{n+1} - \mathbb{E}[Z_{n+1} \mid s])^2 \mid s\right] = 0 \Rightarrow (Z_{n+1} - \mathbb{E}[Z_{n+1} \mid s])^2 = 0 \text{ a.s.} \Rightarrow Z_{n+1} = \mathbb{E}[Z_{n+1} \mid s] \text{ a.s.}$$

The covariance $\mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big)$ can be expressed as the following formula:

$$\mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) = \mathbb{E}\big[\big(G_{t:t+n} - \mathbb{E}[G_{t:t+n} \mid s]\big)\big(Z_{n+1} - \mathbb{E}[Z_{n+1} \mid s]\big) \,\big|\, s\big].$$

Since $Z_{n+1} - \mathbb{E}[Z_{n+1} \mid s] = 0$ a.s., we get the following result:

$$\mathrm{Cov}\big(G_{t:t+n}, Z_{n+1} \mid s\big) = 0.$$

□

Assumption E.2 (Coalition-Variance Monotonicity). *Let $\widehat{V}(C, s)$ be an unbiased estimator of $V(C, s)$ from on-policy data restricted to agents in C, such that $\mathbb{E}[\widehat{V}(C, s)] = V(C, s)$. $\mathrm{Var}(\widehat{V}(C, s))$ is non-decreasing in coalition size $|C|$.*

Remark E.3 (Explanation for Assumption E.2). *Recall from Section C that $v_{C,s}^{v_s(\mathcal{M})} = v_s(\mathcal{M}) v_{C,s}^{1}$. Following Definition 3.1, we can express $v_{C,s}^{v_s(\mathcal{M})}$ as follows:*

$$v_{C,s}^{v_s(\mathcal{M})}(D) = \begin{cases} v_s(\mathcal{M}) & \text{if } D \supseteq C, \\ 0 & \text{otherwise}. \end{cases}$$

As described in Definition 3.2, a state-specific basis game value $v_{C,s}^{v_s(\mathcal{M})}(C)$ can effectively evaluate the performance of a coalition of agents C[9] with an agent-type composition denoted by $\times_{i \in C} T_i$, where T_i indicates the agent i's type. Therefore, we can get the following result: The larger the size of coalition C, the more agent-type compositions need to be explored, so that the estimator $\widehat{V}(C, s)$ of $V(C, s) := v_{C,s}^{v_s(\mathcal{M})}(\mathcal{M})$ can precisely estimate $v_s(\mathcal{M})$. In other words, as the size of coalition C increases, higher variance is incurred to precisely estimate $v_s(\mathcal{M})$, due to the growing agent-type composition space. Similarly, if the sizes of two coalitions are equal, their variances are equal. This leads to Assumption E.2.

[9] For any team of agents $D \supset C$, the state-specific basis game value would be the same as $v_{C,s}^{v_s(\mathcal{M})}(C)$, so it is ineffective to evaluate the performance of the D with the larger size than C.

Remark E.4 (Empirical Verification for Assumption E.2). *To make Assumption E.2 more convincing, we additionally provide a new proof-of-concept experiment to demonstrate its reasonableness. The key idea behind Assumption E.2 is that a larger coalition C is associated with a larger agent-type composition. Therefore, the variance of estimation during training would be larger.*

Because we are not allowed to change the number of agents in the existing environment (which is fixed with pre-trained teammates), we justify Assumption E.2 by showing the relation between the agent-type compositions and variances, because an agent-type composition is corresponding to a coalition of agents, as discussed in Remark E.3. In more detail, we would like to show that when the number of agent-type compositions increases, the variance will not decrease. We design three settings denoted as Exp A, Exp B and Exp C to capture different numbers of agent-type compositions. The number of agent-type compositions in descent would be Exp A $\geq$ Exp B $\geq$ Exp C, and the variance should be in the same order in expectation.[10]

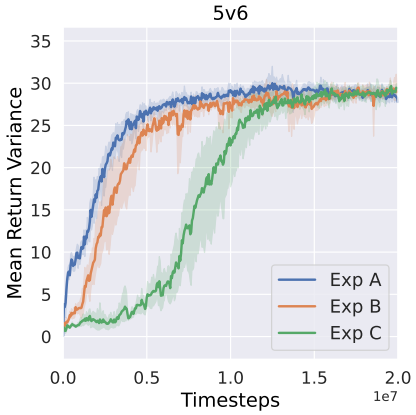


Figure 7: Empirical verification for Assumption E.2.

In Figure 7, it can be observed that the relation between variances of three setups above is almost always as the following order: Exp A $\geq$ Exp B $\geq$ Exp C, over the whole training procedure (i.e., Exp A bounded above by Exp B, and Exp B bounded above by Exp C).[11] *This result conforms to our expectation above and thus justifies the reasonableness of Assumption E.2.*

Proposition E.5 (Variance Monotonicity along the Coalition Enumeration). *Fix $s \in \mathcal{S}$ and the sequences (C_i) and (n_i) in Definition D.2. Under the non-negative autocovariance condition of Lemma E.1 and Assumption E.2, the sequences*

$$\left(\mathrm{Var}(\widehat{V}(C_i, s))\right)_{i=1}^{m} \quad \textit{and} \quad (\mathrm{Var}(G_{t:t+n_i} \mid s))_{i=1}^{m}$$

are both non-decreasing.

Proof. Since (n_i) is non-decreasing, Lemma E.1 yields a non-decreasing variance sequence $(\mathrm{Var}(G_{t:t+n_i} \mid s))_{i=1}^{m}$. Assumption E.2 gives non-decreasing $\mathrm{Var}(\widehat{V}(C_i, s))$ as $|C_i|$ increases. Since (C_i) is a non-decreasing sequence in size, as defined in Definition D.2, the result follows. □

[10]The agent type composition can be controlled by changing how the teammate types are sampled. We take the environment of 5v6 in SMAC as a testbed, where at most 5 agents in a team. Specifically, the original setting is sampling at most 4 teammates with various types, denoted as Exp A. In our new setting, we also fixed the cap of the number of teammates to sample as 3 and 2 teammates, denoted as Exp B and Exp C. In this way, the number of agent type compositions in descent would be Exp A $\geq$ Exp B $\geq$ Exp C.

[11]The reason why the variances of three settings finally converge to the same value is that models under all scenarios have reached at least an approximate solution to the optimal one. As we mentioned in the NAHT decomposition in Section 3, there could be some unseen teammate compositions (with unseen behaviours) appearing during the training process. As the training procedure proceeds, there could be more diverse teammate compositions (including both seen and unseen teammates) that lead to various evaluation results, which directly leads to increasingly higher variances. The variances due to the diverse teammate compositions dominate the variance decrease led by coalition values. The above result implies that the current approaches have not yet well learned to generalize to new teammates. However, it does not influence the contribution of this paper that improves on the existing methods, nor the conclusion we would like to show here. Instead, it demonstrates that this field is worth being investigated in the future.

F Proof for Results of Incorporating Linearity Axiom

Proposition F.1. *Given the expression of state-specific basis game values in Eq.* (4)*, applying the Linearity axiom to state-specific cooperative games of Eq.* (2) *can lead to the following equation:*

$$\phi(v_{s_t}) = \mathbb{E}_\pi \left[\sum_{n=1}^{m} k'_{C_n} \cdot \sum_{\tau=0}^{n-1} \gamma^\tau \phi(R_{t+\tau}) + \gamma^n \phi(v_{s_{t+n}}) \,\middle|\, s_t \right].$$

Proof. By Lemma C.1, a state-specific cooperative game of Eq. (2) can be equivalently represented by the form: $v_s = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k'_C \cdot v_{C,s}^{v_s(\mathcal{M})}$ with setting $z = v_s(\mathcal{M})$ and $k'_C = \frac{k_C}{v_s(\mathcal{M})}$, and $v_s(\mathcal{M}) \neq 0$.[12] Herein, the set of $v_{C,s}^{v_s(\mathcal{M})}$ become a set of new state-specific basis games and k'_C are their corresponding weightings.

Introducing a linear multidimensional operator $\phi : \mathcal{G}(s) \to \mathbb{R}^M$ (i.e., supposing that ϕ satisfies the Linearity axiom) on a state-specific cooperative game v_s defined by Lemma C.1, we derive the following formula:

$$\phi(v_s) = \sum_{\emptyset \neq C \subseteq \mathcal{M}} k'_C \cdot \phi\left(v_{C,s}^{v_s(\mathcal{M})}\right). \tag{19}$$

Recall that $v_{C_n,s_t}^{v_{s_t}(\mathcal{M})}(\mathcal{M})$ is denoted by $V(C_n, s_t)$ in Eq. (4). Substituting the game expansion of $v_{C_n,s_t}^{v_s(\mathcal{M})}$, such that

$$v_{C_n,s_t}^{v_s(\mathcal{M})} = \mathbb{E}_\pi \left[\sum_{\tau=0}^{n-1} \gamma^\tau R_{t+\tau} + \gamma^n v_{s_{t+n}} \,\middle|\, s_t \right],$$

into the Linearity axiom in Eq. (19) and applying linearity of the expectation operator, we obtain the following formula:

$$\phi(v_{s_t}) = \mathbb{E}_\pi \left[\sum_{n=1}^{m} k'_{C_n} \cdot \sum_{\tau=0}^{n-1} \gamma^\tau \phi(R_{t+\tau}) + \gamma^n \phi(v_{s_{t+n}}) \,\middle|\, s_t \right].$$

□

G Proof for Results of Incorporating Efficiency and Symmetry Axioms

Proposition G.1 (Representation of Transformed Rewards). *Given the condition that* $\sum_{i=1}^{M} \phi_i(R_t) = R_t$*, each agent's payoff allocation defined on rewards* R_t*, i.e.,* $\phi_i(R_t)$ *can be expressed as*

$$\phi_i(R_t) := R_t - \sum_{j \neq i} \left(\phi_j(v_{s_t}) - \gamma \phi_j(v_{s_{t+1}}) \right).$$

Proof. Recall that $\phi : \mathcal{G}(s) \to \mathbb{R}^M$ is a multidimensional linear transformation,[13] where M is the number of agents. We now express $\phi(R_t)$ according to the Efficiency axiom.

To satisfy the Efficiency axiom such that $\sum_{i=1}^{M} \phi_i(v_{s_t}) = v_{s_t}(\mathcal{M})$, it is reasonable to assume that $\sum_{i=1}^{M} \phi_i(R_t) = R_t$. In other words, each agent's value expansion can be expressed independently in its own $\phi_i(R_t)$, justified by Theorem 4.5.

Next, we aim to show how each $\phi_i(R_t)$ is expressed in terms of $\phi(v_{s_t})$.

It is not difficult to observe that for each agent $i \in \mathcal{M}$, we have

$$\phi_i(v_{s_t}) = \phi_i(R_t) + \gamma \phi_i(v_{s_{t+1}}).$$

[12] If $v_s(\mathcal{M}) = 0$, it implies that a team of agents unable to cooperate, contradicting the purpose of the NAHT.

[13] In contrast to the classical formulation of Dubey [1975], we additionally assume that the multidimensional transformation is linear across both the time and agent dimensions.

By the condition $\sum_{i=1}^{M} \phi_i(R_t) = R_t$, we can derive a formula such that

$$\phi_i(R_t) = R_t - \sum_{j \neq i} \phi_j(R_t).$$

Since $\phi_j(R_t) = \phi_j(v_{s_t}) - \gamma\phi_j(v_{s_{t+1}})$, we can express $\phi_i(R_t)$ as follows:

$$\phi_i(R_t) := R_t - \sum_{j \neq i} \left(\phi_j(v_{s_t}) - \gamma\phi_j(v_{s_{t+1}}) \right).$$

□

Theorem G.2. *The payoff allocation operator satisfying permutation-equivariance is a sufficient condition for the Symmetry axiom.*

Proof. Let $\mathcal{M} = \{1, \ldots, M\}$. Let $\mathcal{G}$ be the set of cooperative games $v : 2^{\mathcal{M}} \to \mathbb{R}$, and S_C be a permutation group acting on $C \subseteq \mathcal{M}$.

Let S_C act on an arbitrary $C \subseteq \mathcal{M}$ by exchanging agents:

$$(\sigma \cdot v)(C) \ := \ v(\sigma^{-1}(C)), \qquad \sigma \in S_{\mathcal{M}},\ C \subseteq \mathcal{M}.$$

Let a payoff allocation operator $F : \mathcal{G} \to \mathbb{R}^M$ output payoffs $F(v) = (F_1(v), \ldots, F_n(v))$.

Let $S_{\mathcal{M}}$ act on $\mathbb{R}^M$ by permuting coordinates: $(\sigma \cdot x)_i := x_{\sigma^{-1}(i)}$.

F is permutation-equivariant if

$$F(\sigma \cdot v) \ = \ \sigma \cdot F(v), \qquad \forall\, \sigma \in S_{\mathcal{M}},\ v \in \mathcal{G}. \tag{20}$$

Recall that agents i and j are symmetric in v if

$$v(C \cup \{i\}) = v(C \cup \{j\}), \quad \forall\, C \subseteq \mathcal{M} \setminus \{i, j\}.$$

Equivalently, the transposition $\tau = (i\ j)$ leaves the game v invariant:

$$\tau \cdot v \ = \ v. \tag{21}$$

Next, we aim to show $F_i(v) = F_j(v)$ to prove the statement in the theorem.

By Eqs. (20) and (21), we have

$$F(v) \ = \ F(\tau \cdot v) \ = \ \tau \cdot F(v).$$

Note that $\tau \cdot F(v)$ is just $F(v)$ with coordinates i and j swapped. Therefore, $F(v) = \tau \cdot F(v)$ implies the i-th and j-th entries are equal:

$$F_i(v) = F_j(v).$$

□

Theorem G.3. *Shapley Machine is an algorithm enforcing V_i to fulfil Efficiency, Symmetry and Linearity, so the V_i is the Shapley value for Dec-POMDPs.*

Proof. By construction, the Shapley Machine enforces the individual value functions V_i (payoff allocation functions) to satisfy the axioms of Efficiency, Symmetry and Linearity. Each state-specific cooperative game space $\mathcal{G}(s)$ that aims to meet the requirement of Dec-POMDPs is isomorphic to the canonical cooperative game space $\mathcal{G}$. Hence, the properties of payoff allocation functions that hold on $\mathcal{G}$ also hold on $\mathcal{G}(s)$.

Now, recall that the Shapley value is uniquely characterized as the value function that satisfies the axioms of Efficiency, Symmetry and Additivity (Theorem 2.2). Since Linearity is a stronger condition than Additivity (indeed, setting all $\alpha_i = 1$ in the definition of Linearity yields Additivity), any payoff allocation function that is Efficient, Symmetric, and Linear must coincide with the Shapley value. Therefore, the individual value functions V_i produced by the Shapley Machine are exactly the Shapley values for Dec-POMDPs. □

H Experimental Details

H.1 Implementation Details

Our algorithm is built upon POAM and IPPO. All loss functions for training the encoder-decoder model, policy networks and value networks have been remained. Please refer to Appendix B.4 for details. For conciseness, we only list the novel loss functions proposed in this paper.

H.1.1 Shapley Machine

We now describe the details about implementing our proposed algorithm, referred to as Shapley Machine. In general, our algorithm is established based on the base algorithms POAM and IPPO, with modification to fulfil all the three axioms of the Shapley value: Efficiency, Linearity and Symmetry. Since the Symmetry axiom has been realised in POAM and IPPO, we only need to fulfil the Efficiency and Linearity axioms.

Implementing the Linearity Axiom. In general, both POAM and IPPO have implemented TD(λ), which does not strictly conform to the principle of the Linearity axiom. To address this problem, we change the TD(λ) to TTD(λ), where the number of n-step return components is equal to the number of the non-empty coalition basis games, as informed by our theory. Note that in some scenarios the episode length is smaller than the number of non-empty coalition basis games. In these cases, the TTD(λ) can be reduced to the TD(λ) for the finite-horizon tasks with the episode length as T, equivalently, the TTD(λ) with $m = T$. Alternatively, we can select a value of m for each task based on extra conditions, which is left for the future work. In this paper, we set $m = T$ for the scenarios 8v9 and 10v11. For the 8v9 scenario, the episode length as 120 is not too far from the number of non-empty coalition basis games as 127. For the 10v11 scenario, the episode length is 150, while the number of non-empty coalition basis games is 511.

Implementing the Efficiency Axiom. By introducing the partial observation in Dec-POMDPs, the Eq. (8) we have derived the condition for realizing the Efficiency axiom is transformed as follows:

$$\phi_i(R_t) = R_t - \sum_{j \neq i} \left(V_j(h_{t,j}) - \gamma V_j(h_{t+1,j})\right),$$

where $V_j(h_{t,j})$ indicates an agent j's non-parameterized individual value. During the practical training procedure, the $V_j(h_{t,j})$ generated by the individual value network could be severely inaccurate in the beginning, which may result in the instability of learning. To mitigate this issue, we add an extra coefficient $\alpha \in (0, 1)$ to the term $\sum_{j \neq i} \left(V_j(h_{t,j}) - \gamma V_j(h_{t+1,j})\right)$, such that

$$\phi_i(R_t) = R_t - \alpha \sum_{j \neq i} \left(V_j(h_{t,j}) - \gamma V_j(h_{t+1,j})\right). \tag{22}$$

This coefficient α can be either manually set up as a fixed value, or implemented by a scheduler starting from 0 to some preset upper limit. The R_t in $L_\theta(h_{t,i}, a_{t,i})$ and $L_{\theta^c}(h_{t,i})$ is replaced by the above $\phi_i(R_t)$. To clarify this change, the two new losses are expressed as $\hat{L}_\theta(h_{t,i}, a_{t,i})$ and $\hat{L}_{\theta^c}(h_{t,i})$.

Furthermore, it is needed to search the underlying factorization scheme of the transition function in Dec-POMDPs, according to Theorem 4.5. To implement this requirement, we need to fulfil the following condition:

$$\sum_{i=1}^{M} V_i(h_{t,i}) = v_{h_t}(\mathcal{M}). \tag{23}$$

Here, $h_t = (h_{t,1}, \ldots, h_{t,M})$ denotes the joint history, and $v_{h_t}(\mathcal{M})$ denotes the team value conditioned on this joint history. The above equality is implemented as a regularization term during training.

To maintain consistency with the critic losses, we consider to use the λ-return denoted by $\hat{G}_t^\lambda$ to represent $v_{h_t}(\mathcal{M})$. In turn, the above equation can be expressed as

$$v_{h_t}(\mathcal{M}) = \mathbb{E}_\pi \left[G_t^\lambda | h_t\right],$$

where each $\hat{G}_{t:t+n}$ contributing to the $\hat{G}_t^\lambda$ represented in TTD(λ) is expressed as follows:

$$\begin{aligned}\hat{G}_{t:t+n} &= R_t + \gamma R_{t+1} + \cdots + \gamma^{n-1} R_{t+n-1} + \gamma^n v_{h_{t+n}}(\mathcal{M}) \\ &= R_t + \gamma R_{t+1} + \cdots + \gamma^{n-1} R_{t+n-1} + \gamma^n \sum_{i=1}^{M} V_i(h_{t+n,i}) \quad \text{(By Eq. (23))}.\end{aligned}$$

Substituting the above formula into Eq. (23), we can obtain the regularization term, referred to as the **efficiency loss** for one timestep t, as follows:

$$L^e_{\theta^c}(h_t) = \frac{1}{2}\left(\hat{G}_t^\lambda - \sum_{i=1}^{M} V_i^{\theta^c}(h_{t,i})\right)^2.$$

In summary, the total loss function of the Shapley Machine for POAM (SM-POAM) is as follows:

$$L_{\texttt{SM-POAM}} = \frac{1}{T}\sum_{t=1}^{T}\left(\sum_{i\in\mathcal{C}_t}\hat{L}_\theta(h_{t,i}, a_{t,i}) + \beta_1 \sum_{i\in\mathcal{M}}\hat{L}_{\theta^c}(h_{t,i}) + \beta_2 L^e_{\theta^c}(h_t) + L_{\theta^e,\theta^o,\theta^a}(h_{t,i}, o_{t,-i}, a_{t,-i})\right),$$

where T is the episode length; $\beta_1, \beta_2 \in (0, 1)$ are two coefficients to control the importance of the two losses; $\mathcal{C}_t$ indicates the controlled agent set at timestep t and $\mathcal{M}$ indicates the ad hoc team following the convention in Wang et al. [2025].

Similarly, the total loss function of the Shapley Machine for IPPO (SM-IPPO) is as follows:

$$L_{\texttt{SM-IPPO}} = \frac{1}{T}\sum_{t=1}^{T}\left(\sum_{i\in\mathcal{C}_t}\hat{L}_\theta(h_{t,i}, a_{t,i}) + \beta_1 \sum_{i\in\mathcal{M}}\hat{L}_{\theta^c}(h_{t,i}) + \beta_2 L^e_{\theta^c}(h_t)\right).$$

Implementation of $k'_C > 0$. As mentioned in Section C, it is necessary to fulfil $k'_C > 0$ for reaching cooperation, which can be implemented following the convention of TD(λ) in the RL [Sutton, 1988]. Specifically, the weightings for m basis games $(k'_{C_1}, k'_{C_2}, \cdots, k'_{C_m})$ are generated using a geometric distribution P_λ with the parameter $0 < \lambda < 1$, such that $k'_{C_n} = P_\lambda(n)$, resulting a tuple $((1-\lambda), (1-\lambda)\lambda, \cdots, (1-\lambda)\lambda^{m-2}, \lambda^{m-1})$. With this condition, Eq. (6) becomes the well-known **truncated TD($\boldsymbol{\lambda}$)**, shortened as **TTD($\boldsymbol{\lambda}$)** [Cichosz, 1994].

Partial Observations. In practice, a controlled agent is only able to receive an observation, following the settings of Dec-POMDPs. Therefore, an agent is required to infer the state of the environment as an individual hidden state, through the history of observations. To this end, the policy and individual networks are realized by recurrent neural networks (RNNs) (e.g., GRUs [Chung et al., 2014]), where observations or representations transformed from observations are used as inputs.

Implementation of the Symmetry Axiom. It can be observed that both POAM and IPPO are implemented by the sharing parameters, with agent ID to differentiate agent identities. Given an agent ID, its input is either individual observations or individual observations + teammate embeddings, where the teammate embeddings are transformed from individual observations through an encoder network.

Estimation of Uncontrolled Agent Individual Values. We maintain all possible controlled agent policy and individual value networks as implemented by Wang et al. [2025], given that the maximum number of controlled agents is known in the experimental settings of the NAHT, but only part of controlled agents in an episode can make decision. Thanks to the parameter sharing technique, this can be simply implemented by maintaining only a policy and/or an individual value network, with a set of agent IDs to represent and differentiate all possible controlled agents. During the training phase, the uncontrolled agents' individual values are also required to implement the Efficiency axiom, as informed in Eq. (9), but they are unknown to the controlled agents; all training paradigms that we design should be based on the view of the controlled agents. To address this problem, the uncontrolled agents' individual values are approximated by the maintained individual value function with the agent IDs of those inactive controlled agents, given that the observations of the uncontrolled agents are available for controlled agents during training.

Example H.1 (Estimating an Uncontrolled Agent's Individual Value)**.** *Consider a team $\mathcal{M} = \{c_1, c_2, u\}$ consisting of two controlled agents, $\mathcal{C} = \{c_1, c_2\}$, and one uncontrolled agent, $\mathcal{U} = \{u\}$. Suppose the implementation maintains a shared policy network and a shared individual value network, with three available controlled-agent IDs: $\{1, 2, 3\}$. In an episode, c_1 and c_2 use IDs 1 and 2, respectively, leaving ID 3 unused.*

Let $V_\psi(h, i)$ denote the shared individual value network evaluated using observation history h and agent ID i. Since the uncontrolled agent's observations are available during training, we approximate its individual value by supplying its observation history to the shared network with the unused ID:

$$\widehat{V}_u(h_{t,u}) := V_\psi(h_{t,u}, 3).$$

As a result, the controlled agent associated with ID 3 does not participate in the episode, but its ID is used to obtain a proxy value estimate for u.

H.1.2 POAM and IPPO

The implementations of POAM and IPPO have been detailed in Appendix B.4.

In summary, the total loss function of POAM is as follows:

$$L_{\texttt{POAM}} = \frac{1}{T}\sum_{t=1}^{T}\left(\sum_{i\in\mathcal{C}_t} L_\theta(h_{t,i}, a_{t,i}) + \beta_1 \sum_{i\in\mathcal{M}} L_{\theta^c}(h_{t,i}) + L_{\theta^e,\theta^o,\theta^a}(h_{t,i}, o_{t,-i}, a_{t,-i})\right).$$

The total loss function of IPPO is as follows:

$$L_{\texttt{IPPO}} = \frac{1}{T}\sum_{t=1}^{T}\left(\sum_{i\in\mathcal{C}_t} L_\theta(h_{t,i}, a_{t,i}) + \beta_1 \sum_{i\in\mathcal{M}} L_{\theta^c}(h_{t,i})\right).$$

H.1.3 Banzhaf Machine

The overall loss function and implementation of the Banzhaf Machine closely follow those of POAM and IPPO. The only distinction is that Banzhaf Machine employs the TTD(λ) in place of the TD(λ) that is applied in POAM and IPPO. This implies that Banzhaf Machine uses $\phi_i(R_t) = R_t$ as its implementation, but this does not mean that it is the only implementation.

H.2 Experimental Domains

We now briefly introduce the experimental domains for running experiments. If one would like to know more about details, please refer to Wang et al. [2025].

H.2.1 MPE Predator Prey (MPE-PP)

The MPE Predator Prey (MPE-PP) environment is a predator-prey task implemented within the Multi-Agent Particle Environment (MPE) framework. It simulates interactions within a two-dimensional space populated by two static obstacles, where three pursuer agents must cooperate to capture a single adversarial evader. A successful capture is defined as at least two pursuers simultaneously colliding with the evader, upon which the pursuers receive a positive reward of +1. If the capture is unsuccessful, no reward is granted. This environment is designed to test the ability of agents to coordinate under spatial and dynamic constraints. The length of an episode is $T = 100$,

H.2.2 The StarCraft Multi-Agent Challenge (SMAC)

The StarCraft Multi-Agent Challenge (SMAC) serves as a benchmark suite for evaluating CMARL algorithms in partially observable settings. Built on top of the StarCraft II game engine, SMAC presents a variety of micromanagement tasks where each agent (e.g., a Marine or Stalker) operates based on limited local observations and must coordinate actions with teammates to overcome enemy units. In this work, we focus on four specific SMAC scenarios: **5v6**: five allied Marines versus six enemy Marines, **8v9**: eight allied Marines versus nine enemy Marines, **10v11**: ten allied Marines versus eleven enemy Marines, **3s5z**: three allied Stalkers versus five enemy Zealots. At each timestep, agents receive a shaped reward proportional to the damage they inflict on opponents, along with

bonus rewards of 10 points for each enemy defeated and 200 points for achieving overall victory by eliminating all adversaries. The total return is normalized so that the maximum achievable return in each scenario is 20. The action space in SMAC is discrete, enabling each agent to choose actions such as attacking a particular enemy, moving in a specific direction, or remaining idle. The length of an episode varies across different scenarios: 3sv5z with $T = 250$, 5v6 with $T = 70$, 8v9 with $T = 120$ and 10v11 with $T = 150$.

H.3 Evaluation Metrics

For in-distribution evaluation, the mean return (or winrate) is computed over E randomly sampled episodes. In each episode, we form a joint policy $\pi^{(\mathcal{M})}$ by sampling N agents uniformly from $\mathcal{C}$ and the remaining $M - N$ agents from $\mathcal{U}$. For the out-of-distribution (OOD) evaluation, the M–N score [Wang et al., 2025] is applied. Given a set of controllable agents $\mathcal{C}$ and uncontrolled agents $\mathcal{U}$, the M–N score measures how well mixed teams of these agents cooperate in the NAHT setting. The score is computed exhaustively by changing the number of controllable agents included in the team. Specifically, for each $N \in \{1, \ldots, M-1\}$, we form a joint policy $\pi^{(\mathcal{M})}$ by sampling N agents uniformly from $\mathcal{C}$ and the remaining $M - N$ agents from $\mathcal{U}$. The mixed teams are then evaluated on the task for E episodes. This procedure yields $(M-1) \cdot E$ returns (or winrates) in total, whose average defines the M–N score.

In this experiment, each convention of an uncontrolled agent is a random seed that pretrains it. For the OOD evaluation on the unseen conventions, the M–N score is computed by averaging the per-convention M–N scores across 4 other conventions for each agent type. For the OOD evaluation on the unseen agent types, the per-convention M–N score is computed by averaging the per-convention M–N score across 5 conventions for each unseen agent type. All results are obtained by first computing each metric with 128 episodes and then averaging these per-seed metrics across 5 random seeds, with 95% confidence intervals.

H.4 Experimental Settings

The training procedure of the NAHT process is briefly introduced here. For more details, please refer to Wang et al. [2025]. For each scenario (e.g. MPE, 3sv5z, 5v6, 8v9 and 10v11), there are five groups of pretrained agents (e.g. IQL, MAPPO, VDN, QMIX and IPPO) acting as the uncontrolled agents. Given that each task specifies a fixed total number of agents M, the number of controlled agents is N. For each episode evaluation, the number of uncontrolled agents $M - N$ is sampled, and then a group of pretrained agents is sampled.[14] The distribution for all sampling procedures are modelled as the multinational distribution with no replacement. Each uncontrolled agent executes the greedy policy in both training and testing procedures. In contrast, each controlled agent executes the on-policy sampling via the parameterized policy during the training procedure, while the greedy policy during the testing procedure.

H.5 Hyperparameter Settings

Since our algorithm is established based on POAM and IPPO, most hyperparameter settings follow that in Wang et al. [2025]. First, the actors and critics are implemented in recurrent neural networks, with full parameter sharing. Specifically, they are implemented by two fully connected layers followed by a GRU layer and an output layer. Each layer has 64 dimensions with a ReLU activation function, and layer normalization is applied. The encoder-decoder networks for inferring agent characteristics are also implemented in parameter sharing. The encoder is implemented by a GRU layer, followed by a fully connected layer with a ReLU activation function and an output layer. The decoder is implemented by two fully connected layers with ReLU activation functions, followed by an output layer. Adam Optimizer is used to train all models. The detailed hyperparameters for experiments are shown in Tables 1 and 2. For the scenarios such as MPE, 3sv5z and 5v6, the values of m are set as the number of non-empty coalition basis games. For the scenarios such as 8v9 and 10v11, the values of m are simply set as the length of an episode (see Appendix H.1 for more details). Note that the term $\sum_{j \neq i} (V_j(s_t) - \gamma V_j(s_{t+1}))$ in shaped rewards is standardised to match the scales of standardised

[14] Note that this is still a special case of openness. For a varying number of controlled agents, the number of uncontrolled agents is also varied in correspondence, constrained by the total number of agents related to the task specifications.

rewards. To stabilize learning in large-scale scenarios, we add a value loss clip to Shapley Machine and Banzhaf Machine for POAM on both 8v9 and 10v11, and for IPPO on 10v11 only.

Table 1: Key hyperparameters of Shapley and Banzhaf Machine: λ indicates the parameter for the weighting functions of TTD(λ), m indicates the number of basis games, α controls the importance of the term $\sum_{j \neq i} (V_j(s_t) - \gamma V_j(s_{t+1}))$ in Eq. (22), β_1 controls the importance of the critic loss $\sum_{i \in \mathcal{M}} \hat{L}_{\theta^c}(h_{t,i})$, and β_2 controls the importance of the efficiency loss $L^e_{\theta^c}(h_t)$.

Algorithm	Task	λ	m	α	β_1	β_2
Shapley Machine	MPE-PP	0.85	7	0.01	0.5	0.01
	3sv5z	0.85	7	0.01	0.5	0.01
	5v6	0.85	31	0.01	0.5	0.001
	8v9	0.95	120	0.01	0.5	0.001
	10v11	0.95	150	0.01	0.5	0.001
Banzhaf Machine	MPE-PP	0.85	7	N/A	0.5	N/A
	3sv5z	0.85	7	N/A	0.5	N/A
	5v6	0.85	31	N/A	0.5	N/A
	8v9	0.95	120	N/A	0.5	N/A
	10v11	0.95	150	N/A	0.5	N/A

Table 2: Common hyperparameters of RL settings.

Hyperparameter	Value
LR	0.0005
Epochs	5
Minibatches	1
Buffer size	256
Entropy coefficient	0.05
Clip	0.2
ED LR	0.0005
ED epochs	1
ED Minibatches	1
Optim_alpha (Adam)	0.99
Optim_eps (Adam)	0.00001
Use_obs_norm	True
Use_orthogonal_init	True
Use_adv_std	True
Standardise_rewards	True
num_parallel_envs	8

H.6 Computational Resources

All experiments are conducted on Intel Xeon Gold 6230 CPUs and Nvidia V100-SXM2 GPUs. Each experiment run on MPE takes approximately 7 hours, utilizing 20 CPU cores and 1 GPU. Each experiment run on SMAC takes between 8 and 19 hours, utilizing 30 CPU cores and 1 GPU. All experiments are trained with 20M timesteps. The post-training evaluation for each scenario takes between 6 and 16 hours, utilizing 8 CPU cores and 1 GPU.

I Additional Experiments

I.1 Empirical Evidence for the Strength of the Axiomatic Framework

We now discuss the phenomenon observed from the performance under stochastic policy exploration during training. Discussing this phenomenon is meaningful: When a machine learning algorithm is deployed and has to adapt to real-world environments through online learning due to the mismatch between simulators and real-world environments, it is expected that random exploration diminishes rapidly over time to avoid the damage on real-world environments (e.g. physical systems).

Recall that the PPO family algorithms implement exploration by the natural stochastic policy sampling from the learned policy distribution and maximizing policy entropy. As seen from Figure 8, either Shapley Machine or Banzhaf Machine can outperform their base algorithm counterparts with a large margin. The reasons are as follows: As shown in Figure 9, policy entropy decreases rapidly over the course of training (except for the IPPO implementation on 3sv5z, which underperforms due to incompatibility between IPPO and the task). This indicates that the learned policies become more deterministic than those of the base algorithms, despite all methods being optimized with an entropy-maximization objective. A possible explanation for this phenomenon is illustrated in Figure 10: The critic losses of Shapley and Banzhaf Machines decrease faster than that of their base algorithm counterparts. This provides evidence for the advantage of structured individual value functions under our proposed axiomatic framework.

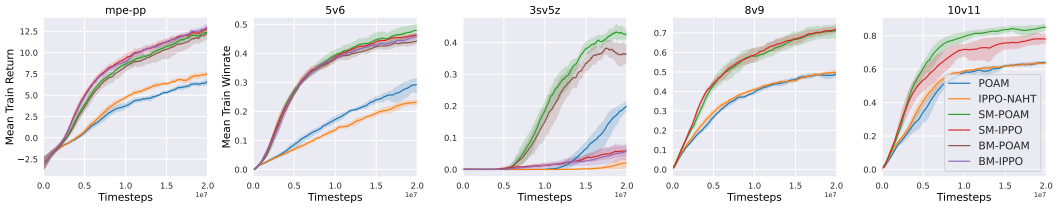


Figure 8: Performance under the stochastic exploration policy during training in NAHT.

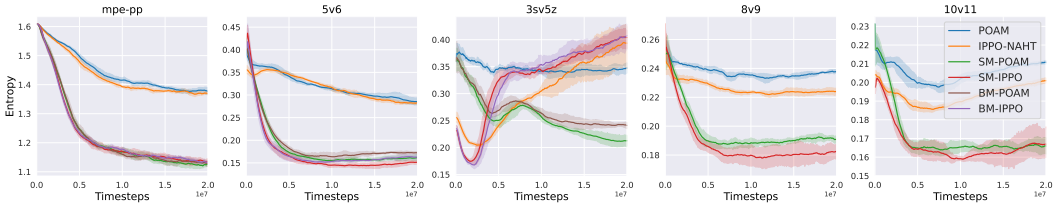


Figure 9: Policy entropy across scenarios during training in NAHT.

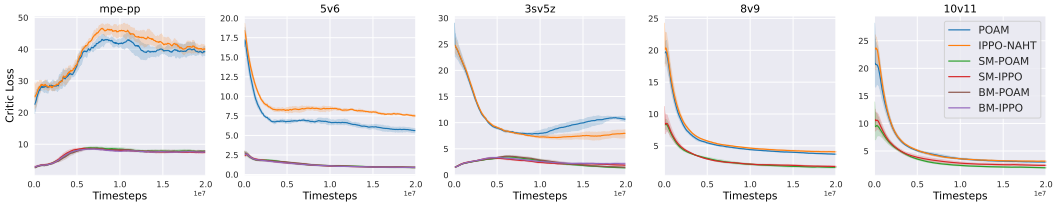


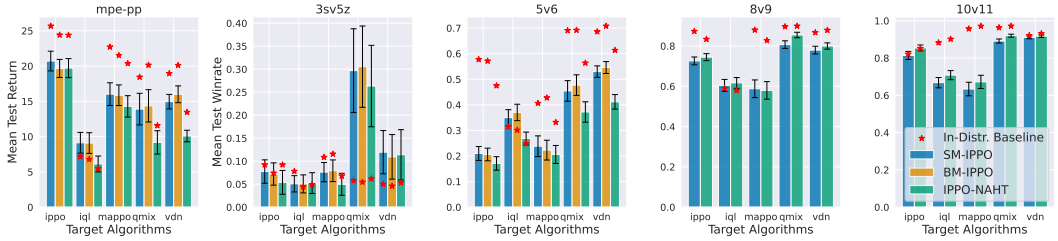
Figure 10: Critic loss across scenarios during training in NAHT.

I.2 Analyzing Effects of Efficiency Axioms

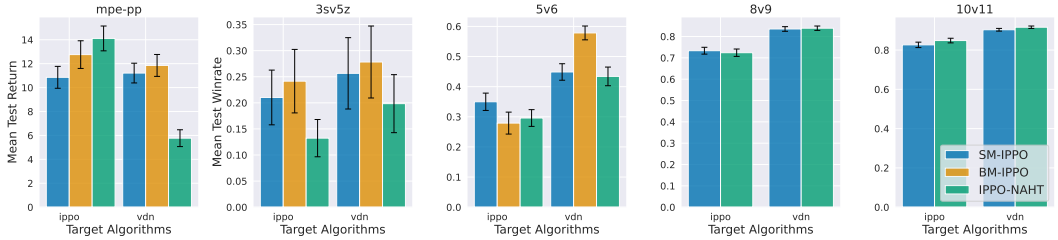
We now demonstrate the association between the changing of the shaping rewards and the response of corresponding testing returns. As shown in Figure 13, the shaped reward increases in the beginning and then gradually reaches a plateau. The efficiency loss also exhibits the similar phenomenon. As shown in Figure 14, it can be observed that when instability happens during training as evidenced by the sudden changes of test returns, the shaped reward can consistently manifest this situation (as highlighted in red vertical dashed lines). This feature is crucial in the NAHT, since it could be a frequent and typical case when an unseen agent appears in the environment, which may perturb the training stability. This justifies the effectiveness of the shaping rewards $\phi_i(R_t)$ we propose to fulfil the Efficiency axiom. More importantly, this highlights the importance and necessity of the Efficiency axiom to perceive and measure the changing of environments.

I.3 Extra Results of OOD Evaluation

We also conducted experiments on evaluating the OOD performance of Shapley Machine and Banzhaf Machine on the base algorithm IPPO (IPPO-NAHT). As seen from Figure 11a, Banzhaf Machine



(a) Evaluation on unseen conventions (identical algorithms with different random seeds). The star denotes the in-convention baseline, and the histograms show performance on unseen conventions.



(b) Evaluation on unseen agent types. Train set: MAPPO, QMIX, IQL; test set: IPPO, VDN.

Figure 11: Out-of-distribution (OOD) evaluation after training, assessing unseen conventions and unseen agent types during training, between learned agents (IPPO-NAHT, SM-IPPO and BM-IPPO) and uncontrolled agent types (IPPO, VDN, IQL, QMIX and MAPPO). The performance is evaluated by averaging all pairings of N controlled and $M - N$ uncontrolled agents and their corresponding random seeds, called M–N score [Wang et al., 2025].

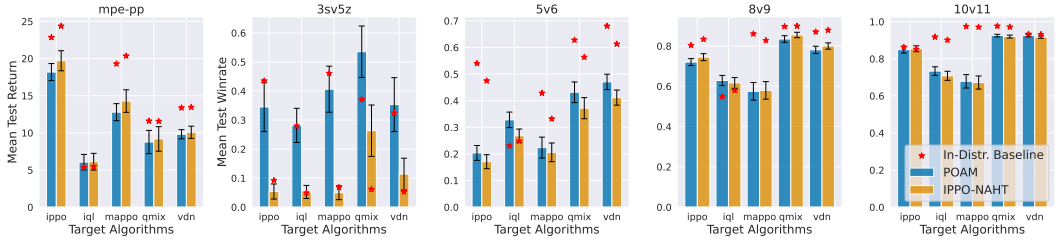
is competitive to Shapley Machine on dealing with unseen conventions. Although it seems that the result is opposite to the conclusion drawn from the POAM related results, the underlying reason could be the general weaker performance of IPPO than POAM due to the lack of agent modelling, as shown in Figure 15. This could inhibit the full capability of Shapley Machine. As seen from Figure 11b, the general performance of Banzhaf Machine is better than Shapley Machine. This is consistent with the conclusion drawn from the POAM related results. In Figure 12, we show additional comparisons between algorithms: POAM vs. IPPO, SM-POAM vs. SM-IPPO and BM-POAM vs. BM-IPPO. We find no strong evidence that the axioms are sensitive to the choice of base algorithm.

I.4 Potential Instability for Large-Scale Scenarios

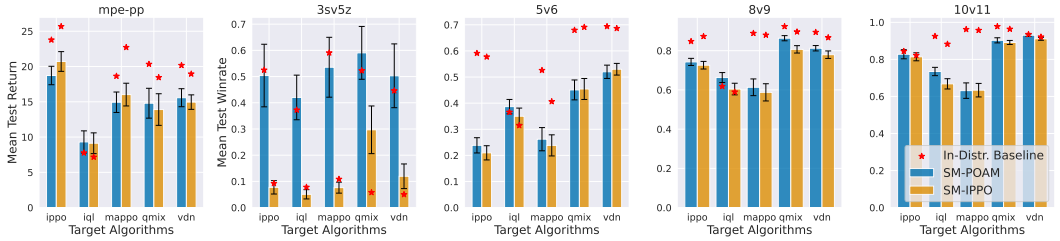
During our repeated trials, we observe that the performance of Shapley Machine is not stable in large-scale scenarios (e.g. 8v9 and 10v11), compared with in small-scale scenarios (e.g. MPE-PP, 3sv5z and 5v6). We now analyze the potential reason behind this phenomenon. As observed from Figures 16 and 17, it can be confirmed that the instability of shaped rewards is the key reason to cause the instability of learning procedure for large-scale scenarios. According to the functionality of shaping rewards discussed in Appendix I.2, we hypothesize that the instability could be caused by the unstable changing of value difference in the shaping reward: $\sum_{j\neq i}\left(V_j(s_t) - \gamma V_j(s_{t+1})\right)$. Specifically, when the number of agents increases, the accumulating prediction error of individual value functions will be amplified.

To mitigate the above issue, we have added the value loss clip to stabilize the learning procedure of individual value functions for the large-scale scenarios such as 8v9 and 10v11. As seen from Figure 3, even with the value loss clip the instability of SM-IPPO is still out of control. This could be the cascading effect caused by the lack of agent modelling in contrast to SM-POAM, the instability of which is far beyond the capability of value loss clip. This reflects the value of agent modelling in POAM for tackling the large-scale multi-agent scenarios.

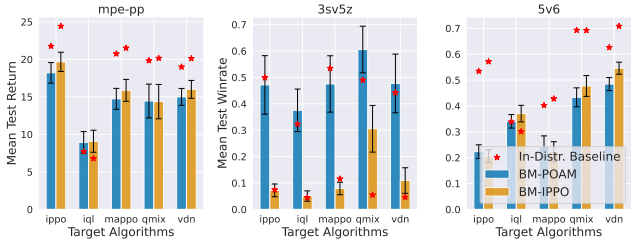
As seen from Figure 18, the learning procedures with the value loss clip for the SM-POAM apparently perform more stably than those without. This verifies our initial hypothesis. Because this strategy will slow down and even hinder the performance for the other three scenarios, we have not posted this



(a) Evaluation on unseen agent types between POAM and IPPO (IPPO-NAHT).



(b) Evaluation on unseen agent types between SM-POAM and SM-IPPO.



(c) Evaluation on unseen agent types between BM-POAM and BM-IPPO. Since the implementation of Banzhaf Machine is equivalent to its counterpart baseline algorithms for 8v9 and 10v11 due to approximating the number of non-empty coalition basis games as the episode length, we ignore their evaluation here.

Figure 12: Out-of-distribution (OOD) evaluation after training, assessing unseen conventions during training, between learned agents (IPPO-NAHT, SM-IPPO, and BM-IPPO) and uncontrolled agent types (IPPO, VDN, IQL, QMIX and MAPPO). The performance is evaluated by averaging all pairings of N controlled and $M - N$ uncontrolled agents and their corresponding random seeds, called M–N score [Wang et al., 2025].

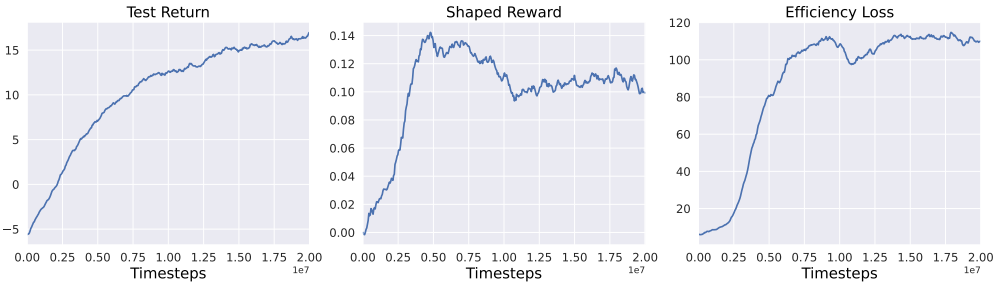


Figure 13: One run of MPE to show the variation of test return, shaped reward and efficiency loss.

as a common strategy. We believe this deserves further investigation in the future before any claims can be made about its general performance.

I.5 OOD Evaluation Across Different Numbers of Controlled Agents

We report the results on unseen conventions and agent types for each specific number of controlled agents N in Figures 19-22, by fixing the N value in the M–N score. We would like to highlight that the NAHT problem is a far more difficult task to deal with than the CMARL problem. As a result,

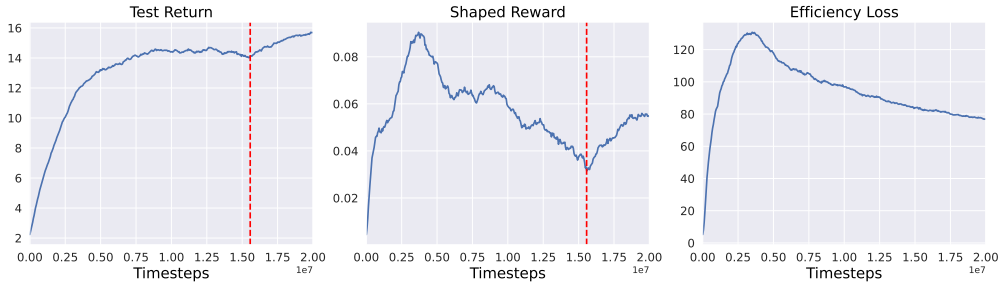


Figure 14: One run of 5v6 to show the variation of test return, shaped reward and efficiency loss.

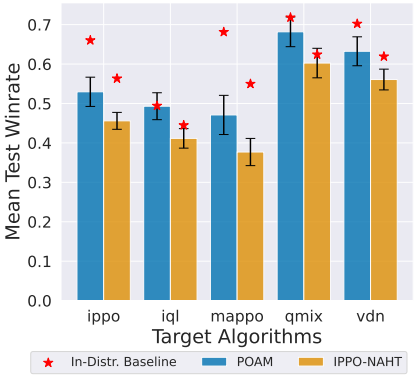


Figure 15: Average evaluation performance on unseen conventions across 4 SMAC tasks. The star denotes the in-convention baseline, and the histograms denotes the unseen conventions.

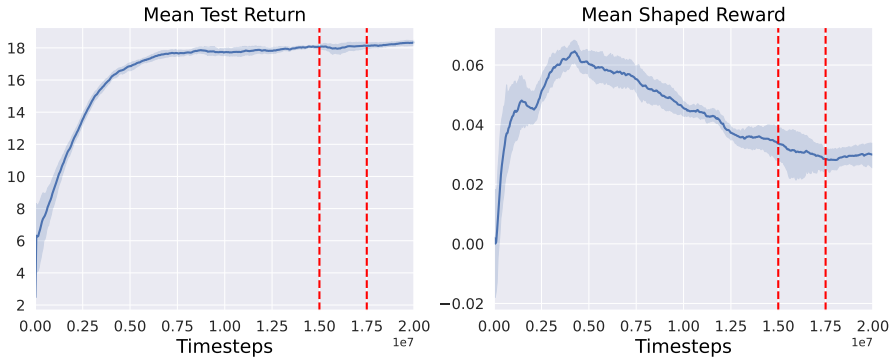


Figure 16: A case of 8v9 with SM-POAM to show the instability of learning progress via the mean shaped rewards. The range bounded by two red dashed lines shows the fluctuation of the learning process.

our work is only a starting point in this direction, and we believe that the potential stability issue is due to the lack of more exact axioms tailored to the concrete task to face. However, following the framework and principle we develop in this work, this can be improved by considering more axioms that we do not cover in this work, from either the game-theoretical perspective or the physical world.

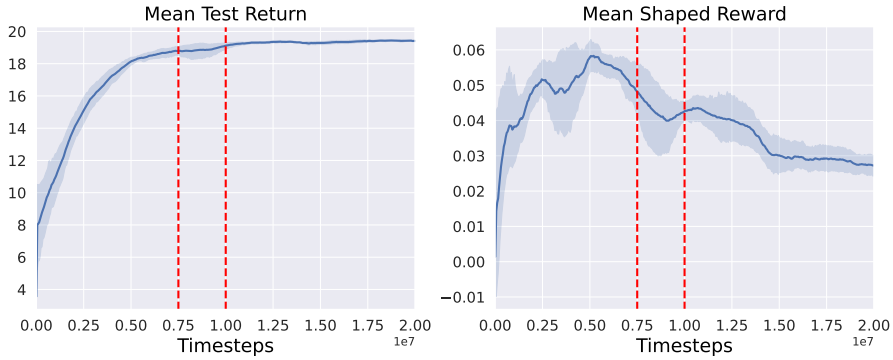


Figure 17: A case of 10v11 with SM-POAM to show the instability of learning progress via the mean shaped rewards. The range bounded by two red dashed lines shows the fluctuation of the learning procedure.

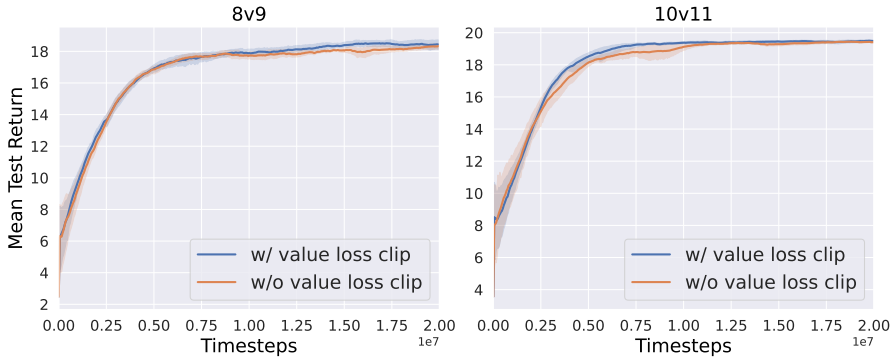


Figure 18: Comparison between the learning procedure of SM-POAM with and without the value loss clip in the 8v9 and 10v11 scenarios.

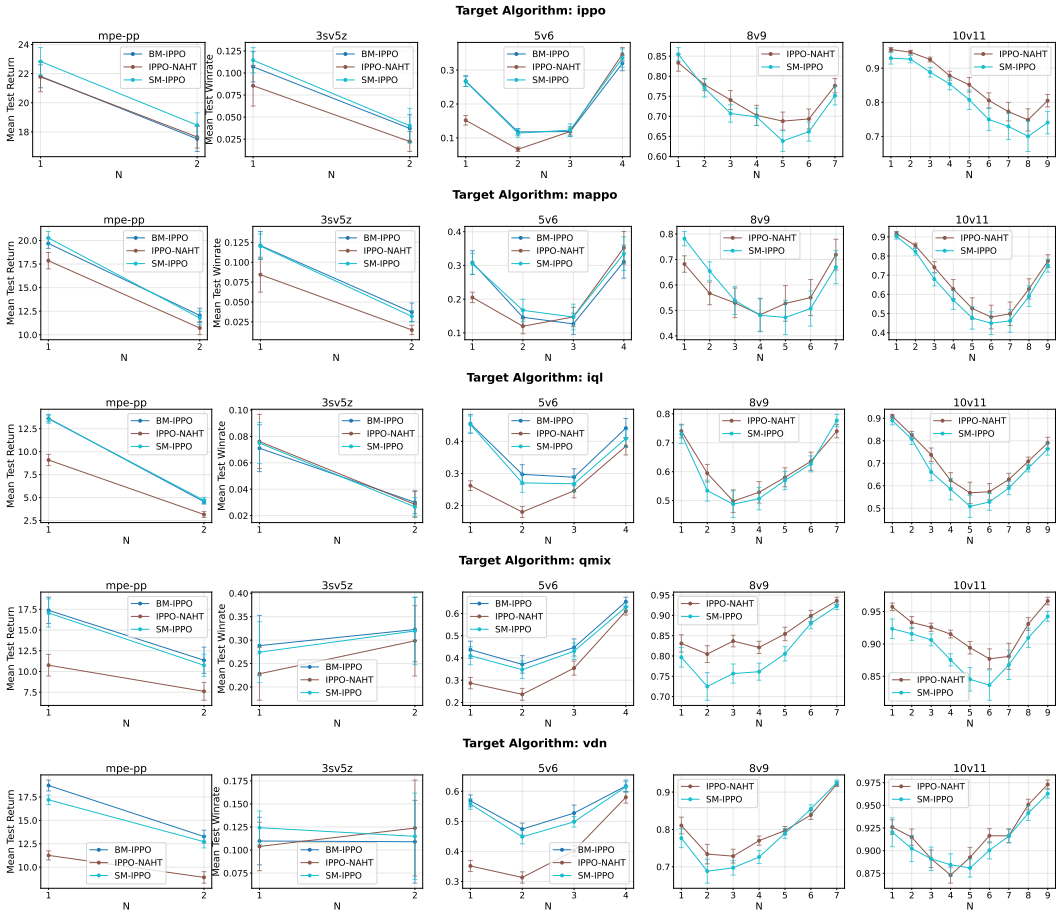


Figure 19: Out-of-distribution (OOD) evaluation after training, assessing unseen conventions during training, between learned agents (IPPO-NAHT, SM-IPPO and BM-IPPO) and uncontrolled agent types (IPPO, VDN, IQL, QMIX and MAPPO).

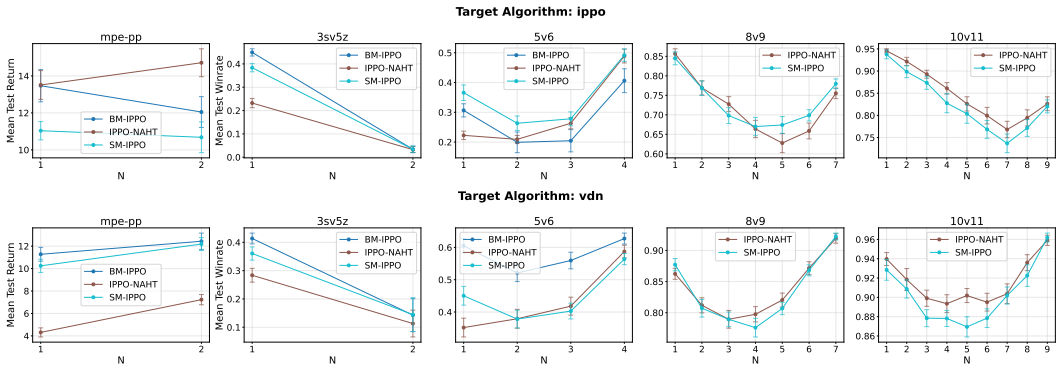


Figure 20: Out-of-distribution (OOD) evaluation after training, assessing unseen agent types during training, between learned agents (IPPO-NAHT, SM-IPPO and BM-IPPO) and unseen uncontrolled agent types (IPPO and VDN).

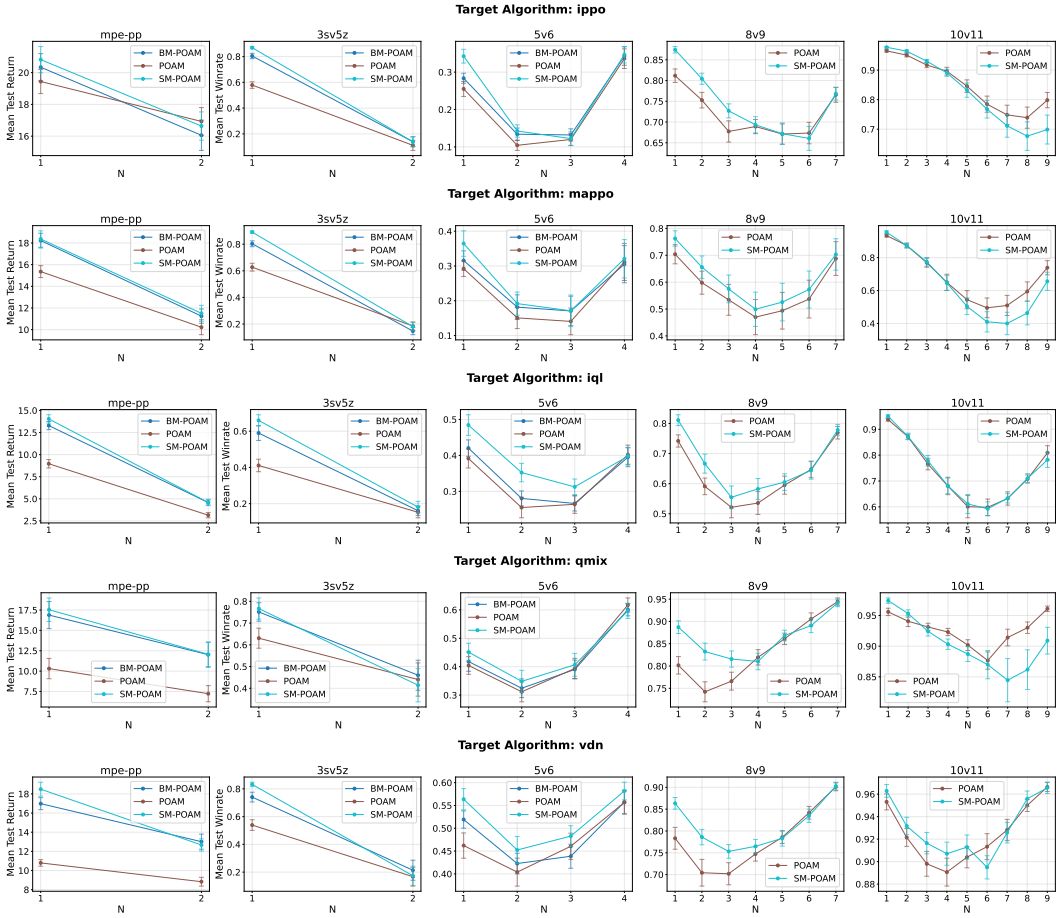


Figure 21: Out-of-distribution (OOD) evaluation after training, assessing unseen conventions during training, between learned agents (POAM, SM-POAM, and BM-POAM) and uncontrolled agent types (IPPO, VDN, IQL, QMIX and MAPPO).

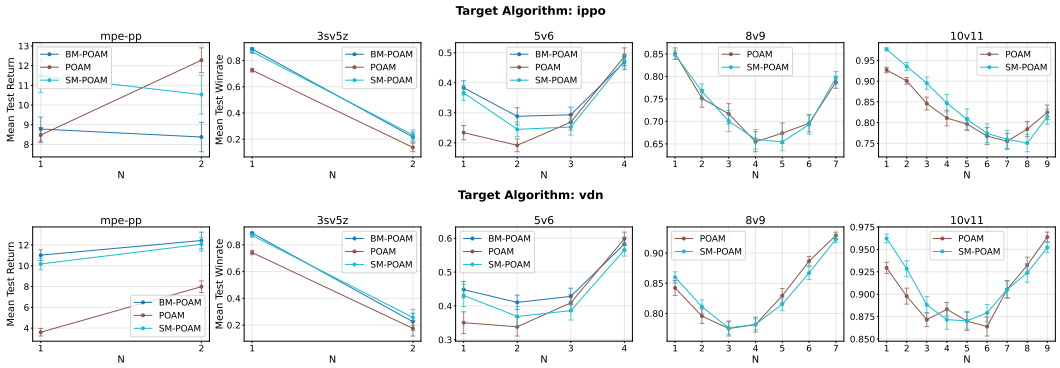


Figure 22: Out-of-distribution (OOD) evaluation after training, assessing unseen agent types during training, between learned agents (POAM, SM-POAM and BM-POAM) and unseen uncontrolled agent types (IPPO and VDN).

J Broader Impact

This paper introduces an axiomatic framework for learning in open multi-agent systems, with potential applications in domains such as smart grids, transportation networks, and multi-robot systems. By providing a principled foundation for coordinating with partially controllable and potentially unknown agents, the proposed approach may improve the reliability, adaptability, and generalisation of multi-agent systems deployed in real-world environments.

J.1 Positive Impacts

A key potential benefit lies in infrastructure systems, such as power distribution networks, where coordination among heterogeneous and partially controllable agents (e.g., distributed energy resources, consumer devices) is essential. More robust and generalisable learning algorithms could improve efficiency, reduce energy waste, and enhance grid stability. Similarly, in autonomous transportation and logistics, improved coordination with unknown or non-cooperative agents may lead to safer and more efficient traffic management and resource allocation.

The proposed axiomatic perspective may also contribute to trustworthy AI, as it provides interpretable structural constraints on learning algorithms. By grounding value function design in well-established principles from cooperative game theory, the framework may improve transparency and facilitate reasoning about agent behaviour in safety-critical settings.

J.2 Potential Risks and Negative Impacts

Despite these benefits, the deployment of open multi-agent systems introduces several risks. First, improved coordination capabilities could be applied in strategic or adversarial settings, such as automated market manipulation or competitive multi-agent environments, where agents learn to exploit or collude with others. The ability to generalise to unseen agent behaviours may amplify such risks.

Second, systems trained under the proposed framework may exhibit unintended coordination behaviours when interacting with human users or legacy systems, especially if the underlying assumptions about cooperation do not hold. In safety-critical domains, such behaviours could lead to undesirable or unpredictable outcomes.

Third, while the framework introduces structural constraints, it does not inherently guarantee fairness or alignment with human values. The allocation of value among agents, even when principled from a game-theoretic perspective, may not correspond to socially desirable outcomes in all contexts.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: All key contributions covered in the abstract and introduction can be well matched to the theoretical results described in Sections 3 and 4, as well as experimental results as shown in Section 5. The scope of this paper is restricted to n-agent ad hoc teamwork, an open multi-agent system problem.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We have discussed the key limitation of this work in the Section 6. The key limitation lies in a parameter of our algorithm that could not well adapt to large-scale agent teams, but it can be improved by relaxing the assumption of our theoretical framework.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: All theoretical claims are numbered and we have clarified clearly the corresponding sections in Appendix that include their proofs. In short words, all proofs and justifications for assumptions are given in Appendices C to G.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: All implementation details, evaluation metrics and hyperparameters for reproducing results are disclosed in Appendices H.1, H.3 and H.5.

Guidelines:

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

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: The opensource code is uploaded as supplemental materials, along with instructions to reproduce key results.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: All training and testing details are shown in Appendices H.3 and H.5.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: All experimental results are demonstrated following mean with 95% CIs over 5 random seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: All computer resources for experiments are disclosed in Appendix H.6.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: This paper does not violate any term in the code of ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We have clearly discussed both positive and potentially negative impacts in Appendix J.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: Our work is mainly about theoretical aspects, with no relation to such issues.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All baseline algorithm (codes) we used for experiments have been credited by citing the corresponding work.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: The code for training has been with a clear documentation.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: Our work has no relation to this term.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper is with no relation to this term.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: The core part of research does not involve LLMs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.