Neural Networks 203 (2026) 109103

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Full Length Article

HIVE: A hypergraph-based game-theoretic interactive value decomposition engine for multi-lateral agents collaboration

Changdong Zhou, Ruofan Hu, Naiyao Wang, Ke Song, Hongbo Liu *

College of Artificial Intelligence, Dalian Maritime University, Dalian, 116026, China

ARTICLE INFO

Keywords:
Multi-agent reinforcement learning
Value decomposition
Hypergraph game theory
Individual-global-max

ABSTRACT

Multi-lateral agents collaboration plays a crucial role in realizing collective intelligence. However, it remains constrained due to challenges such as capturing higher-order dependencies in dynamic interactions and imbalanced credit assignment. In this paper, we propose the Hypergraph-based game-theoretic Interactive Value decomposition Engine (HIVE). The engine employs the hypergraph-weighted Position value, a game-theoretic allocation rule, to decompose the joint state-action value function into three components: individual state-action values, a global coordination baseline, and a dynamic cooperation residual. To implement this theoretical decomposition in practice, we design three neural networks to approximate these corresponding components. We further prove that this game-theoretic allocation rule admits a marginal utility function, which guarantees payoff allocation efficiency. And this allocation rule satisfies invariance to hyperedge-redundant payoffs. We also establish the necessary and sufficient conditions for HIVE to satisfy the Individual-Global-Max (IGM) principle. Experimental results show that HIVE converges consistently to optimal joint actions in matrix games and achieves an average win rate of 87.25% across SMAC and its SMACv2 extension, outperforming state-of-the-art value decomposition methods. This demonstrates the effectiveness of HIVE in enabling balanced credit assignment and reliable coordination.

1. Introduction

Recent years have witnessed remarkable progress in cooperative Multi-Agent Reinforcement Learning (MARL), where the central objective is to maximize global return through distributed decision-making mechanisms (Gorsane et al., 2022; Phan et al., 2021). This paradigm underpins a wide range of critical applications, including multi-robot control (Huang et al., 2025; Moldagalieva et al., 2024; Wang et al., 2024), autonomous driving coordination (Liu et al., 2024; Parada et al., 2025; Wang et al., 2022a), and large-scale online games (Mai et al., 2024; Shen et al., 2022; Ye et al., 2020). A widely adopted solution is the Centralized Training with Decentralized Execution (CTDE) framework (Su et al., 2024; Zhang et al., 2024), where multi-agent value decomposition (Hu et al., 2025; Li et al., 2024d) has emerged as a representative research direction. By decomposing the global Q-value into agent-specific utilities, these approaches provide actionable signals for local decision-making.

However, collaborative decision-making among multiple agents remains constrained by two fundamental challenges. First, the complexity of multi-lateral interactions makes it difficult to capture higher-order dependencies, resulting in inaccurate characterizations of group dynamics and hindered coordinated decision-making. Second, imbalanced credit assignment arises when the mapping between global rewards and individual contributions is ambiguous, leading to distorted utility evaluation and inequitable distribution of cooperative returns.

In this paper, we propose a Hypergraph-based game-theoretic Interactive Value decomposition Engine (HIVE), for enabling cooperation among multiple agents. As illustrated in Fig. 1, hypergraphs provide a natural way to group multiple agents into high-order structures, while the extension to hyperedge coalitions further captures overlapping and dynamic interactions. Specifically, HIVE decomposes the joint state-action value into three components: individual state-action values, a global cooperative baseline, and dynamic coordination residuals. This decomposition is achieved through the hypergraph-weighted Position value and provides a theoretically exact factorization of the joint value function rather than an approximate decoupling. To make this factorization operational in practice, we approximate the three components using neural networks: an agent network, Global Baseline Integrator (GBI), and the Dynamic Coordination Integrator (DCI). The agent network employs recurrent neural networks to map partial

* Corresponding author.
E-mail addresses: zhou.changdong@dlmu.edu.cn (C. Zhou), ruofanhu@dlmu.edu.cn (R. Hu), wny@dlmu.edu.cn (N. Wang), kesong@dlmu.edu.cn (K. Song), lhb@dlmu.edu.cn (H. Liu).

https://doi.org/10.1016/j.neunet.2026.109103
Received 1 October 2025; Received in revised form 1 May 2026; Accepted 11 May 2026
Available online 13 May 2026
0893-6080/© 2026 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

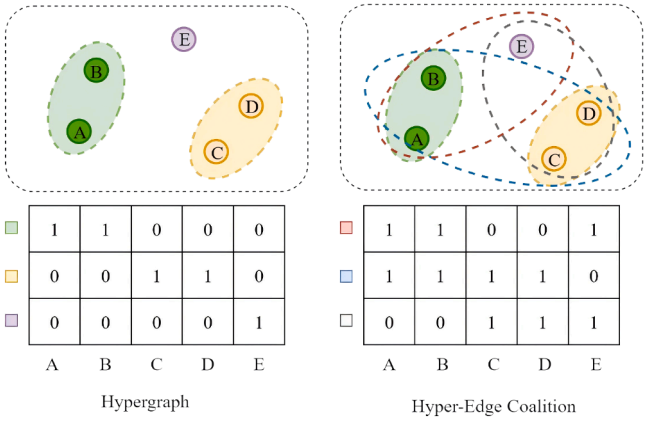


Fig. 1. The left part illustrates a hypergraph consisting of three hyperedges, which connect nodes (A, B), (C, D), and (E), respectively. The right part presents the corresponding hyper-edge coalition structure, where each hyperedge can form an individual coalition, and exemplary pairwise combinations of hyperedges along with their associated adjacency matrix are also shown.

observations into individual state-action values. Since both GBI and DCI rely on structured representations of multi-agent interactions, we introduce an attention-driven hypergraph constructor, which transforms agents' partial observations and action histories into dynamic hypergraph topologies. Subsequently, GBI takes the hypergraph topology together with the joint action-observation history as input to generate the global coordination baseline, thereby assessing the overall situation. Meanwhile, DCI samples hyperedge coalitions via a Monte Carlo sampler and applies a corrected marginal contribution estimator to approximate the dynamic cooperative residual.

The main contributions of this work are as follows:

1) We introduce HIVE, a value decomposition framework grounded in the hypergraph-weighted Position value, which rigorously decomposes the joint Q-value into individual state-action values, a global baseline, and a cooperative residual.
2) We design two neural modules (GBI and DCI) for scalable approximation of the global cooperative baseline and dynamic cooperation residual, with an attention-driven hypergraph constructor that captures high-order interactions.
3) We derive the hypergraph-weighted Position value to ensure payoff invariance even under redundant hyperedges, which admits a marginal utility function guaranteeing payoff allocation efficiency. Its necessary and sufficient conditions are proved theoretically to satisfy the IGM principle.

2. Related work

In this section, we provide a concise review of three key enabling technologies relevant to our investigation: value decomposition methods, graph network approaches, and game-theoretic approaches.

2.1. Value decomposition methods

Value decomposition constitutes a pivotal research direction in MARL, demonstrating remarkable advantages in addressing the credit assignment problem inherent in multi-agent collaborative systems (Li et al., 2022; Liu et al., 2025). Sunehag et al. proposed representing the global value Q_{tot} as a straightforward summation of individual agents' Q-values (Sunehag et al., 2018). However, this approach relies on a strict additivity assumption. Rashid et al. introduced a framework that estimates Q_{tot} through monotonic nonlinear combinations of individual Q_i values to address this limitation (Rashid et al., 2020b). Subsequently, numerous variants of the QMIX approach have emerged, including W-QMIX (Rashid et al., 2020a), QTRAN (Son et al., 2019), and TransfQMix (Gallici et al., 2023). Li et al. developed an optimistic teacher model to guide value decomposition, which refines the global action-value function by predicting optimal joint actions (Li et al., 2024a). Although existing methods have made progress in value decomposition, they remain limited by structural constraints, which render the decomposition merely sufficient. Therefore, exploring a general decomposition framework that characterizes both necessary and sufficient conditions for value function decomposability remains a promising direction for further research.

2.2. Graph network approaches

Graph Neural Networks (GNNs) have achieved substantial progress in relational modeling, offering novel perspectives for MARL by representing multi-agent relationships as graph structures (Goeckner et al., 2024; Naderializadeh et al., 2020; Zhang et al., 2025b). Bai et al. pioneered the modeling of agents and their interactions through hypergraph structures, employing hypergraph convolutional neural networks for spatial transformation to address the "lazy agent" problem (Bai et al., 2022). Using hypergraph representations, Zhu et al. developed a communication modeling framework that enhances accuracy and specificity in multi-agent information exchange (Zhu et al., 2024). Zhang et al. propose IESCG, which leverages influence networks to construct sparse time-varying coordination graphs and incorporates recurrent payoff networks with sparse advantage coefficients to enhance coordination stability in MARL (Zhang et al., 2025a). Wei et al. proposed a graph attention neural network-based value decomposition method, theoretically proving its compliance with decomposability conditions during centralized training processes (Wei et al., 2024). Furthermore, Zhang et al. propose a hypergraph-based MARL method that adaptively builds agent interaction hypergraphs and uses hypergraph convolution for efficient cooperative policy learning (Zhang et al., 2022).

2.3. Game-theoretic approaches

Cooperative game theory represents a pivotal research direction in MARL, offering formal mechanisms for analyzing strategic interactions among rational agents (He et al., 2024). Nie et al. developed the Stackelberg-Nash Multi-Agent Proximal Policy Optimization algorithm, which addresses temporal coordination challenges in conventional demand response frameworks by establishing hierarchical decision-making protocols (Nie et al., 2024). Peters et al. proposed a contingency game theory approach that enables dynamic adjustment of critical temporal parameters, allowing agents to formulate strategies based on multiple potential intentions while adapting to real-time environmental fluctuations (Peters et al., 2024). Building upon this foundation, Wang et al. introduced a Shapley Q-value-based local reward allocation mechanism that more accurately reflects individual agents' actual contributions compared to traditional global reward sharing schemes (Wang et al., 2020). Furthermore, Wang et al. established an interpretable value decomposition framework grounded in cooperative game theory, successfully extending Shapley value principles to Markov convex games through rigorous theoretical derivations (Wang et al., 2022b).

3. Preliminaries

In this section, we provide an introduction to important background knowledge and some necessary definitions. The main notation used throughout the paper is summarized in Table 1.

3.1. Problem formulation

Cooperative MARL tasks are typically formalized as decentralized partially observable Markov decision processes

Table 1
Summary of main notations.

Notation	Description
C	coalition
$\mathbb{E}$	expected value
$\mathcal{E}$	hyperedge set
e	hyperedge
$\mathcal{H}$	hypergraph
$\mathcal{N}$	the set of all agents
o_i	the partial observation of agent i
P	the state transition function
P	the hypergraph-weight Position value
r	the reward function
$\mathbb{R}$	the real number field
$\mathcal{S}$	the state space
$\mathcal{U}$	the action space
u_i	the action selected by agent i
$\boldsymbol{u}$	the joint actions
$v(\cdot)$	the utility (characteristic) function of coalition
γ	discount factor
θ	the parameters of the networks
π	the strategy
$\boldsymbol{\pi}$	the joint strategy
τ	action-observation history
$\boldsymbol{\tau}$	the joint action-observation history

(Dec-POMDPs) (Oliehoek et al., 2008), defined by the tuple $\langle \mathcal{N}, \mathcal{S}, \mathcal{U}, P, r, \mathcal{O}, \gamma \rangle$. Here, $\mathcal{N} = \{1, \ldots, n\}$ denotes the set of agents, $s \in \mathcal{S}$ is the finite global state, $\mathcal{U}$ denotes the joint action space, namely the set of all possible combinations of the agents' individual actions. $P : \mathcal{S} \times \mathcal{U} \to \Delta(\mathcal{S})$ is the state transition function, $r : \mathcal{S} \times \mathcal{U} \to \mathbb{R}$ is the reward function, $\mathcal{O}$ is the observation space, and $\gamma \in (0, 1]$ is the discount factor. At each timestep, agent $i \in \mathcal{N}$ selects an action u_i according to its individual action-observation history τ_i through policy $\pi_i(u_i|\tau_i)$. The joint action $\boldsymbol{u} = \{u_1, \ldots, u_n\}$ represents one specific combination of the agents' actions, i.e., an element of the joint action space $\mathcal{U}$, and it induces a state transition $P(s'|s, \boldsymbol{u})$, updating the environment state from s to s'. Each agent then receives a new observation o_i' and updates its history as $\tau_i' = \tau_i \cup \{o_i', u_i\}$ for future decisions. The objective is to learn an optimal joint policy $\boldsymbol{\pi}$ that maximizes the joint state-action value function:

$$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u}) = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{t=0}^{\infty} \gamma^t r_{t+1} \,\middle|\, \boldsymbol{\tau}, \boldsymbol{u} \right], \tag{1}$$

where $\boldsymbol{\tau}$ denotes the joint action-observation history and the expectation is taken under policy $\boldsymbol{\pi}$.

3.2. Value decomposition

The objective of value decomposition methods under the CTDE paradigm is to learn a joint action-value function $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u})$. To enable decentralized execution, Q_{tot} must be decomposed into individual action-value functions $[Q_i(\tau_i, u_i)]_{i=1}^n$, as shown in Eq. (2).

$$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u}) = \mathcal{F}(Q_1(\tau_1, u_1), \ldots, Q_n(\tau_n, u_n)), \tag{2}$$

where $\mathcal{F}(\cdot)$ denotes the value decomposition function. To ensure consistency between joint and individual policies, the IGM principle must be satisfied (Son et al., 2019), which serves as a necessary condition for valid value decomposition. Formally, for a joint action-value function $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u})$, if there exist n independent utility functions $[Q_i(\tau_i, u_i)]_{i=1}^n$ satisfying:

$$\arg\max_{\boldsymbol{u}} Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u}) = (\arg\max_{u_1} Q_1(\tau_1, u_1), \ldots, \arg\max_{u_n} Q_n(\tau_n, u_n)), \tag{3}$$

then $[Q_i(\tau_i, u_i)]_{i=1}^n$ satisfies the IGM principle with respect to Q_{tot} under $\boldsymbol{\tau}$. In this case, we also say that $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u})$ is factorized by $[Q_i(\tau_i, u_i)]_{i=1}^n$.

3.3. Definitions

In this section, we provide the relevant definitions to this research.

Definition 1. [Allocation Efficiency] Let $v : 2^{\mathcal{N}} \to \mathbb{R}$ be the characteristic function of a cooperative game. A payoff vector $\boldsymbol{\phi} = (\phi_i)_{i \in C}$ satisfies allocation validity if it fulfills Eq. (4).

$$\sum_{i \in \mathcal{N}} \phi_i = v(\mathcal{N}), \tag{4}$$

where $\mathcal{N}$ represents the grand coalition (all agents), and $v(\cdot)$ denotes the coalition utility (characteristic) function in a standard cooperative game, which maps any coalition to its total achievable utility.

Definition 2. [Redundant Hyperedge] Let $\mathcal{H} = (\mathcal{N}, \mathcal{E})$ be a hypergraph and $C = (\mathcal{N}_C, \mathcal{E}_C) \subseteq \mathcal{H}$ be any sub-hypergraph. A hyperedge $e \in \mathcal{E}_C$ is called redundant in C if there exists e such that Eq. (5) holds:

$$f(C) = f(C \setminus e), \tag{5}$$

where $f : 2^{\mathcal{H}} \to \mathbb{R}$ is a function measuring hypergraph properties, and $C \setminus e := (\mathcal{N}_C, \mathcal{E}_C \setminus \{e\})$ denotes the sub-hypergraph with e removed.

Definition 3. [Hyperedge Redundancy Payoff Invariance] Let a hypergraph cooperative game be defined by a characteristic function $v : 2^{\mathcal{H}} \to \mathbb{R}$ with $\mathcal{H} = (\mathcal{N}, \mathcal{E})$. An allocation rule $\phi : \mathcal{H} \to \mathbb{R}^{|\mathcal{N}|}$ satisfies hyperedge redundancy payoff invariance if for any redundant hyperedge $e \in \mathcal{E}$ and all agents $i \in \mathcal{N}$, Eq. (6) holds:

$$\phi_i(\mathcal{H}) = \phi_i(\mathcal{H} \setminus e). \tag{6}$$

4. Methodology

In this section, we introduce the HIVE model. First, we present the hypergraph-weighted Position value game theory, and then use this game theory to decompose the joint state-action value Q_{tot} into the individual state-action value Q_i, the global collaborative baseline $\tilde{\Phi}_{\Gamma}^{\mathcal{H}}$, and the dynamic cooperative residual $\tilde{\Phi}_{\Gamma^e}^{C_e}$. These components are subsequently instantiated through three neural modules, namely the agent network, the global baseline integrator (GBI), and the dynamic coordination integrator (DCI), which correspond to the individual values, the global baseline, and the cooperative residual, together forming an implementable architecture.

4.1. Hypergraph-weighted position value

To explicitly model high-order agent interactions and achieve precise value decomposition, we propose the hypergraph-weighted Position value $\mathrm{P}^{\mathcal{H}}(\boldsymbol{\tau}, \boldsymbol{u})$, as defined in Eq. (7). In contrast to the classical Position Value, which equally distributes coalition payoffs among participants on each edge, hypergraph-weighted Position value incorporates a degree-based weighting mechanism to capture the heterogeneous roles of agents in multi-lateral collaborations. Here, the weight $d_{i,e}$ quantifies the relative involvement of agent i in hyperedge e, reflecting the intuition that agents participating in more hyperedges have greater structural capacity for collaboration. Accordingly, the marginal utility Φ_e of a hyperedge is allocated proportionally to $\frac{d_{i,e}}{\sum_{j \in \mathcal{N}} d_{j,e}}$.

$$\mathrm{P}_i^{\mathcal{H}}(\boldsymbol{\tau}, \boldsymbol{u}) = \sum_{e \in \mathcal{E}} \mathbb{E}_{C_e \sim Pr(C_e | \mathcal{E} \setminus \{e\})} \left[\frac{d_{i,e}}{\sum_{j \in \mathcal{N}} d_{j,e}} \Phi_e(\boldsymbol{\tau}, \boldsymbol{u}_{c_e \cup \{e\}}) \right], \tag{7}$$

where $\mathcal{H} = (\mathcal{N}, \mathcal{E})$ denotes the hypergraph consisting of all agents, $\mathcal{N}$ is the agent (node) set, $\mathcal{E}$ is the hyperedge set, C denotes a coalition whose participants are hyperedges, and $C_e \subseteq \mathcal{H}$ indicates a coalition excluding hyperedge e, $d_{i,e}$ is the weight of hyperedge e with respect to agent i, and $\Phi_e(\boldsymbol{\tau}, \boldsymbol{u}_{c_e \cup \{e\}})$ denotes the marginal utility of hyperedge e to coalition C_e.

Theorem 1. *For any hypergraph structure $\mathcal{H} = (\mathcal{N}, \mathcal{E})$, there exists a marginal utility function Φ such that the hypergraph-weighted Position value $\mathrm{P}^{\mathcal{H}}(\boldsymbol{\tau}, \boldsymbol{u})$ satisfies the allocation validity criterion. Moreover, $\mathrm{P}^{\mathcal{H}}(\boldsymbol{\tau}, \boldsymbol{u})$ preserves the payoff invariance property under redundant hyperedges.*

According to Definitions 1 and 3, Theorem 1 ensures that the joint state-action value function can be decomposed in additive form as:

$$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u}) = \sum_{i \in \mathcal{N}} \mathrm{P}_i^{\mathcal{H}}(\boldsymbol{\tau}, \boldsymbol{u}), \tag{8}$$

while guaranteeing that the decomposition remains correct even in the presence of redundant hyperedges.

4.2. Value decomposition

For a given joint action-observation history $\boldsymbol{\tau}$, Theorem 1 formally establishes the additive decomposition of the joint state-action value function Q_{tot} via the hypergraph-weighted Position value, as shown in Eq. (8). Since the hypergraph-weighted Position value remains implicit and cannot be directly computed in practice, we further decompose it. Theorem 2 presents this decomposition, which expresses Q_{tot} in terms of individual state-action values $[Q_i]_{i=1}^n$, a global coordination baseline, and a dynamic cooperation residual, as shown in Eq. (9).

Theorem 2. *A joint state-action value function*

$$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u}) = \sum_{i \in \mathcal{N}} Q_i(\tau_i, u_i) + \tilde{\Phi}_{\Gamma}^{\mathcal{H}}(\boldsymbol{\tau}) + \sum_{i \in \mathcal{N}} \sum_{e \in \mathcal{E}} \mathbb{E}_{C_e \sim Pr(C_e | \mathcal{E} \setminus \{e\})} \left[\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \boldsymbol{u}_{C_e'}) \right], \tag{9}$$

is factorized by $[Q_i(\tau_i, u_i)]_{i=1}^n$, *provided that* $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \boldsymbol{u}_{C_e'}) \le 0$ *and* $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \bar{\boldsymbol{u}}_{C_e'}) = 0$, *where* $C_e' = C_e \cup \{e\} \cup \{i\}$ *denotes the corresponding extended coalition, and* $\bar{\boldsymbol{u}}_{C_e'} \triangleq \left\{ \arg\max_{u_i} Q_i(\tau_i, u_i) \right\}_{i \in C_e'}$ *is the joint greedy action induced by the individual action-value functions.*

Theorem 2 indicates that if the residual terms satisfy $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \boldsymbol{u}_{C_e'}) \le 0$ and $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \bar{\boldsymbol{u}}_{C_e'}) = 0$, then any Q_{tot} of the form in Eq. (9) admits a greedy factorization by the individual utilities $[Q_i(\tau_i, u_i)]_{i=1}^n$, and hence satisfies the IGM principle. Intuitively, these constraints act as a canonical normalization for enforcing IGM: the residual never increases the value of any non-greedy joint action and vanishes at the joint greedy action induced by $\{\arg\max_{u_i} Q_i(\tau_i, u_i)\}$. Importantly, since this constraint is imposed only on the residual as an argmax-consistency correction, it does not reduce the representational capacity of the overall model.

Theorem 3. *For any joint state-action value function* $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u})$, *there exist individual state-action functions* $[Q_i(\tau_i, u_i)]_{i=1}^n$ *satisfying the IGM principle such that Eq. (9) holds, with* $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \boldsymbol{u}_{C_e'}) \le 0$ *and* $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \bar{\boldsymbol{u}}_{C_e'}) = 0$.

Theorem 3 establishes the necessary condition: for any Q_{tot}, if the IGM principle holds, then there always exists a decomposition of the form in Eq. (9) that preserves full equivalence between its optimal policy and that induced by the individual functions $[Q_i]_{i=1}^n$. The detailed proof is provided in Section 4.6.3, where we explicitly demonstrate the decomposition procedure of Eq. (9).

Theorems 2 and 3 establish the sufficiency and necessity of the proposed decomposition form in Eq. (9) with respect to the IGM principle, characterizing a complete equivalence between this decomposition family and the IGM principle. Crucially, while this equivalence is established via the specific hypergraph-based structure, Theorem 3 guarantees that this framework is functionally universal for the IGM class. Since any joint action-value function satisfying the IGM principle necessarily admits such a decomposition, the structural constraints in Eq. (9) are not arbitrary or restrictive assumptions, but rather the mathematical equivalent conditions required to represent the IGM principle in a linear decomposition form.

4.3. Neural networks

Fig. 2 illustrates the overall architecture of the HIVE model. The individual components $[Q_i(\tau_i, u_i)]_{i=1}^n$ are implemented as recurrent neural networks (RNNs) with GRUs (Chung et al., 2014), consistent with QMIX (Rashid et al., 2020b). The global coordination baseline $\tilde{\Phi}_{\Gamma}^{\mathcal{H}}$ and the dynamic cooperation residual $\tilde{\Phi}_{\Gamma^\circ}^{C_e}$ are approximated by two dedicated neural modules: the Global Baseline Integrator (GBI) and the Dynamic Coordination Integrator (DCI).

Since both GBI and DCI depend on the underlying interaction topology, we propose an Attention-driven Hypergraph Constructor (AHC) to capture high-order dependencies among agents effectively. Unlike conventional graph-based approaches that only consider pairwise relations, AHC leverages the hypergraph structure $\mathcal{H} = (\mathcal{N}, \mathcal{E})$ to explicitly encode multi-lateral dependencies. Specifically, it takes as input each agent's partial observation and historical actions, embeds them into latent feature representations, and applies an attention mechanism to adaptively measure inter-agent relevance in the learned latent space, which captures collaborative relevance or coupling strength rather than physical similarity in the raw observation space. The detailed procedure of AHC is summarized in Algorithm 1. It is worth noting that γ and m in Algorithm 1 play orthogonal roles. The threshold γ controls the selectivity of hyperedge acceptance and thus the hypergraph sparsity, whereas m sets the sampling budget for proposing hyperedges, mainly determining interaction coverage and computational complexity.

GBI. We design a dual-stream feature extraction architecture. One stream models high-order relationships in the collaborative topology $\mathcal{H}$ through hypergraph convolutional layers (HGCN), while the other analyzes latent patterns in the joint action-observation history $\boldsymbol{\tau}^t$ using multi-layer perceptrons (MLP). The dependence on $\boldsymbol{\tau}$ arises naturally in the proof of Theorem 3 (Section 4.6.3), where a baseline term depending solely on $\boldsymbol{\tau}$ emerges, thus motivating this design choice. The heterogeneous feature spaces are first merged through nonlinear cross-integration and then mapped by an MLP to the global collaborative baseline space, as formalized in Eq. (10):

$$\tilde{\Phi}_{\Gamma}^{\mathcal{H}} = \mathcal{F}_{\eta}(\mathcal{F}_{\psi}(\mathcal{H}) \oplus \mathcal{F}_{\eta}(\boldsymbol{\tau})), \tag{10}$$

where $\mathcal{F}_{\eta}(\cdot)$ denotes MLP, $\mathcal{F}_{\psi}(\cdot)$ represents HGCN, and $\oplus$ indicates feature merging operations.

Algorithm 1 Attention-driven hypergraph constructor (AHC).

Require: Partial observations $\{o_i^t\}_{i=1}^n$, historical actions $\{u_i^{t-1}\}_{i=1}^n$, similarity threshold γ, number of hyperedges m
Ensure: Hypergraph $\mathcal{H} = (\mathcal{N}, \mathcal{E})$
$x_i \leftarrow [o_i^t, u_i^{t-1}]$; ▷Concatenate each agent's input
$h_i \leftarrow f_{\text{enc}}(x_i)$; ▷ Encode inputs into latent representations
for each agent $i \in \mathcal{N}$ **do**
  $\alpha_{i,j} = \frac{\exp(\text{score}(h_i, h_j))}{\sum_{k=1}^n \exp(\text{score}(h_i, h_k))}$;
  $\tilde{h}_i \leftarrow \sum_{j=1}^n \alpha_{i,j} h_j$;
end for
Initialize hyperedge set: $\mathcal{E} \leftarrow \emptyset$;
for $r = 1$ to m **do**
  $i_r \sim \text{Uniform}(\mathcal{N})$
  for each $j \in \mathcal{N}, j \neq i_r$ **do**
    $s_{i_r,j} \leftarrow \text{sim}(\tilde{h}_{i_r}, \tilde{h}_j)$ ▷ compute similarity
  end for
  $\mathcal{N}_{i_r} \leftarrow \{j \mid s_{i_r,j} > \gamma\}$ ▷ select neighbors
  $e_r \leftarrow \{i_r\} \cup \mathcal{N}_{i_r}$
  $\mathcal{E} \leftarrow \mathcal{E} \cup \{e_r\}$
end for
$\mathcal{H} = (\mathcal{N}, \mathcal{E})$;
return $\mathcal{H}$

DCI. We first employ a Monte Carlo Sampler (MCS) to sample hyperedges and construct hyperedge coalitions, as defined in Eq. (11):

$$C_e \sim \mathcal{P}, \quad P(C_e) = \frac{|C_e|!(|\mathcal{E}| - |C_e| - 1)!}{|\mathcal{E}|!}, \tag{11}$$

where C_e denotes a hyperedge coalition, $\mathcal{P}$ is the coalition distribution, and $P(C_e)$ is its probability.

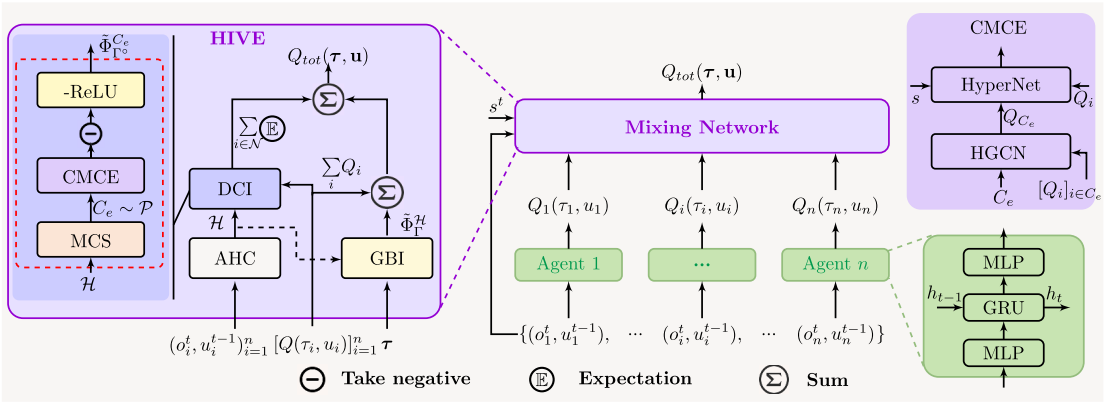


Fig. 2. Overall architecture of the HIVE model. Q_{tot} is decomposed into Q_i, $\tilde{\Phi}_{\Gamma}^{\mathcal{H}}$, and $\tilde{\Phi}_{\Gamma^\circ}^{C_e}$, approximated by the Agent, GBI, and DCI modules to ensure IGM consistency. Here, AHC denotes the Attention-driven Hypergraph Constructor, MCS the Monte Carlo Sampler, CMCE the Corrected Marginal Contribution Estimator, HGCN the Hypergraph Convolutional Network, and HyperNet the Hypernetwork.

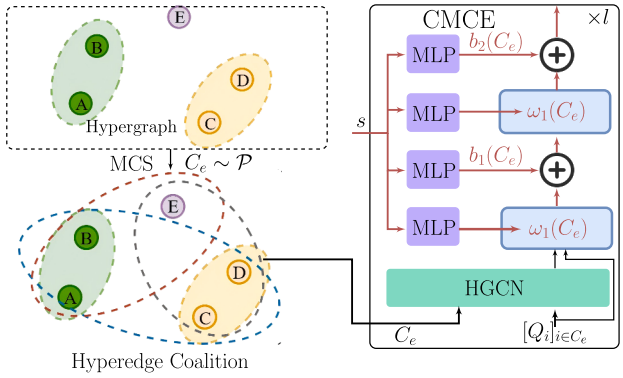


Fig. 3. Structure diagram of CMCE. MCS samples hyperedge coalitions, which are encoded by HGCN and processed by MLP-based hypernetworks, and the outputs are further shaped with a negative ReLU function to form the dynamic cooperative residual.

Following this sampling process, we estimate the marginal corrected utility for each hyperedge using the Corrected Marginal Contribution Estimator (CMCE) (Fig. 3). CMCE consists of two components: (i) a hypergraph convolutional network (HGCN) that encodes high-order dependencies among agents, and (ii) a hypernetwork structure, analogous to that used in QMIX (Rashid et al., 2020b), enabling effective approximation of the dynamic cooperative residual $\tilde{\Phi}_{\Gamma^\circ}^{C_e}$.

To satisfy the constraints in Theorem 2, the residual is explicitly constructed to enforce both conditions. As shown in Eq. (12), when coalition C_e' selects the joint greedy action $\bar{\boldsymbol{u}}_{C_e'}$ induced by the individual action-value functions, the residual is set to zero, i.e., $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \bar{\boldsymbol{u}}_{C_e'}) = 0$. For any non-greedy joint action $\boldsymbol{u}_{C_e'} \neq \bar{\boldsymbol{u}}_{C_e'}$, a negative ReLU activation guarantees $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}, \boldsymbol{u}_{C_e'}) \leq 0$:

$$\tilde{\Phi}_{\Gamma^\circ}^{C_e} = \begin{cases} 0, & \boldsymbol{u}_{C_e'} = \bar{\boldsymbol{u}}_{C_e'} \\ -\mathrm{ReLU}(-\mathcal{F}_s(Q_{C_e}(\boldsymbol{\tau}_{C_e}, \boldsymbol{u}_{C_e}), Q_i(\tau_i, u_i))), & \boldsymbol{u}_{C_e'} \neq \bar{\boldsymbol{u}}_{C_e'} \end{cases} \tag{12}$$

where $\mathcal{F}_s(\cdot)$ denotes the hypernetwork, and $Q_{C_e}(\boldsymbol{\tau}_{C_e}, \boldsymbol{u}_{C_e})$ is computed by the HGCN $\mathcal{F}_\psi(\cdot)$ as given in Eq. (13):

$$Q_{C_e}(\boldsymbol{\tau}_{C_e}, \boldsymbol{u}_{C_e}) = \mathcal{F}_\psi(C_e, [Q_i]_{i \in C_e}). \tag{13}$$

Algorithm 2 Dynamic coordination integrator (DCI).

Require: Hypergraph $\mathcal{H} = (\mathcal{N}, \mathcal{E})$, individual action-value functions $\{Q_i(\tau_i, u_i)\}_{i \in \mathcal{N}}$, number of Monte Carlo samples K, coalition distribution $\mathcal{P}$

Ensure: Dynamic cooperative residuals $\{\tilde{\Phi}_{\Gamma^\circ}^{C_e}\}_{C_e \subseteq \mathcal{E}}$

1: **for** $k = 1$ to K **do**
2: Sample hyperedge coalition $C_e^k \sim \mathcal{P}$ by Eq. (11);
3: $Q_{C_e^k}(\boldsymbol{\tau}_{C_e^k}, \boldsymbol{u}_{C_e^k}) \leftarrow$ Eq. (13);
4: **end for**
5: $\tilde{\Phi}_{\Gamma^\circ}^{C_e} \leftarrow$ Eq. (12);
6: **return** $\tilde{\Phi}_{\Gamma^\circ}^{C_e}$

4.4. Model training

The complete training process of HIVE is concisely outlined in Algorithm 3, where the algorithm takes partial observations as input and learns robust collective policies.

4.5. Complexity analysis

4.5.1. Space complexity

The space complexity of HIVE's training process is mainly determined by the learning parameters of the hypernetwork (mixer), the self-gating layer in the marginal utility computation, and the intermediate tensors generated during DCI calculation. Let B denote the batch size, T the episode length, N the number of agents, E the number of hyperedges, S the sampling number of hyperedge coalitions, D_s the global state dimension, D_o the observation dimension, and D_h the hidden dimension of the hypernetwork. The basic feature storage consists of the observation matrix $\mathbf{H} \in \mathbb{R}^{N \times D_o}$ and the hypergraph adjacency matrix $\mathbf{G} \in \mathbb{R}^{N \times E}$. In terms of parameters, the largest component is the hypernetwork with size $\mathbb{R}^{D_s \times NED_h}$, while the gating layers $\mathbb{R}^{D_o \times D_o}$ and other linear layers contribute relatively little. For intermediate tensors, the dominant memory cost arises from DCI, whose scale is approximately $BTSNED_h$. Therefore, the overall space complexity of HIVE can be approximated as $O(BTSNED_h + D_sNED_h)$. Since $BTS \gg D_s$ in practice, the complexity reduces to $O(BTSNED_h)$ and the memory requirements remain moderate under typical experimental settings.

4.5.2. Time complexity

The time complexity of HIVE's training process mainly arises from three components: hypergraph construction, marginal utility

Algorithm 3 Training of HIVE.

Require: $n = |\mathcal{N}|$, number of hyperedge E, Hyper-parameters $\varepsilon, episode$, Adam hyper-parameters $\beta_1, \beta_2, \beta_3$.
Ensure: Parameters θ of the trained HIVE model.

```
Initialize the reply buffer B;
Initialize the parameter θ of HIVE;
Initialize target parameters θ⁻ ← θ;
for epi = 1 to episode do
    Observe initial state s⁰ and observation [o⁰]_{i=1}^n for each agent i;
    for t = 1 to T do
        Get [Q_i^t(τ_i^t, u_i^t)]_{i=1}^n by Agent network;
        With ε_greedy policy select action:
            u_i^t = { random action,        with probability ε
                    { arg max_{u_i} Q_i^t(τ_i^t, u_i),   otherwise
        Take action u^t = [u_i^t]_{i=1}^n, and retrieve next observation and reward r^t;
        Store transition {τ^t, u^t, r^t, τ^{t+1}} in B;
    end for
    Sample a minibatch of data with batch size B from B;
    while each sampled batch do
        H ← Algorithm 1;
        Get [Q_i(τ_i, u_i)]_{i=1}^n by Agent network for each agent;
        Φ̃_Γ^H ← GBI;
        Φ̃_{Γ°}^H ← Algorithm 2;
        Q_tot ← Eq. (9);
        y^tot ← r + γ max_{u'} Q_tot(τ', u'; θ⁻);
        L_θ ← (1/B) Σ_{b=1}^B [y_b^tot − Q_tot(τ, u; θ)]²;
        θ ← Adam(L_θ, β1, β2, β3);
        Periodically update target network parameters θ⁻ ← θ;
    end while
end for
Output θ                                   ▷ HIVE model parameters
```

aggregation, and the DCI module. For hypergraph construction, local observations are first encoded by a multi-head attention mechanism with complexity $O(BND_o^2)$, after which pairwise similarity is computed between agents, neighbors are selected according to a threshold γ, and hyperedges are dynamically updated. This procedure introduces an additional complexity of $O(BN^2D_o)$. For marginal utility aggregation, local Q-values are propagated through the hypergraph adjacency matrix, requiring $O(BSE^2N)$ operations. In the DCI module, the global state ($BTSD_s$) is mapped by a hypernetwork to constraint terms ($BTSNED_h$), resulting in $O(BTSD_sNED_h)$ complexity, while the state-to-weight mapping and subsequent aggregation with local Q-values incur additional costs of order $O(BTSD_sD_h)$ and $O(BTSNED_h)$, respectively. Therefore, the overall time complexity of HIVE can be approximated as $O(BTSD_sNED_h)$. Since B, T, and S are usually moderate in practical settings, and N task-specific constraints bound and E, the computational overhead remains manageable despite the quadratic term in hypergraph construction.

4.6. Theoretical Proof

In this section, we provide the proofs of theorems.

4.6.1. Proof of Theorem 1

First, we prove the existence theorem of allocative efficiency.

Proof. To establish validity, we prove the existence of a marginal utility function Φ_e satisfying Eq. (14).

$$Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{i\in\mathcal{N}} \mathrm{P}_i^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})], \tag{14}$$

where $C_e \subseteq \mathcal{H}$ denote the coalition of hyperedges that excludes hyperedge e, where $\mathcal{H}$ represents the hypergraph encompassing all agents (the grand coalition). According to the definition of the Shapley value (Li et al., 2024b), the derivation for any given $\boldsymbol{\tau}$ and $\boldsymbol{u}$ is as follows:

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})],\\
&= \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[v(C_e\cup\{e\}) - v(C_e)],\\
&= \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} Sh_e(v),\\
&= \sum_{e\in\mathcal{E}}\sum_{i\in\mathcal{N}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} Sh_e(v),\\
&= \sum_{e\in\mathcal{E}} Sh_e(v),
\end{aligned} \tag{15}$$

where Sh_e is the Shapley value of hyperedges. For any $\boldsymbol{\tau}$ and $\boldsymbol{u}$, we can find a characteristic function v satisfying:

$$v(\emptyset) = 0, v(\mathcal{H}) = Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}), \tag{16}$$

where $\emptyset$ denotes the empty coalition. Substituting Eq. (16) into Eq. (15) yields the following expression:

$$\sum_{e\in\mathcal{E}} Sh_e(v) = v(\mathcal{H}) - v(\emptyset) = Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}). \tag{17}$$

Thus, there exists a marginal utility function such that the hypergraph-weighted Position value exhibits allocation validity. In fact, Eq. (17) is exactly the condition required in Definition 1, hence the claim follows. □

Then, we prove the hyperedge redundancy payoff invariance.

Proof. Let e' be a redundant hyperedge with respect to the characteristic function v. By Definition 2, for any coalition $C \subseteq \mathcal{E}$, we have $v(C) = v(C \setminus e')$. Consequently, the Shapley value of e' satisfies $Sh_{e'}(v) = 0$. The Position value under the reduced hypergraph $\mathcal{H} \setminus e'$ can be derived as follows:

$$\mathrm{P}_i^{\mathcal{H}\setminus e'}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{e\in\mathcal{E}\setminus e'} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \Psi_e, \tag{18}$$

where we introduce the auxiliary terms

$$\alpha(C) = \frac{|C|!(|\mathcal{E}| - |C| - 1)!}{|\mathcal{E}|!}, \tag{19}$$

$$\Delta_e(C) = v(C\cup\{e\}) - v(C), \tag{20}$$

$$\Delta_{e,e'}(C) = v(C\cup\{e,e'\}) - v(C\setminus\{e'\}). \tag{21}$$

Then the inner summation can be simplified step by step:

$$\begin{aligned}
\Psi_e &= \sum_{C\subseteq\mathcal{H}\setminus\{e,e'\}} \frac{|C|!(|\mathcal{E}| - 1 - |C| - 1)!}{(|\mathcal{E}| - 1)!} \Delta_{e,e'}(C)\\
&= \sum_{C\subseteq\mathcal{H}\setminus\{e,e'\}} \frac{|C|!(|\mathcal{E}| - |C| - 2)!}{(|\mathcal{E}| - 1)!} \Delta_e(C)\\
&= \sum_{C\subseteq\mathcal{H}\setminus\{e,e'\}} \frac{|\mathcal{E}|}{|\mathcal{E}| - |C| - 1} \alpha(C)\,\Delta_e(C)\\
&= \sum_{C\subseteq\mathcal{H}\setminus\{e,e'\}} \left(1 + \frac{|C|+1}{|\mathcal{E}|-|C|-1}\right)\alpha(C)\,\Delta_e(C)\\
&= \sum_{C\subseteq\mathcal{H}\setminus\{e,e'\}} \alpha(C)\,\Delta_e(C) + \sum_{C\subseteq\mathcal{H}\setminus\{e,e'\}} \alpha(C\cup\{e'\})\,\Delta_e(C).
\end{aligned} \tag{22}$$

Finally, we obtain

$$\Psi_e = \sum_{C\subseteq\mathcal{H}\setminus\{e\}} \alpha(C)\,\Delta_e(C), \tag{23}$$

$$\mathrm{P}_i^{\mathcal{H}\setminus e'}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{e\in\mathcal{E}\setminus e'} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} Sh_e(v). \tag{24}$$

Given $Sh_{e'}(v)=0$, we obtain:

$$\begin{aligned}\mathrm{P}_i^{\mathcal{H}\setminus e'}(\boldsymbol{\tau},\boldsymbol{u}) &= \sum_{e\in\mathcal{E}\setminus e'}\frac{d_{i,e}}{\sum_{i\in\mathcal{N}} d_{i,e}}Sh_e(v)\\ &= \sum_{e\in\mathcal{E}}\frac{d_{i,e}}{\sum_{i\in\mathcal{N}} d_{i,e}}Sh_e(v)\\ &= \mathrm{P}_i^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}).\end{aligned} \tag{25}$$

Hence, by Definition 3, the hypergraph-weighted Position value exhibits redundant hyperedge payoff invariance. □

4.6.2. Proof of Theorem 2

Proof. Since $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau}_{C_e},\boldsymbol{u}_{C_e})\le 0$, Eq. (26) holds.

$$\begin{aligned}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) &\le \sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i)+\tilde{\Phi}_{\Gamma}^{\mathcal{H}}(\boldsymbol{\tau}),\\ &\le \sum_{i\in\mathcal{N}} Q_i(\tau_i,\bar{u}_i)+\tilde{\Phi}_{\Gamma}^{\mathcal{H}}(\boldsymbol{\tau}).\end{aligned} \tag{26}$$

Additionally, since $\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau},\bar{\boldsymbol{u}}_{C_e'})=0$, Eq. (27) holds.

$$\begin{aligned}\sum_{i\in\mathcal{N}} Q_i(\tau_i,\bar{u}_i)+\tilde{\Phi}_{\Gamma}^{\mathcal{H}}(\boldsymbol{\tau}) &= \sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i)+\tilde{\Phi}_{\Gamma}^{\mathcal{H}}(\boldsymbol{\tau})+\sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}}\mathbb{E}_{C_e\sim Pr}[\tilde{\Phi}_{\Gamma^\circ}^{C_e}(\boldsymbol{\tau},\bar{\boldsymbol{u}}_{C_e'})],\\ &= Q_{tot}(\boldsymbol{\tau},\bar{\boldsymbol{u}}).\end{aligned} \tag{27}$$

Combining Eqs. (26) and (27), the following expression is obtained:

$$Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})\le Q_{tot}(\boldsymbol{\tau},\bar{\boldsymbol{u}}), \tag{28}$$

Thus, it follows that $\arg\max_{\boldsymbol{u}} Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})=\bar{\boldsymbol{u}}$, where $\bar{\boldsymbol{u}}=[\arg\max_{u_i} Q_i(\tau_i,u_i)]_{i=1}^n$. The proof of the theorem is complete. □

4.6.3. Proof of Theorem 3

Before proving Theorem 3, we first present two lemmas.

Lemma 1. *Let the hypergraph $\mathcal{H}=(\mathcal{N},\mathcal{E})$ be given. For any coalition of hyperedges C_e that does not contain hyperedge e, a history trajectory $\boldsymbol{\tau}$, and a joint action $\boldsymbol{u}$, there exists a marginal utility function $\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})$ that satisfies the allocation validity of the hypergraph-weighted Position value and ensures that Eq. (29) holds:*

$$\boldsymbol{u}^*_{C_e\cup\{e\}}=\arg\max_{\boldsymbol{u}_{C_e\cup\{e\}}}\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}}), \tag{29}$$

where $\boldsymbol{u}^*_{C_e\cup\{e\}}$ denotes the action of the global optimal $\boldsymbol{u}^*$ after removing the actions of agents not in the coalition $C_e\cup\{e\}$, with $\boldsymbol{u}^*=\arg\max_{\boldsymbol{u}} Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})$ representing the global optimal action.

Proof. Given any $\boldsymbol{\tau}$ and $\boldsymbol{u}$, we construct the following characteristic function $v(C)$:

$$v(C)=\max_{\boldsymbol{u}'_{-C}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}_C,\boldsymbol{u}'_{-C})]-\max_{\boldsymbol{u}'}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}')+\frac{|C|}{|\mathcal{E}|}\max_{\boldsymbol{u}'}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}'), \tag{30}$$

where the notation $-C$ represents the set of agents not in the coalition C. Thus, the marginal utility function can be expressed as Eq. (31).

$$\begin{aligned}\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}}) &= v(C_e\cup\{e\})-v(C_e),\\ &= \max_{\boldsymbol{u}'_{-(C_e\cup\{e\})}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}},\boldsymbol{u}'_{-(C_e\cup\{e\})})]\\ &\quad-\max_{\boldsymbol{u}'_{-C_e}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}_{C_e},\boldsymbol{u}'_{-C_e})]\\ &\quad+\frac{1}{|\mathcal{E}|}\max_{\boldsymbol{u}'}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}').\end{aligned} \tag{31}$$

Furthermore, this characteristic function ensures that Eq. (32) holds, and therefore, the characteristic function $v(C_e)$ satisfies the allocation validity (Theorem 1).

$$\begin{aligned}v(\emptyset) &= \max_{\boldsymbol{u}'}Q(\boldsymbol{\tau},\boldsymbol{u}')-\max_{\boldsymbol{u}'}Q(\boldsymbol{\tau},\boldsymbol{u}')+0=0,\\ v(\mathcal{H}) &= Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})-\max_{\boldsymbol{u}'}Q(\boldsymbol{\tau},\boldsymbol{u}')+1\cdot\max_{\boldsymbol{u}'}Q(\boldsymbol{\tau},\boldsymbol{u}')\\ &= Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}).\end{aligned} \tag{32}$$

Note that $\max_{\boldsymbol{u}'_{-(C_e\cup\{e\})}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}},\boldsymbol{u}'_{-(C_e\cup\{e\})})]\le\max_{\boldsymbol{u}'_{-C_e}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}_{C_e},\boldsymbol{u}'_{-C_e})]$ always holds, and when $\boldsymbol{u}=\boldsymbol{u}^*$, Eq. (33) holds:

$$\max_{\boldsymbol{u}'_{-(C_e\cup\{e\})}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}^*_{C_e\cup\{e\}},\boldsymbol{u}'_{-(C_e\cup\{e\})})]=\max_{\boldsymbol{u}'_{-C_e}}[Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}^*_{C_e},\boldsymbol{u}'_{-C_e})]=Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}^*). \tag{33}$$

From this, it follows that when $\boldsymbol{u}_{C_e\cup\{e\}}=\boldsymbol{u}^*_{C_e\cup\{e\}}$, $\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})$ reaches its maximum value, which is $\frac{1}{|\mathcal{E}|}\max_{\boldsymbol{u}}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})$. Hence, The proof of the lemma is complete. □

Lemma 2. *Given a hypergraph $\mathcal{H}=(\mathcal{N},\mathcal{E})$ and a history trajectory $\boldsymbol{\tau}$, there exists a individual state-action value function Q_i. When $d_{i,e}\neq 0$, its local optimal action $\bar{u}_i$ satisfies:*

$$\bar{\boldsymbol{u}}=\arg\max_{\boldsymbol{u}}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}),\bar{\boldsymbol{u}}_{C_e'}=\arg\max_{\boldsymbol{u}_{C_e'}}\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}_{C_e'}), \tag{34}$$

where $\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}_{C_e'})=\frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})-\frac{\sum_{i\in\mathcal{N}}d_{i,e}}{|\mathcal{E}|\cdot d_{i,e}}Q_i(\tau_i,u_i)]$, Φ_e denotes the marginal utility function, and $C_e'=C_e\cup\{e\}\cup\{i\}$. $\bar{\boldsymbol{u}}_{C_e'}$ represents the action set of the agents in the hyperedge coalition C_e' when all agents choose the local optimal action $\bar{\boldsymbol{u}}$, with $\bar{\boldsymbol{u}}=[\arg\max_{u_i}Q_i(\tau_i,u_i)]_{i=1}^n$.

Proof. First, we define the minimum coalition reward difference L for the global optimal action $\boldsymbol{u}^*$:

$$L=\min_{e,C_e,\boldsymbol{u}'\neq\boldsymbol{u}^*}(\Phi_e(\boldsymbol{\tau},\boldsymbol{u}^*_{C_e\cup\{e\}})-\Phi_e(\boldsymbol{\tau},\boldsymbol{u}'_{C_e\cup\{e\}})), \tag{35}$$

where $\boldsymbol{u}^*_{C_e\cup\{e\}}$ denotes the action of coalition $C_e\cup\{e\}$ when the global optimal action is taken. By Lemma 1, the minimum coalition reward difference $L>0$. Next, we construct the individual state-action value function Q_i:

$$Q_i(\tau_i,u_i)=\begin{cases}\frac{L}{2}\cdot|\mathcal{E}|\cdot\frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}, & \text{if } u_i=u_i^*,\\ 0, & \text{otherwise,}\end{cases} \tag{36}$$

where $\boldsymbol{u}_i^*$ denotes the action of agent i when the global optimal action is taken. Clearly, this individual state-action value function satisfies $\bar{\boldsymbol{u}}=\arg\max_{\boldsymbol{u}}Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})$ and $Q_i\ge 0$.

When $\boldsymbol{u}=\boldsymbol{u}^*$, we have:

$$\begin{aligned}\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}_{C_e'}) &= \frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})-\frac{1}{|\mathcal{E}|\cdot\frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}}Q_i(\tau_i,u_i)],\\ &= \frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})-\frac{L}{2}],\\ &\ge \frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}\Phi_e(\boldsymbol{\tau},\boldsymbol{u}'_{C_e\cup\{e\}}),\end{aligned} \tag{37}$$

where $\boldsymbol{u}'\neq\boldsymbol{u}^*$. Since the individual state-action value function Q_i is non-negative, Eq. (38) holds when $\boldsymbol{u}\neq\boldsymbol{u}^*$.

$$\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}_{C_e'})\le\frac{d_{i,e}}{\sum_{j\in\mathcal{N}}d_{j,e}}\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})\le\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}^*_{C_e'}). \tag{38}$$

Therefore, $\boldsymbol{u}^*_{C_e'}$ is the optimal action for $\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}_{C_e'})$. When the individual state-action value function satisfies Eq. (36), it is clear that $\boldsymbol{u}_i^*=\bar{u}_i$ holds, where $\bar{u}_i=\arg\max_{u_i}Q_i(\tau_i,u_i)$. Thus, Eq. (39) holds:

$$\bar{\boldsymbol{u}}_{C_e'}=\boldsymbol{u}^*_{C_e'}=\arg\max_{\boldsymbol{u}_{C_e'}}\tilde{\Phi}_{i,e}^{\mathcal{H}}(\boldsymbol{\tau},\boldsymbol{u}_{C_e'}). \tag{39}$$

The proof of the lemma is complete. □

Proof of Theorem 3:

Proof. By Lemma 1, there exists a marginal utility function Φ_e, such that Eq. (40) holds.

$$Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})]. \tag{40}$$

Therefore, the joint state-action value function Q_{tot} can be decomposed as follows:

$$\begin{aligned}
&Q_{tot}(\boldsymbol{\tau},\boldsymbol{u})\\
&= \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}\left[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})\right],\\
&= \sum_{i\in\mathcal{N}}\sum_{\substack{e\in\mathcal{E}\\ d_{i,e}\neq 0}} \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}\left[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})\right],\\
&= \sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i)\\
&+ \sum_{i\in\mathcal{N}}\sum_{\substack{e\in\mathcal{E}\\ d_{i,e}\neq 0}} \left[\frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}\left[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}})\right]\right.\\
&\left. - \frac{1}{|\mathcal{E}|} Q_i(\tau_i,u_i)\right],\\
&= \sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i)\\
&+ \sum_{i\in\mathcal{N}}\sum_{\substack{e\in\mathcal{E}\\ d_{i,e}\neq 0}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}\left[\frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}}\right.\\
&\left.\times\left[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}}) - \frac{\sum_{j\in\mathcal{N}} d_{j,e}}{|\mathcal{E}|\cdot d_{i,e}} Q_i(\tau_i,u_i)\right]\right].
\end{aligned} \tag{41}$$

According to Lemma 2, there exists a set of individual state-action value functions Q_i such that the locally optimal actions satisfy $\arg\max_{\boldsymbol{u}} Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) = [\arg\max_{u_i} Q_i(\tau_i,u_i)]_{i=1}^n$. Therefore, when Q_i satisfies the conditions in Lemma 2, Eq. (41) follows the IGM principle. For ease of exposition, we define the following function:

$$\tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) = \frac{d_{i,e}}{\sum_{j\in\mathcal{N}} d_{j,e}}[\Phi_e(\boldsymbol{\tau},\boldsymbol{u}_{C_e\cup\{e\}}) - \frac{\sum_{i\in\mathcal{N}} d_{i,e}}{|\mathcal{E}|\cdot d_{i,e}} Q_i(\tau_i,u_i)]. \tag{42}$$

Thus, Q_{tot} can be rewritten in the form of Eq. (43):

$$Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i) + \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E},d_{i,e}\neq 0} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[\tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e})]. \tag{43}$$

Next, we define a new function $\tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau}) = \max \tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\cdot)$. Clearly, $\tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) - \tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau}) \leq 0$, and when $\boldsymbol{u} = \boldsymbol{u}^*$, by Lemma 2, we derive that $\tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) - \tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau}) = 0$. We then rewrite Q_{tot} as follows:

$$\begin{aligned}
Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) = &\sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i) + \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E},d_{i,e}\neq 0} \tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau})\\
&+ \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E},d_{i,e}\neq 0} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[\tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) - \tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau})].
\end{aligned} \tag{44}$$

Let

$$\tilde{\Phi}^{C_e}_{\Gamma^\circ}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) = \begin{cases} 0, & \text{if } d_{i,e} = 0,\\ \tilde{\Phi}^{\mathcal{H}}_{i,e}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) - \tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau}), & \text{otherwise.}\end{cases} \tag{45}$$

Then, $\tilde{\Phi}^{C_e}_{\Gamma^\circ}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) \leq 0$, and when $\boldsymbol{u} = \boldsymbol{u}^*$, $\tilde{\Phi}^{C_e}_{\Gamma^\circ}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e}) = 0$. Define $\tilde{\Phi}^{\mathcal{H}}_{\Gamma}(\boldsymbol{\tau}) = \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E},d_{i,e}\neq 0} \tilde{\Phi}^{\Gamma}_{i,e}(\boldsymbol{\tau})$. Thus, Q_{tot} can be written as:

$$Q_{tot}(\boldsymbol{\tau},\boldsymbol{u}) = \sum_{i\in\mathcal{N}} Q_i(\tau_i,u_i) + \tilde{\Phi}^{\mathcal{H}}_{\Gamma}(\boldsymbol{\tau}) + \sum_{i\in\mathcal{N}}\sum_{e\in\mathcal{E}} \mathbb{E}_{C_e\sim Pr(C_e|\mathcal{E}\setminus\{e\})}[\tilde{\Phi}^{C_e}_{\Gamma^\circ}(\boldsymbol{\tau},\boldsymbol{u}_{C'_e})]. \tag{46}$$

The proof of the theorem is complete. □

5. Experiments

To evaluate the performance of HIVE, we validate it across a spectrum of MARL benchmarks, from fundamental games to large-scale cooperative tasks, demonstrating consistent advantages over state-of-the-art baselines. We further conduct ablation studies on both model components and hyperedge configurations to quantify their respective contributions. The source code of our method is publicly available at https://github.com/ACoTAI/HIVE.

5.1. Experimental platforms

We evaluated our algorithm on two widely used benchmark platforms: matrix games (Hu et al., 2025), where we also consider extended settings with more agents and larger action spaces, and the StarCraft Multi-Agent Challenge (SMAC) (Samvelyan et al., 2019) along with its successor SMACv2 (Ellis et al., 2023), an enhanced version that introduces greater stochasticity and increased environmental complexity. Fig. 4 illustrates representative scenarios from the two benchmark platforms.

Matrix games. We first consider a two-agent matrix game, where each agent has three discrete actions. As shown in Fig. 4(a), the optimal joint action (A, A) yields a reward of 8, while the suboptimal joint action (C, C) provides a reward of 7. The narrow margin of 1 between the optimal and suboptimal strategies, combined with the fact that the neighborhood around the optimal action is surrounded by penalizing rewards of -12, creates a challenging configuration in which exploratory noise can easily drive algorithms into suboptimal local traps.

To further examine scalability, we extend the number of agents in the matrix game from the original two to six and ten, enabling evaluation in larger cooperative scenarios. In addition, for the two-agent setting, we enlarge the action space from 3 to 21 dimensions, with the joint reward function defined in Eq. (47). The reward distribution is illustrated in Fig. 4(c), where the joint action $(u_1, u_2) = (5, 15)$ corresponds to the optimal reward, while $(u_1, u_2) = (15, 5)$ yields a suboptimal reward.

$$R(u_1,u_2) = \max\left(5 - \frac{(15-u_1)^2}{9} - \frac{(5-u_2)^2}{9},\ 10 - (5-u_1)^2 - (15-u_2)^2\right). \tag{47}$$

SMAC. We also evaluate HIVE on SMAC (Samvelyan et al., 2019), a widely adopted benchmark for cooperative multi-agent reinforcement learning. Fig. 4(b) shows the challenging 2c_vs_64zg scenario, where two allied units must coordinate to defeat a large number of enemy units. In addition to the original SMAC, we further conduct experiments on SMACv2 (Ellis et al., 2023), which enhances stochasticity and partial observability by randomizing unit compositions and starting positions, and adopting more realistic attack and sight ranges. These modifications force agents to learn cooperative behaviors in previously unseen scenarios and increase the difficulty of decentralized decision-making.

5.2. Experimental settings

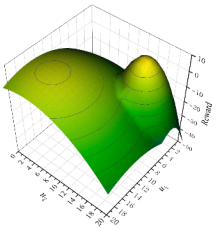
To ensure fair comparisons and reproducibility, all baseline methods are implemented using the official publicly available code released by the original authors, and their hyperparameters strictly follow the standard settings reported in the corresponding original papers. For our method, the number of hyperedges is set to 16 in most SMAC scenarios. For the matrix game tasks, we set the number of hyperedges to 4 when there are 2 agents; when the number of agents is 6 and 10, the numbers of hyperedges are set to 8 and 16, respectively. For the AHC module, the similarity threshold γ is set according to the agent-type composition of the scenario in SMAC/SMACv2: for scenarios without heterogeneous agents, γ is set to 0.6 (e.g., 5m_vs_6m and 6h_vs_8z), while for scenarios

Agent 1 \ Agent 2	A	B	C
A	8	-12	-12
B	-12	6	0
C	-12	0	7

(a) Payoff of matrix game



(b) 2c_vs_64zg scenario in SMAC



(c) Reward distribution of 21-action matrix game

Fig. 4. Experimental platforms. (a) Payoff matrices of the matrix game, (b) the 2c_vs_64zg battle scenario in SMAC, and (c) the reward distribution of the 21-dimensional matrix game.

Table 2
Table of experimental setups.

Hyperparameters	Value	Description
batch size	32	number of samples per pass
discount factor γ	0.99	future reward importance rate
optimizer	Adam	optimization algorithm for parameter updates
learning rate	0.0005	step size for updating model weights
max training step	2,000,000	number of training steps
target update interval	200	frequency of target network updates
exploration step	50,000	steps to decay exploration over time
epsilon start	1.0	starting value of exploration rate ϵ
epsilon finish	0.05	final value of exploration rate ϵ
test interval	10,000	interval for running evaluation during training
test episode	32	number of episodes per evaluation test
replay buffer size	5000	maximum episodes stored in memory

Table 3
Experimental hardware and system configuration.

Component	Specification
Workstation	Dell Precision 7810 Tower
Operating System	Ubuntu 20.04.6 LTS (Kernel 5.15.0)
CPU	Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20 GHz
CPU Cores / Threads	10 cores / 20 threads
Memory	256 GB RAM
GPU	NVIDIA GeForce RTX 3090 (24 GB VRAM)
Motherboard	Dell Inc. 0KJCC5

with heterogeneous agents, γ is set to 0.4 (e.g., MMM2 and 1c3s5z). In all matrix-game settings, γ is fixed at 0.5. The main hyperparameters are summarized in Table 2. In addition, the maximum number of training steps is set to 50,000 for matrix games (Hu et al., 2025) and 2,000,000 for SMAC (Samvelyan et al., 2019).

All experiments were executed on a unified hardware and system environment to ensure reproducibility and eliminate implementation-related bias. The detailed hardware configuration and system environment are summarized in Table 3.

5.3. Baseline

We compare the proposed method with the following baselines:

QTRAN (Son et al., 2019) overcomes the structural constraints of VDN (Sunehag et al., 2018) and QMIX (Rashid et al., 2020b), enabling more general factorization and superior performance in tasks with aggressive penalties for non-cooperative behavior.

QMIX (Rashid et al., 2020b) is a value-based multi-agent reinforcement learning method that leverages centralized training to learn decentralized policies by combining individual agent values in a monotonic way, ensuring efficient off-policy learning and policy consistency.

QPLEX (Wang et al., 2021) proposes a multi-agent value function factorization method based on a duplex dueling architecture, whose core idea is to strictly satisfy the Individual-Global-Max (IGM) principle through neural network architecture design, thereby enabling efficient learning while preserving policy expressiveness.

VAOS (Li et al., 2024c) is a Multi-agent value decomposition stabilization method that combines value averaging and operator switching to reduce overestimation errors, theoretically proven to enhance learning stability while maintaining optimal policy learning capabilities.

HGCN-MIX (Bai et al., 2022) combines hypergraph convolutional networks with value decomposition, leveraging hypergraphs to capture how the observations and actions of other agents influence the decision-making of a given agent.

GraphMIX (Naderializadeh et al., 2020) is a MARL framework that uses attention-based GNNs to factorize team value functions into individual agent policies with explicit credit assignment, enabling centralized training and decentralized execution.

SHAQ (Wang et al., 2022b) is an interpretable MARL algorithm based on Markov Shapley value theory, which solves the Shapley-Bellman optimality equation for value factorization in global reward games, demonstrating superior performance and theoretical consistency in multi-agent tasks.

ResQ (Shen et al., 2022) is a MARL value factorization method based on residual functions, which decomposes the joint value into a main and a residual part, providing a unified framework for mean-value and stochastic-value RL with theoretical guarantees for both IGM and distributional IGM.

W-QMIX (Rashid et al., 2020a) is a weighted-projection extension of QMIX that reweights joint-action targets during learning to overcome monotonic mixing's representational limitation and provably recover the correct greedy action, leading to improved performance in cooperative MARL tasks.

5.4. Evaluation metrics

To comprehensively evaluate the effectiveness of our method, we conduct experiments on both matrix games and the StarCraft Multi-Agent Challenge (SMAC). For matrix games, the primary evaluation metric is the average reward value, which directly reflects the agents' ability to maximize returns under different interaction settings. For SMAC, we adopt several widely used metrics, including the median test win rate (computed as the median over six independent runs to mitigate randomness) and the average episode length (measuring the temporal efficiency of achieving objectives). However, these metrics mainly reflect whether a task is successfully completed and how quickly it is achieved, focusing on outcome-level and temporal efficiency. They do not explicitly characterize how effectively agents coordinate during the process, nor do they capture the cost-aware quality of cooperation underlying a successful outcome.

Table 4
Median test winning rate in SMAC.

Scenarios	Models										
	HIVE	QTRAN (Son et al., 2019)	QMIX (Rashid et al., 2020b)	QPLEX (Wang et al., 2021)	VAOS (Li et al., 2024c)	HGCN-MIX (Bai et al., 2022)	GraphMIX (Naderializadeh et al., 2020)	SHAQ (Wang et al., 2022b)	ResQ (Shen et al., 2022)	OW-QMIX (Rashid et al., 2020a)	CW-QMIX (Rashid et al., 2020a)
2s_vs_1sc	**100%**	0%	96%	**100%**	**100%**	95%	**100%**	88%	**100%**	**100%**	**100%**
2s3z	99%	10%	99%	99%	**100%**	90%	0%	95%	96%	**100%**	94%
1c3s5z	**100%**	30%	97%	90%	97%	90%	90%	90%	96%	98%	98%
3s5z	94%	3%	85%	**95%**	85%	0%	9%	90%	**95%**	18%	56%
5m_vs_6m	**82%**	67%	72%	65%	42%	62%	50%	70%	28%	70%	62%
8m_vs_9m	88%	2%	81%	43%	77%	63%	58%	62%	83%	**90%**	60%
2c_vs_64zg	**96%**	10%	23%	20%	74%	7%	74%	4%	88%	71%	79%
4m_vs_5m	**63%**	0%	8%	50%	25%	30%	21%	47%	13%	22%	30%
bane_vs_bane	**100%**	**100%**	85%	**100%**	**100%**	**100%**	90%	**100%**	85%	**100%**	**100%**
3s_vs_5z	96%	0%	70%	80%	40%	85%	73%	87%	**97%**	90%	95%
MMM2	**93%**	2%	60%	74%	73%	83%	0%	0%	80%	30%	4%
6h_vs_8z	**75%**	2%	5%	5%	9%	68%	68%	10%	2%	3%	2%
Average	**90.5%**	18.8%	65.1%	68.4%	68.5%	64.4%	52.8%	61.9%	71.9%	66.0%	65.0%

Table 5
Median test winning rate in SMACv2.

Scenarios	Models										
	HIVE	QTRAN (Son et al., 2019)	QMIX (Rashid et al., 2020b)	QPLEX (Wang et al., 2021)	VAOS (Li et al., 2024c)	HGCN-MIX (Bai et al., 2022)	GraphMIX (Naderializadeh et al., 2020)	SHAQ (Wang et al., 2022b)	ResQ (Shen et al., 2022)	CW-QMIX (Rashid et al., 2020a)	OW-QMIX (Rashid et al., 2020a)
terran_5_vs_5	**83%**	40%	52%	0%	64%	26%	56%	61%	50%	58%	64%
terran_10_vs_10	**89%**	33%	61%	50%	78%	22%	55%	50%	50%	55%	75%
protoss_10_vs_10	**83%**	10%	65%	54%	47%	8%	18%	30%	40%	47%	48%
zerg_10_vs_10	**85%**	12%	56%	43%	67%	15%	46%	40%	55%	50%	61%
zerg_20_vs_20	**80%**	20%	45%	18%	35%	17%	31%	35%	56%	26%	63%
Average	**84%**	23%	56%	33%	59%	18%	41%	43%	50%	47%	62%

$Q_1 \backslash Q_2$	**0.27**(A)	0.08(B)	0.17(C)
0.25(A)	8.00	-12.08	-11.99
0.07(B)	-12.07	6.01	-0.03
0.16(C)	-11.99	0.05	6.96

(a) HIVE: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	**0.25**(A)	0.11(B)	0.15(C)
0.25(A)	7.98	7.85	7.88
0.11(B)	7.85	7.71	7.74
0.14(C)	7.88	7.74	7.77

(b) QTRAN: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	**0.05**(A)	-0.03(B)	-0.03(C)
0.05(A)	8.00	-11.99	-11.99
-0.03(B)	-11.99	6.01	-0.16
-0.04(C)	-11.99	-0.11	6.73

(c) ResQ: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	-19.02(A)	-1.15(B)	**0.75**(C)
-14.49(A)	-12.01	-12.01	-12.01
-1.14(B)	-12.00	-0.10	0.27
0.75(C)	-12.00	0.25	7.18

(d) QMIX: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	**0.52**(A)	-20.93(B)	-21.08(C)
0.43(A)	8.15	-10.53	-10.53
-17.55(B)	-10.53	-10.53	-10.53
-17.57(C)	-10.53	-10.53	-10.53

(e) CW-QMIX: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	**0.09**(A)	-1.21(B)	-1.22(C)
0.09(A)	7.99	-8.69	-8.69
-1.32(B)	-8.69	-8.69	-8.69
-1.31(C)	-8.69	-8.69	-8.69

(f) OW-QMIX: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	-15.06(A)	-2.66(B)	**3.44**(C)
1.12(A)	6.99	-11.63	-11.96
-17.37(B)	-12.02	5.98	0.00
3.50(C)	-11.90	0.00	7.00

(g) QPLEX: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	26.88(A)	-0.61(B)	**0.32**(C)
-25.31(A)	-10.27	-10.27	-10.27
-0.61(B)	-10.27	0.51	0.75
0.32(C)	-10.27	0.69	7.13

(h) VAOS: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	-17.85(A)	-1.36(B)	**0.61**(C)
-20.53(A)	-11.93	-11.92	-11.91
-1.43(B)	-11.91	-0.19	0.18
0.63(C)	-11.91	0.19	7.13

(i) HGCN-MIX: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	-11.74(A)	0.33(B)	**3.23**(C)
-11.79(A)	-11.82	-11.74	-11.71
0.34(B)	-11.74	-2.93	0.12
3.23(C)	-11.71	0.12	6.71

(j) GraphMIX: Q_1, Q_2, Q_{tot}

$Q_1 \backslash Q_2$	-2.99(A)	-0.33(B)	**3.50**(C)
-2.99(A)	-31.43	-16.05	-11.93
-0.32(B)	-16.10	-0.81	-0.01
3.47(C)	-11.99	0.25	6.97

(k) SHAQ: Q_1, Q_2, Q_{tot}

Fig. 5. Reward matrices of matrix games and value functions of approximately optimal policy. **Bold** text indicates greedy actions, red denotes correctly estimated optimal actions, while blue represents incorrectly estimated ones.

To address this limitation, we propose the collaborative score, a novel metric designed to quantify coordination quality by jointly considering ally survivability and enemy elimination. Unlike the binary win rate, which cannot distinguish between a "pyrrhic victory" (high casualties) and a "flawless victory" (zero casualties), this score serves as a normalized loss-exchange ratio. Specifically, as defined in Eq. (48), the metric increases only when agents maximize enemy defeats while simultaneously preserving their own units. This explicitly reflects the method's ability to achieve victory with minimal casualty costs, thereby offering a more granular and complementary insight into collaborative effectiveness beyond standard metrics.

$$\text{Score} = \frac{1}{2} \times \left(1 - \frac{I_{\text{deaths}}}{I_{\text{num}}}\right) + \frac{1}{2} \times \frac{E_{\text{deaths}}}{E_{\text{num}}}, \tag{48}$$

where I_{deaths} and E_{deaths} represent the number of deaths of the allies and enemies, respectively, and I_{num} and E_{num} represent the total number of allies and enemies, respectively.

5.5. Matrix game

We evaluate the approximation capabilities of different methods in the matrix game. As shown in Fig. 5, We evaluate the approximation capabilities of different methods in the matrix game. As shown in Fig. 5, most value-decomposition baselines (e.g., QMIX, QPLEX, VAOS, HGCN-MIX, GraphMIX, and SHAQ) converge to suboptimal solutions, typically preferring the suboptimal joint action (C, C) rather than the optimal (A, A). In contrast, HIVE, QTRAN, ResQ, CW-QMIX, and OW-QMIX successfully recover the optimal policy, all identifying (A, A) as the maximizing joint action (with $Q_{tot} \approx 8.00$, 7.98, 8.00, 8.15, and 7.99, respectively).

Notably, although QTRAN is able to converge to the optimal policy, its learned joint state-action value function Q_{tot} exhibits severe overestimation errors. This phenomenon is likely attributable to QTRAN's constraint relaxation mechanism, which tends to inflate value estimates in complex action spaces. Similarly, CW-QMIX slightly overestimates the optimal payoff (8.15 vs. 8.00), indicating that relaxing monotonicity alone does not necessarily guarantee faithful value reconstruction. OW-QMIX, on the other hand, produces an almost unbiased estimate at the optimum (7.99), but its value landscape remains highly distorted for non-optimal joint actions (e.g., many entries collapse to large negative values), suggesting that its weighting scheme may trade off reconstruction fidelity for selecting the correct maximizing action.

In comparison with ResQ, HIVE attains substantially lower reconstruction error in the matrix game, reducing the mean absolute error by nearly 50% (0.033 vs. 0.066) and the maximum error by about 70% (0.08 vs. 0.27). It also exhibits smaller deviations on neutral interactions with true payoff 0, where the absolute error remains within 0.05 for HIVE but exceeds 0.11 for ResQ. Moreover, HIVE provides a more faithful estimate of the suboptimal positive payoff at (C, C), yielding 6.96 as opposed to 6.73 for ResQ.

Fig. 6 presents the original reward distribution of the two-agent, three-action matrix game together with those reconstructed by several representative methods. HIVE's reconstructed reward distribution (Fig. 6(b)) preserves the highest structural similarity to the original matrix, highlighting the effectiveness of our approach. ResQ (Fig. 6(d)) also produces a distribution almost indistinguishable from HIVE's, which confirms the consistency of their approximations in this simple matrix game. Nevertheless, unlike ResQ, HIVE provides a theoretically grounded factorization through the hypergraph-weighted position value, ensuring both allocation efficiency and the satisfaction of the IGM principle. Therefore, HIVE exhibits superior performance over ResQ when the number of agents increases and cooperative structures become more complex, a trend that is reflected in larger-scale matrix games with six or ten agents, which we will discuss later. By contrast, Fig. 6(c) illustrates QTRAN's severe value overestimation during policy learning. Although HGCN-MIX (Fig. 6(e)) incorporates hypergraph structures, its graph-convolution-based mechanism does not meet the necessary and sufficient conditions of the IGM principle, leading to suboptimal capture of cooperative patterns. QPLEX (Fig. 6(f)) yields a

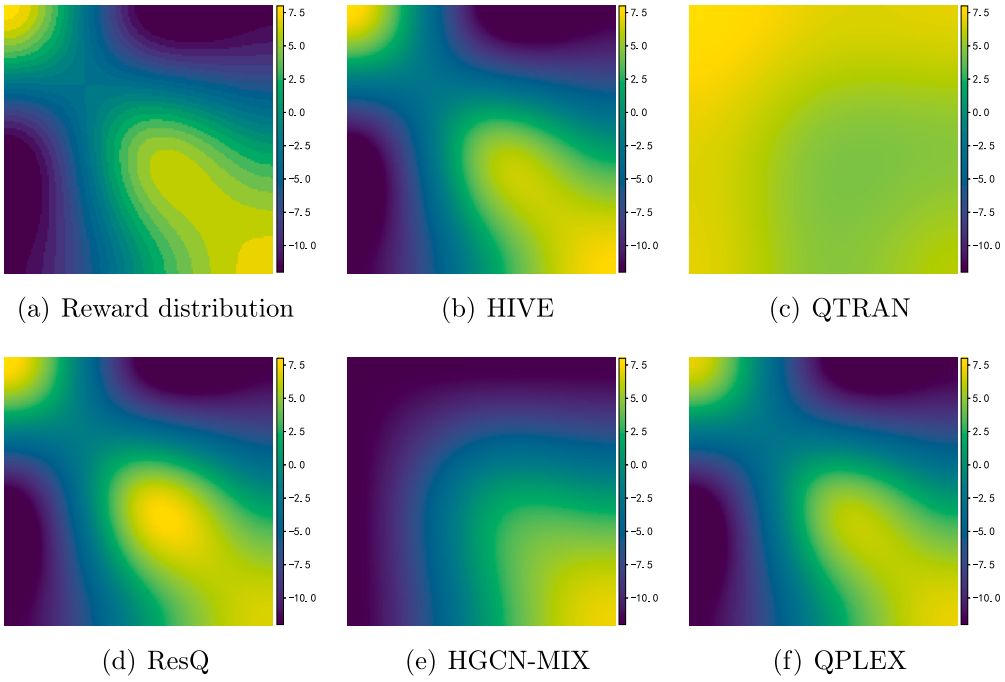


Fig. 6. Matrix game reward distribution.

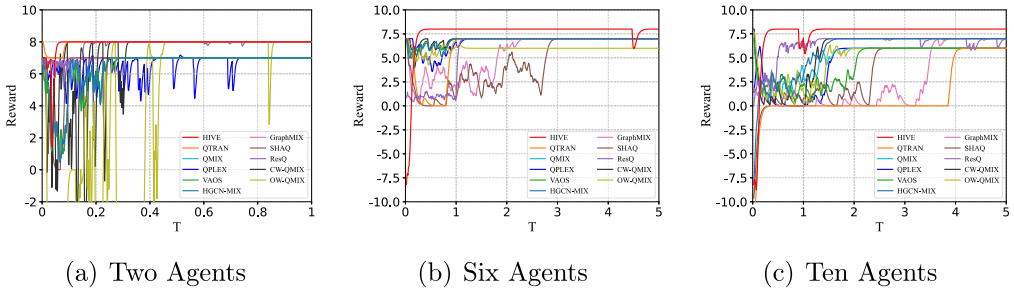


Fig. 7. Matrix game rewards under three different configurations. Here, the x-axis represents the number of training iterations ($\times 10^4$).

distribution closer to the original matrix, but it struggles to converge to the optimal solution due to the minimal reward gap between strategies.

Subsequently, we extend the matrix game to scenarios with 6 and 10 agents, with results shown in Fig. 7. In the 6-agent configuration (Fig. 7(b)), only HIVE converges to the optimal policy, while all other algorithms settle on suboptimal solutions. These results demonstrate HIVE's ability to decompose state-action values with a small approximation gap. Thanks to the incorporation of hypergraph theory, our method maintains performance as the number of agents increases, exhibiting superior scalability compared to QTRAN. In contrast, ResQ, which converges to the optimal solution in the two-agent setting, fails to retain this property as the scale grows and only converges to a suboptimal policy in the six-agent scenario. Similarly, CW-QMIX and OW-QMIX, although they relax the monotonic mixing constraint of QMIX, they also fail to converge to the optimal solution in both the 6-agent and 10-agent settings. The performance degradation of these baseline methods as the agent scale grows likely stems from their lack of mechanisms to capture higher-order dependencies, which become increasingly important in larger-scale environments.

When scaling to 10-agent scenarios (Fig. 7(c)), the performance divergence among methods becomes more pronounced. QTRAN fails to achieve stable convergence, while GraphMIX and HGCN-MIX continue to converge to suboptimal solutions, indicating that graph-based approaches are more robust to increasing agent populations. ResQ, benefiting from its residual correction mechanism, manages to converge but only to a suboptimal solution, CW-QMIX and OW-QMIX also show a similar trend. Notably, across all three matrix game configurations, HIVE shows a significant convergence speed advantage: it converges within an average of 2000 steps, whereas QTRAN requires around 17,000 steps. This efficiency gain is attributed to the use of hypergraph-weighted Position value, which allows each agent to precisely evaluate the marginal contribution of its actions to coalitions. As a result, our approach avoids the local optimum traps that often hinder conventional methods during policy exploration, thereby facilitating the discovery of globally optimal collaborative strategies.

To further evaluate the expressive capacity of HIVE, we extend the two-agent matrix game from a 3-dimensional to a 21-dimensional action space. The results are shown in Fig. 8, where six representative methods are visualized. HIVE (Fig. 8(a)) successfully attains the optimal solution and produces a reward distribution that closely matches the ground-truth distribution (Fig. 8(c)). This superiority arises from the fact that HIVE is built upon a game-theoretic decomposition that not only satisfies but also establishes the necessary and sufficient conditions for the Individual-Global-Max (IGM) principle (Theorem 2 provides the sufficient condition, while Theorem 3 establishes the necessary condition), ensuring theoretically sound and practically effective value factorization. QTRAN (Fig. 8(b)) also identifies the optimal policy, but its estimates exhibit severe overestimation, leading to large deviations from the true distribution.

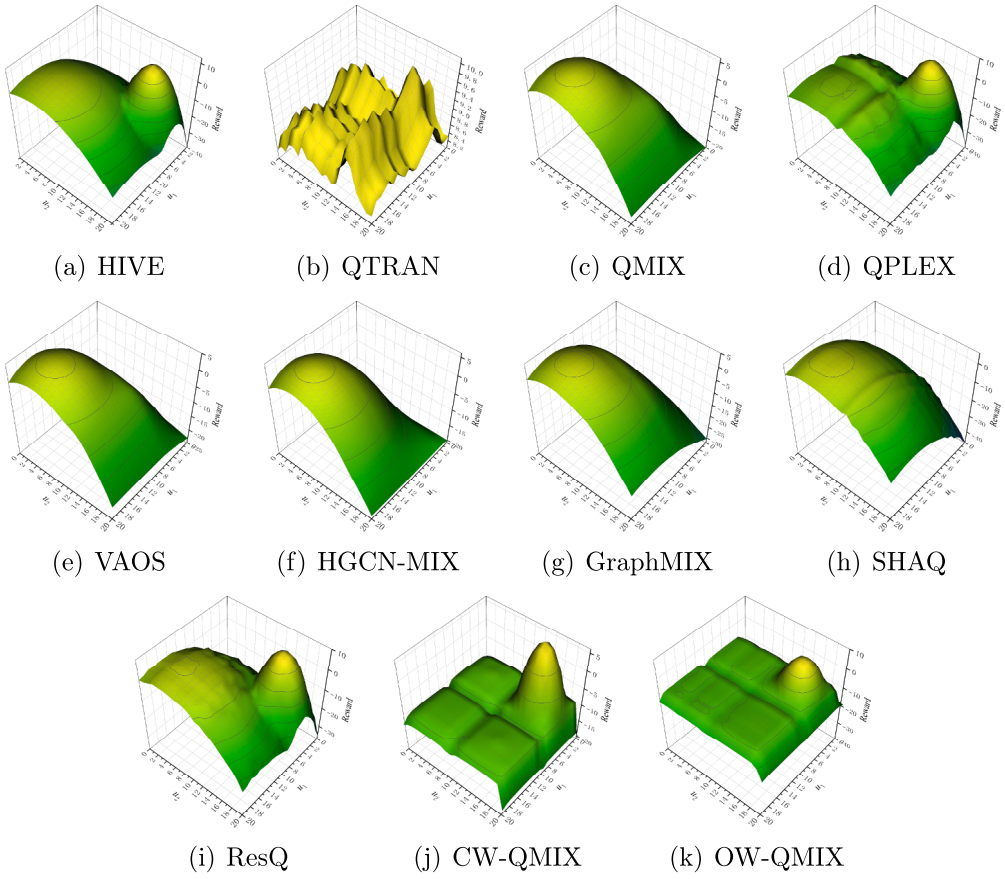


Fig. 8. Reward distribution of the matrix game with 21-dimensional action space.

QPLEX (Fig. 8(d)) and ResQ (Fig. 8(i)) achieve optimal rewards as well, but their guarantees are typically based on sufficiency rather than a full characterization of the IGM condition. We further include two WQMIX variants, CW-QMIX (Fig. 8(j)) and OW-QMIX (Fig. 8(k)). Both variants can also converge to the highest-reward region, confirming that relaxing the monotonicity constraint improves expressiveness. However, their learned landscapes still exhibit substantial discrepancies away from the optimum: CW-QMIX tends to introduce noticeable global warping/overestimation in non-optimal areas, while OW-QMIX shows pronounced distortions with large flat regions and sharp spikes, indicating that recovering the correct maximizer does not necessarily imply faithful reconstruction of the full reward/value distribution. Consequently, their approximations, despite reaching optimal rewards, tend to be less stable and deviate more from the ground-truth distribution compared with HIVE. In contrast, HGCN-MIX (Fig. 8(f)) and SHAQ (Fig. 8(h)) converge only to suboptimal rewards, yielding biased estimates. The remaining baselines, including QMIX, VAOS, and GraphMIX, display qualitatively similar patterns and are omitted for brevity. Taken together, HIVE exhibits the highest fidelity to the ground-truth distribution, with its Position-value-based hypergraph decomposition highlighting both theoretical rigor and superior expressive capacity relative to all baselines.

5.6. SMAC

We further evaluate HIVE on the SMAC benchmark and its extension, SMACv2. For fairness, all algorithms are trained with identical hyperparameter settings and the same optimizer.

5.6.1. Performance analysis

As shown in Fig. 9, we evaluate HIVE and the baseline methods across 12 different scenarios from both benchmarks. Each method is run independently under 6 distinct random seeds. The results report the median performance together with the inter-quartile range (25th-75th percentiles).

HIVE achieved an average win rate of 87.5% across eight SMAC scenarios (Fig. 9(a)–(h)), outperforming the best baseline, ResQ (62.4%), by 25.1 percentage points. It also converges faster and reaches higher asymptotic win rates than all baselines, showing strong robustness and scalability.

In the super-hard MMM2 scenario (Fig. 9(g)), which involves heterogeneous units and fine-grained cooperation, HIVE achieves 93%, while HGCN-MIX reaches 83%. Both use hypergraph structures to capture multi-lateral dependencies, but HIVE's game-theoretic value decomposition further improves coordination and yields better performance. In the 6h_vs_8z scenario (Fig. 9(h)), HIVE obtains 75%, compared with 68% for HGCN-MIX and GraphMIX. These results show that hypergraph- and graph-based methods are effective in this setting, while other baselines remain near zero. HIVE still leads HGCN-MIX by 7 points, confirming the advantage of game-theoretic decomposition.

By contrast, methods that succeed in matrix games fail to scale to SMAC. QTRAN converges to the optimal joint action in matrix games but reaches only 14.3% on SMAC because of Q-value overestimation. ResQ also achieves optimal policies in matrix games but struggles in complex tasks, as it lacks mechanisms to capture higher-order dependencies.

Across the four SMACv2 scenarios (Fig. 9(i)–(l)), HIVE achieves an average win rate of 84%, outperforming the best baseline, QMIX (56%),

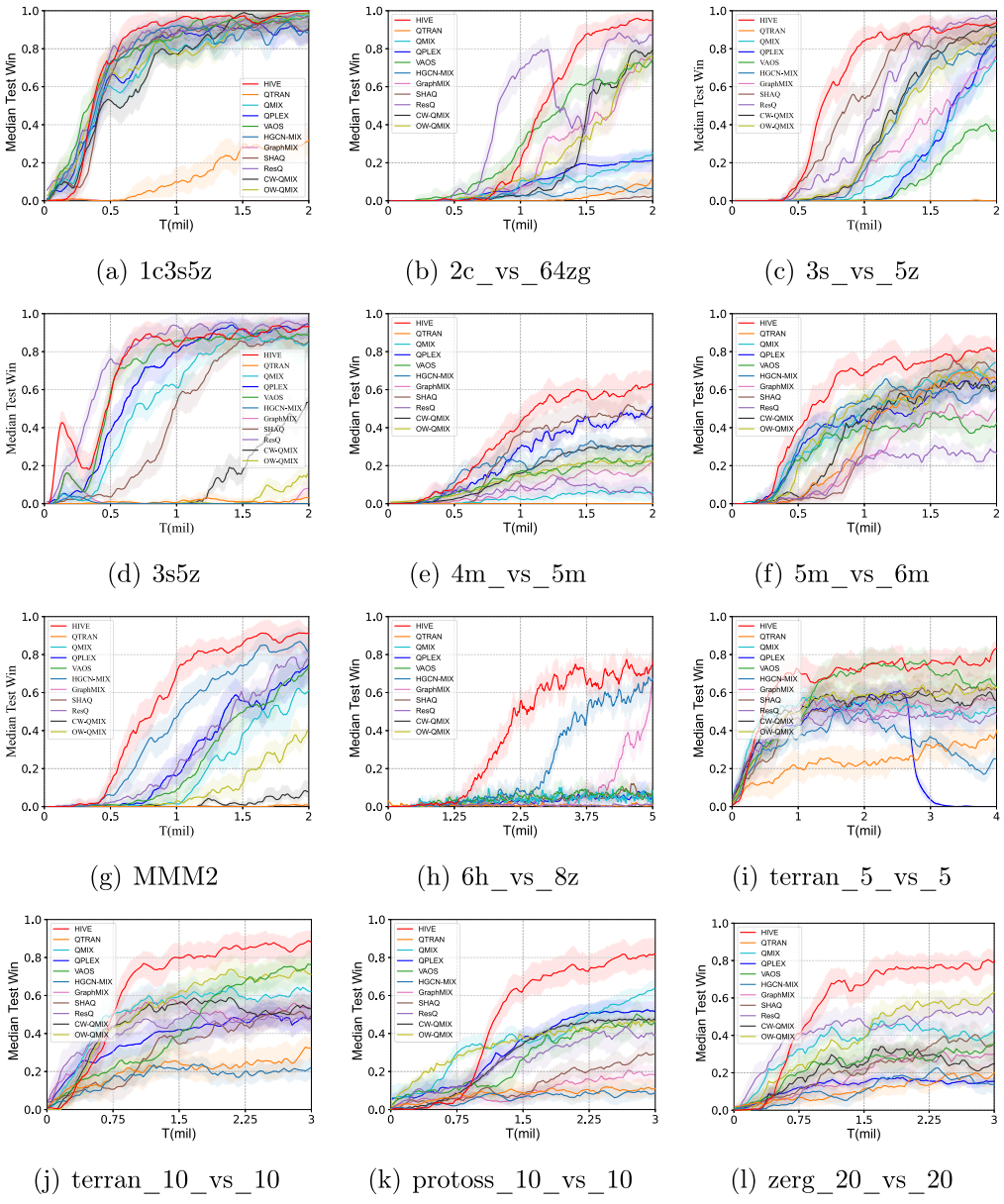
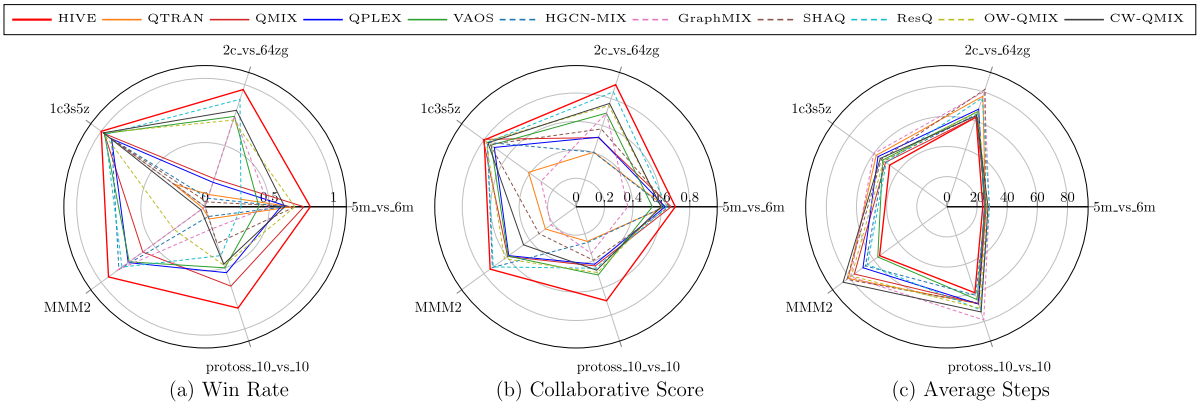


Fig. 9. Median Test winning rate for scenarios of SMAC and SMACv2. Here, the x-axis represents the number of training iterations measured in millions.

Table 6
Complexity-based estimated wall-clock training time (hours, mean ± range) across selected SMAC and SMACv2 scenarios.

Method	3s_vs_5z	5m_vs_6m	MMM2	terran_10_vs_10	protoss_10_vs_10	Mean
HIVE	30.4 ± 2.9	29.1 ± 2.8	39.8 ± 3.8	67.7 ± 6.6	69.5 ± 6.8	47.3 ± 4.6
GraphMIX	34.0 ± 3.4	32.5 ± 3.3	45.3 ± 4.5	78.8 ± 7.8	81.0 ± 8.1	54.3 ± 5.4
SHAQ	26.8 ± 2.7	25.6 ± 2.6	35.6 ± 3.5	62.0 ± 6.2	63.8 ± 6.3	42.8 ± 4.3
QTRAN	26.1 ± 2.6	25.0 ± 2.5	34.7 ± 3.4	60.3 ± 6.0	62.1 ± 6.2	41.6 ± 4.2
HGCN-MIX	29.3 ± 3.0	28.0 ± 2.8	39.0 ± 3.9	67.7 ± 6.8	69.6 ± 6.9	46.7 ± 4.7
QPLEX	25.9 ± 2.6	24.7 ± 2.5	34.4 ± 3.4	59.9 ± 6.0	61.5 ± 6.2	41.3 ± 4.1
VAOS	23.8 ± 2.4	22.8 ± 2.3	31.7 ± 3.2	55.2 ± 5.6	56.7 ± 5.7	38.0 ± 3.8
ResQ	24.5 ± 2.5	23.4 ± 2.4	32.6 ± 3.2	56.7 ± 5.7	58.4 ± 5.9	39.1 ± 3.9
QMIX	22.4 ± 2.3	22.0 ± 2.3	30.9 ± 3.0	53.4 ± 5.3	53.0 ± 5.5	36.3 ± 3.7
CW-QMIX	23.8 ± 2.4	22.8 ± 2.3	31.7 ± 3.2	55.2 ± 5.6	56.7 ± 5.7	38.0 ± 3.8
OW-QMIX	23.4 ± 2.4	22.4 ± 2.3	31.1 ± 3.1	54.2 ± 5.4	55.7 ± 5.6	37.4 ± 3.7



(a) Win Rate (b) Collaborative Score (c) Average Steps

Fig. 10. Visual comparison of the average game win rate, collaborative score, and average step length across some scenarios in SMAC and SMACv2.

by 28 percentage points. This shows that HIVE not only converges faster but also scales more effectively than all alternatives, maintaining strong performance across both small- and large-scale tasks. For example, in the large-scale zerg_20_vs_20 scenario (Fig. 9(l)), HIVE achieves a win rate of 80%, whereas the best baseline, ResQ, reaches only 56%, and most other methods fall below 50%. This superiority can be attributed to HIVE's hypergraph game-theoretic structure: by employing the hypergraph-weighted Position value, it not only captures higher-order dependencies but also performs value decomposition that satisfies the IGM principle, thereby enabling effective and stable coordination in complex environments. The scenario involves 40 agents with dense and overlapping interactions, which poses significant modeling challenges and further highlights HIVE's advantage in large-scale tasks.

Tables 4 and 5 present results on a broader set of SMAC and SMACv2 scenarios, respectively. Across both benchmarks, HIVE achieves the best performance, confirming its robustness under varying levels of complexity and stochasticity. The best-performing model in each scenario is highlighted in bold, while the second-best model is underlined. Overall, HIVE consistently demonstrates strong performance, attaining the highest average win rate of 87.25% across SMAC and SMACv2, which corresponds to a 26.3% improvement over the best baseline, ResQ.

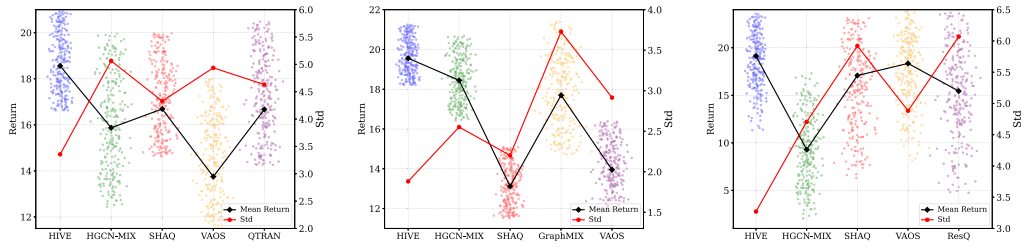
In the relatively simple and small-scale scenarios 2s3z and 3s5z, HIVE's performance is slightly lower–by 1% and 5%, respectively–than the best-performing methods. This gap is likely due to the limited number of agents and the near-linear cooperative topology, where lower-order interactions are sufficient to capture the key dependencies, reducing the benefit of modeling higher-order hyperedges. Notably, in another small-scale scenario, 2c_vs_64zg, HIVE achieves the best performance. This scenario introduces complexity through a large number of enemy units, where cooperation becomes more intricate and difficult to capture with simple local interactions. Here, HIVE's hypergraph mechanism effectively models higher-order cooperative relationships, thereby demonstrating superior performance in complex strategic collaborations.

In Fig. 10, we present a radar plot comparing HIVE with baseline methods on both SMAC and SMACv2 benchmarks, using three evaluation metrics: Win Rate, Collaborative Score, and Average Steps. In Fig. 10(a), we compare performance across different scenarios. On average, HIVE achieves the highest win rate of 89.3%, outperforming the best baseline, VAOS (67.2%), by 22.1 percentage points. We also evaluate advanced weighted value factorization variants, OW-QMIX and CW-QMIX, which achieve average win rates of 63.4% and 58.0%, respectively, while they exhibit strong performance in specific scenarios like 1c3s5z, HIVE maintains a clear advantage overall. This result highlights HIVE's superior stability and generalization across different scenarios. As shown in Fig. 10(b), HIVE achieves the highest Collaborative Score with an average of 0.76 across five scenarios, outperforming the best baseline, ResQ, which reaches 0.68, as well as OW-QMIX (0.648) and CW-QMIX (0.611). This advantage indicates that HIVE optimizes the loss-exchange ratio and effectively avoids pyrrhic victories, where wins are achieved at the cost of heavy allied casualties. By actively preserving unit health while maximizing enemy elimination, HIVE demonstrates superior tactical coordination, particularly in complex scenarios like protoss_10_vs_10, where it outperforms the best baseline (VAOS) by 19.06 percentage points. This substantial gap confirms that HIVE promotes sophisticated cooperative behaviors, ensuring tactical efficiency that standard win-rate metrics fail to capture. Finally, we evaluate the average steps of HIVE and the baseline methods (Fig. 10(c)). HIVE, with an average game length of 49.72 steps, achieves the most efficient coordination, outperforming the best baseline, VAOS, which requires 53.15 steps, and demonstrating significantly faster task completion than OW-QMIX (58.88 steps) and CW-QMIX (60.83 steps). Additionally, HIVE shows a stronger advantage in game steps in the 1c3s5z and MMM2 scenarios, where agents are highly heterogeneous. This result may be attributed to the hypergraph-weighted Position value in HIVE, which facilitates the capture of asymmetric agent interaction patterns.

5.6.2. *Stability analysis*

In Fig. 11, we present the return distributions of HIVE and several baselines on three maps: SMAC (5m_vs_6m and 6h_vs_8z) and SMACv2 (terran_10_vs_10). HIVE achieves the highest average return (19.08) and the lowest standard deviation (2.84) across all environments, indicating that it attains high returns with low variance and thus exhibits notable stability and robustness. We attribute this stability to two key theoretical advantages of our framework. First, unlike heuristic graph-based methods that may break consistency, HIVE enforces strict IGM consistency (Theorems 2 and 3), which prevents the joint policy from oscillating among inconsistent local optima and thereby smooths the learning process. Second, the game-theoretic credit assignment mechanism effectively reduces the variance of gradient estimation; compared with implicit mixing networks, HIVE allocates marginal contributions more fairly, further stabilizing the value decomposition procedure.

On the small and medium scale maps (5m_vs_6m and 6h_vs_8z), HIVE and SHAQ are the most stable. Specifically, HIVE attains average returns of 18.56 and 19.56 with standard deviations of 3.36 and 1.88, respectively, while SHAQ achieves standard deviations of 4.33 and 2.20, both lower than those of HGCN-MIX (5.06) and VAOS (4.94). This suggests that incorporating game-theoretic modeling into value decomposition helps better capture the relationship between individual and global values, thereby reducing training variance and improving stability in cooperative tasks. On the larger and more complex terran_10_vs_10 map, HIVE shows the strongest robustness, followed by HGCN-MIX; their average return/standard deviation are 19.13/3.27 and 9.30/4.71, respectively, both better than ResQ (6.07) and SHAQ (5.92). These results



(a) 5m_vs_6m (b) 6h_vs_8z (c) terran_10_vs_10

Fig. 11. Return distributions of different methods. Scatter plots show sampled returns, where larger standard deviations produce more dispersed points. The black line denotes the mean return, and the red line (right axis) indicates the standard deviation of each method.



Fig. 12. Visualization results in SMAC scenarios. The circles denote the hyperedges among agents. The colors of the circles are purely for visual distinction without any specific meaning. Additionally, for better clarity and intuitiveness, we do not draw all hyperedge connections but selectively display only the crucial ones.

further indicate that, as the number of agents increases and cooperation becomes more complex, hypergraph structures can better model higher-order dependencies, enabling a more accurate characterization of multi-hyperedge interaction relationships and improving robustness in large-scale cooperation.

5.6.3. Visualization of learned hyperedges

To investigate whether the learned hyperedges can capture meaningful cooperative patterns, we conduct a qualitative visual analysis of the dynamic interactions in representative SMAC scenarios (1c3s5z and MMM2) in Fig. 12. The results demonstrate that the key hyperedge structures generated by the model closely align with the high-order tactical maneuvers of human players. Specifically, in the 1c3s5z scenario, hyperedges precisely encapsulate agents executing specific tactical intents: front-line damage-absorbing units and back-line damage-dealing units are partitioned into separate hyperedges during “Kiting” and “Draw Fire”, while flanking units are dynamically clustered during “Pincer Movement” and “Ambush”. This successfully decomposes the global task into local high-order cooperative structures. Furthermore, in the heterogeneous MMM2 scenario, role-based coordination is remarkably precise. Medivacs and injured units are frequently and stably connected during “Healing”, while units attacking the same target are jointly encapsulated during the “Focus Fire” and “Hunting Down” phases. This proves that the model not only learns to allocate interaction structures based on heterogeneous roles but also effectively concentrates firepower resources at critical moments.

5.6.4. Wall-clock training time

As shown in Table 6, we report the complexity-based estimated wall-clock training time (hours, mean ± range) of different methods across selected SMAC/SMACv2 scenarios. The training time increases noticeably with the scenario scale: small-scale scenarios (e.g., 3s_vs_5z and 5m_vs_6m) typically take around 20–35 hours, while larger and more complex scenarios (e.g., terran_10_vs_10 and protoss_10_vs_10) require about 50–80 hours, indicating that the overall cost is mainly driven by scenario complexity.

From a method-wise comparison, HIVE has an average training time of 47.3 ± 4.6 hours, which is slightly higher than QMIX (36.3 ± 3.7) and QPLEX (41.3 ± 4.1), consistent with the additional computation introduced by our structural modeling and cooperation-aware information integration modules. Meanwhile, HIVE remains lower than the heavier GraphMIX (54.3 ± 5.4) and is comparable to HGCN-MIX (46.7 ± 4.7), suggesting that although our method is indeed somewhat slower, its overall complexity still falls within the typical range of mainstream graph-structure-based MARL methods.

Note that wall-clock time also depends on the underlying hardware and system environment. Thus, we report the workstation configuration in Table 3 for reproducibility and fair comparison.

5.7. Ablation study

5.7.1. Hyperedge number ablation

To investigate the impact of the number of hyperedges on HIVE, we conducted experiments in several scenarios from the SMAC benchmark and its more challenging extension, SMACv2, as well as matrix games, as shown in Fig. 13. We tested five settings for the number of hyperedges: 2, 4, 8, 16, and 32. The results show that performance peaks at 16 hyperedges in most scenarios. In contrast, for smaller-agent settings such as 2c_vs_64zg and 4m_vs_5m, optimal performance is already achieved with only 4 and 8 hyperedges, respectively. Since the theoretical maximum numbers of hyperedges in these two tasks are 3 and 15, further increasing the hyperedge count to 16 or 32 inevitably introduces redundancy. Nevertheless, HIVE’s performance remains stable, which can be explained by its satisfaction of the hyperedge redundancy payoff invariance (Theorem 1).

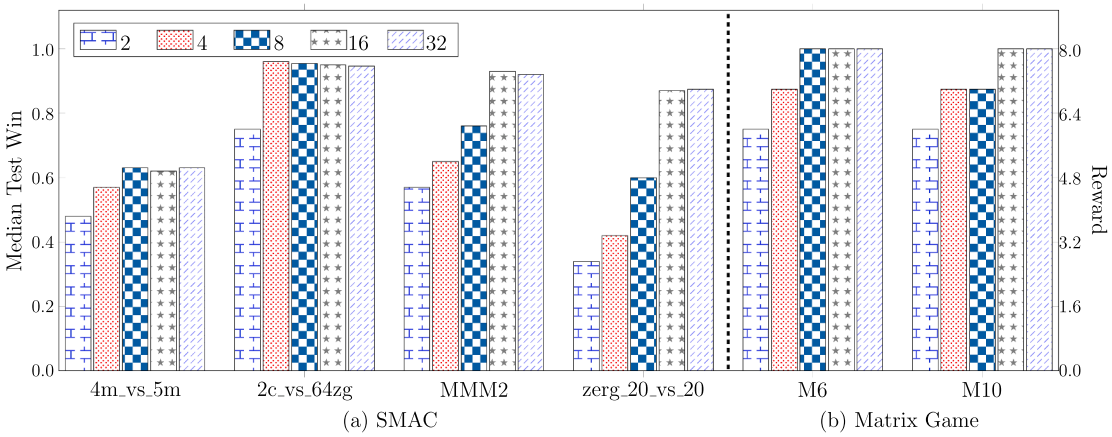


Fig. 13. The impact of the number of hyperedges on HIVE performance in the SMAC environment (including SMACv2 configurations) and matrix games. Here, M6 and M10 denote extended versions of the matrix game: M6 represents a scenario with 6 agents and an action space of 3, while M10 involves 10 agents.

Furthermore, in scenarios such as zerg_20_vs_20 and MMM2, which involve larger agent populations and greater heterogeneity, we observed that optimal performance is achieved with 16 hyperedges. Even though the theoretical maximum number of hyperedges is not reached at 32, increasing the number beyond 16 does not produce noticeable performance fluctuations. Similar patterns are observed in the matrix game experiments (Fig. 13(b)) with 6 and 10 agents. When the number of hyperedges is 2, HIVE converges to the third-best joint reward (6) in both cases. As the number of hyperedges increases, the model reaches the optimal joint action and attains the maximum reward (8) once the hyperedge count reaches 8 or 16. Further increases, although still below the theoretical maximum, bring no additional performance gains. Overall, these results indicate that 16 hyperedges are generally sufficient to capture agent interactions effectively, while larger values yield negligible improvements.

From a training perspective, m mainly controls the structural coverage (i.e., how many candidate interaction groups can be instantiated) of the constructed hypergraph. When m is too small, HIVE may undercover critical interaction patterns, leading to insufficient modeling capacity and slower or suboptimal convergence. Once m is large enough to cover the dominant coordination structures (e.g., $m = 16$ in most scenarios), further increasing m primarily introduces redundant hyperedges; due to the Redundant Hyperedge Payoff Invariance (Theorem 1), such redundancy does not change the optimal solution and thus yields negligible performance gains, while increasing the computational overhead in practice. Based on this observation, we adopt $m = 16$ as the default setting across SMAC/SMACv2 and matrix games, while using smaller m for small-agent scenarios where the interaction space is inherently limited.

5.7.2. Hyperparameter sensitivity analysis

To investigate how the similarity threshold γ in the AHC module affects the hypergraph topology and the resulting cooperative performance, and to validate its robustness under different task properties, we conduct a sensitivity analysis on four representative scenarios (MMM2, 6h_vs_8z, terran_10_vs_10, zerg_20_vs_20). As shown in Fig. 14, the hypergraph sparsity is strongly and negatively correlated with γ: as γ increases from 0.2 to 0.9, the average hyperedge degree (i.e., the average number of agents contained in each hyperedge) consistently decreases across all scenarios. When $\gamma \geq 0.8$, the average degree generally degenerates to around 1.0, indicating that an overly large threshold compresses hyperedges into near-singleton ones, thereby weakening the ability to capture higher-order dependencies. The performance curves are highly consistent with this topological change, suggesting that a moderate threshold achieves a better trade-off between topology and performance. In most scenarios, $\gamma \in [0.4, 0.6]$ yields optimal or the most stable win rates (e.g., MMM2, terran_10_vs_10, and zerg_20_vs_20 peak around $\gamma = 0.4$), whereas an excessively large γ that causes degree degeneration often leads to a notable performance drop, confirming that over-sparsification can disconnect necessary coordination links. Moreover, task properties significantly affect the sensitivity to sparsity. In the homogeneous scenario 6h_vs_8z, where units share the same functionality and rely more on group-level synchronization, the model is particularly sensitive to connectivity: increasing γ from 0.6 to 0.8 quickly degenerates the average hyperedge degree from 3.125 to 1.0, accompanied by a sharp win-rate decline from 0.75 to 0.30. In contrast, for heterogeneous scenarios (e.g., MMM2 and zerg_20_vs_20), a moderately sparse topology is more beneficial for emphasizing critical cross-role interactions while suppressing ineffective couplings, thus leading to better stability and robustness under moderate γ settings.

To determine an appropriate number of Monte Carlo sampling iterations K, we conduct an ablation study with different K values. As shown in Fig. 15, $K = 10$ achieves the best overall performance on both MMM2 and zerg_20_vs_20, exhibiting faster convergence, higher final win rates, and a narrower interquartile range (shaded area), which indicates a better trade-off between contribution-estimation accuracy and training stability. In contrast, with fewer samples $K = 5$, the algorithm becomes markedly less stable and learns more slowly. This is particularly evident in MMM2, where the blue curve remains at a low win rate for an extended period and shows substantially larger variability (wider shading). These results suggest that insufficient sampling yields a noisy approximation of the counterfactual baseline, leading to high-variance gradient estimates that undermine reliable credit assignment and hinder effective policy optimization. On the other hand, increasing the sampling budget $K = 20$ can reduce estimation variance and slightly improve late-stage stability, but the gains do not justify the additional computational cost. Specifically, more samples per update reduce the number of effective optimization steps per unit time, resulting in lower training efficiency and slower performance improvement, with win-rate growth lagging behind $K = 10$ under the same training budget.

5.7.3. Component ablation

To evaluate the approximation ability of the neural network for each module, we conduct ablation studies in both the SMAC and matrix game environments, as shown in Fig. 16. Here, HIVE represents the full model. w\o GBI denotes the removal of the GBI module, and w\o DCI denotes the removal of the DCI module. w\o G&D removes both GBI and DCI modules. In this case, HIVE degrades to $Q_{tot} = \sum_{i \in \mathcal{N}} Q_i$, which is equivalent to the traditional VDN (Sunehag et al., 2018) method.

Fig. 16(a) presents the ablation results of the SAMC environment under four scenarios. Removing GBI reduces the average win rate by 22.75%, while removing DCI causes a much larger drop of 52%,

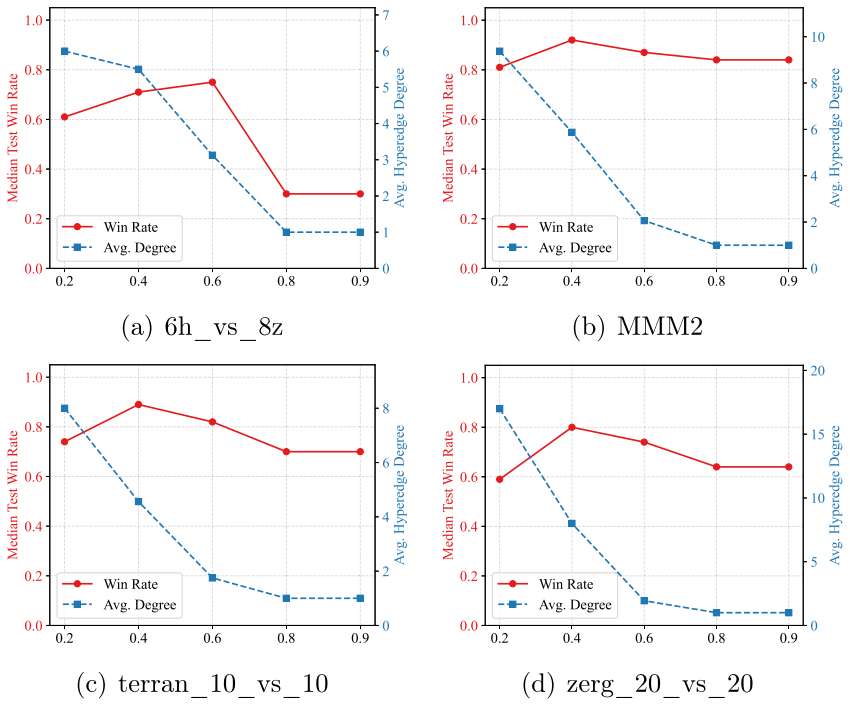


Fig. 14. Sensitivity analysis of the similarity threshold γ in the AHC across four representative scenarios. The horizontal axis represents the value of γ ranging from 0.2 to 0.9. The left vertical axis (red solid line) denotes the Median Test Win Rate, indicating the collaborative performance. The right vertical axis (blue dashed line) denotes the Average Hyperedge Degree, reflecting the sparsity/density of the generated hypergraph topology.

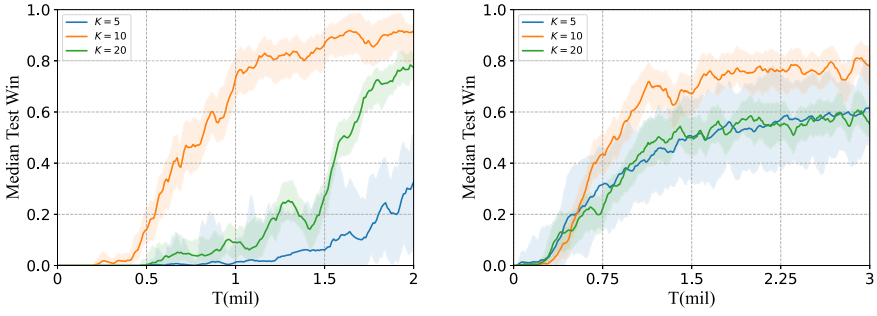


Fig. 15. Ablation study on the number of Monte Carlo sampling iterations K. The x-axis represents the number of sampling iterations, and the y-axis denotes the Median Test Win Rate.

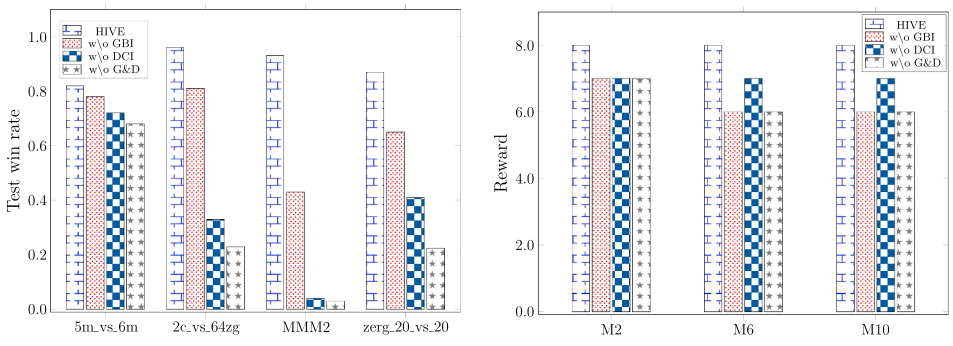


Fig. 16. Study diagram of HIVE component ablation.

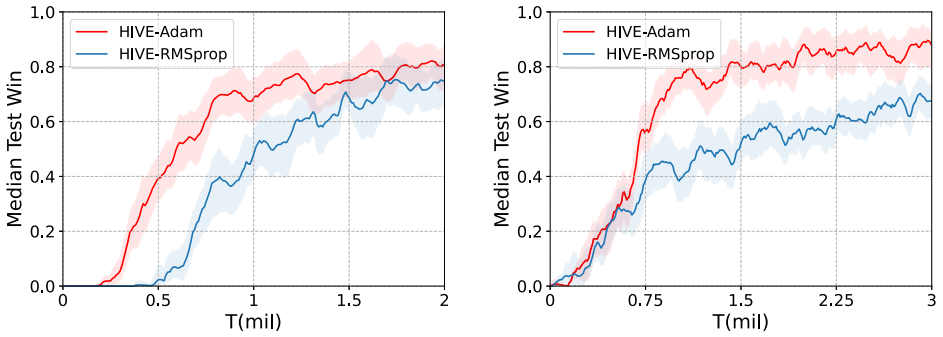


Fig. 17. Optimizer ablation on HIVE under Adam and RMSprop.

exceeding the impact of GBI by 29.25%. This may be because DCI captures higher-order cooperative structures through hypergraph game theory, making it a critical component for performance improvement. Although eliminating GBI renders the HIVE framework theoretically incomplete, the system still preserves partial cooperation due to the presence of DCI. When both GBI and DCI are removed, the win rate declines by 60.38%, confirming the necessity of both GBI and DCI for enhancing multi-agent cooperation modeling.

Fig. 16(b) shows the ablation results in the matrix game with different numbers of agents, where M2, M6, and M10 represent scenarios with 2, 6, and 10 agents, respectively. All ablation settings–w\o DCI, w\o GBI, and w\o G&D–significantly reduce the final reward performance and often lead to training instability or convergence to suboptimal policies. This is because HIVE theoretically decomposes the joint state-action value function Q_{tot} into Q_i, $\Phi_{\Gamma}^{\mathcal{H}}$, and $\Phi_{\Gamma^{\circ}}^{C_e}$. These components are approximated by the agent network, GBI, and DCI, respectively. Removing any of these modules disrupts this theoretical foundation, ultimately preventing the model from converging to the optimal solution.

5.7.4. Ablation study on optimizer choices

To quantitatively evaluate the impact of optimizer selection, we compare the performance of HIVE trained with Adam and RMSprop in Fig. 17. Adam yields consistently higher final performance and faster convergence across all scenarios. In 5m_vs_6m, HIVE-Adam converges to approximately 0.82, outperforming HIVE-RMSprop (0.74) with a relative improvement of 8%. In terran_10_vs_10, the performance gap widens further to around 15%, demonstrating a more prominent advantage in large-scale scenarios.

Furthermore, HIVE-Adam achieves faster early-stage learning and narrower interquartile ranges, implying more stable training dynamics. These results verify that optimizer selection exerts a non-negligible influence within the PyMARL framework. Nevertheless, HIVE maintains stable convergence trends with both optimizers, which demonstrates that our proposed method is not optimizer-specific and can benefit from the superior optimization efficiency of Adam.

6. Conclusion

In this paper, we propose HIVE, a hypergraph-based game-theoretic interactive value decomposition engine for multi-lateral agent cooperation. HIVE decomposes the joint state-action value function into three components: individual state-action values, a global cooperative baseline, and a dynamic cooperative residual. This decomposition is supported by the hypergraph-weighted Position value, which guarantees allocation efficiency through an existence theorem and ensures payoff invariance under redundant hyperedges. We further prove that HIVE provides both sufficient and necessary conditions for the IGM principle. To make this decomposition trainable in practice, we design the global baseline integrator and dynamic cooperation integrator modules, together with an attention-based hypergraph constructor that captures high-order dependencies and provides topological support. Experiments on SMAC, its SMACv2 extension, and matrix games show that HIVE consistently outperforms strong baselines, achieving an average win rate of 87.25% and converging to optimal joint actions while better fitting the reward distribution, thereby demonstrating the effectiveness of HIVE.

HIVE thus offers a principled solution for cooperative multi-agent tasks, demonstrating outstanding performance in multi-lateral collaboration scenarios. In future work, we plan to extend HIVE to cross-domain distributed decision-making problems, such as heterogeneous unmanned aerial vehicle 3D formation control and multi-type robot swarm coordination, to further investigate the scalability and robustness of hypergraph-based MARL in highly dynamic environments.

CRediT authorship contribution statement

Changdong Zhou: Writing – original draft, Visualization, Software, Methodology; **Ruofan Hu:** Validation, Software, Methodology, Formal analysis, Data curation; **Naiyao Wang:** Writing – review & editing, Visualization, Software, Investigation, Formal analysis, Conceptualization; **Ke Song:** Writing – review & editing, Validation, Methodology, Investigation, Formal analysis, Conceptualization; **Hongbo Liu:** Writing – review & editing, Supervision, Resources, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China (Grant No. 62176036), in part by the Fundamental Research Funds for the Central Universities (Grant No. 3132023523), and in part by the Natural Science Foundation of the Liaoning Province, China (Grant No. 2025-BS-0224).

References

Bai, Y., Gong, C., Zhang, B., Fan, G., Hou, X., & Lu, Y. (2022). Cooperative multi-agent reinforcement learning with hypergraph convolution. In *Proceedings of the international conference on neural networks* (pp. 1–8).

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

Ellis, B., Cook, J., Moalla, S., Samvelyan, M., Sun, M., Mahajan, A., Foerster, J., & Whiteson, S. (2023). SMACv2: An improved benchmark for cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, *36*, 37567–37593.

Gallici, M., Martín Muñoz, M., & Masmitja Rusiñol, I. (2023). TransfQMix: Transformers for leveraging the graph structure of multi-agent reinforcement learning problems. In *Proceedings of the international conference on autonomous agents and multiagent systems* (pp. 1679–1687).

Goeckner, A., Sui, Y., Martinet, N., Li, X., & Zhu, Q. (2024). Graph neural network-based multi-agent reinforcement learning for resilient distributed coordination of multi-robot systems. In *Proceedings of the international conference on intelligent robots and systems* (pp. 5732–5739).

Gorsane, R., Mahjoub, O., de Kock, R. J., Dubb, R., Singh, S., & Pretorius, A. (2022). Towards a standardised performance evaluation protocol for cooperative marl. In *Advances in neural information processing systems* (pp. 5510–5521). (*Vol. 35*).

He, B., Yang, N., Zhang, X., & Wang, W. (2024). Game theory and reinforcement learning in cognitive radar game modeling and algorithm research: A review. *IEEE Sensors Journal*, *24*(20), 31696–31711.

Hu, X., Guo, P., Li, Y., Li, G., Cui, Z., & Yang, J. (2025). TVDO: Tchebycheff value-decomposition optimization for multiagent reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, *36*(7), 12521–12534.

Huang, J., Yang, C., Wu, S., & Liu, Q. (2025). Distributed aggregative optimization algorithm for solving multi-robot formation problem. *IEEE Transactions on Control of Network Systems*, *12*(3), 1–12.

Li, C., Zhang, Y., Wang, J., Hu, Y., Dong, S., Li, W., Lv, T., Fan, C., & Gao, Y. (2024a). Optimistic value instructors for cooperative multi-agent reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 17453–17460). (*Vol. 38*).

Li, J., Kuang, K., Wang, B., Liu, F., Chen, L., Fan, C., Wu, F., & Xiao, J. (2022). Deconfounded value decomposition for multi-agent reinforcement learning. In *Proceedings of the international conference on machine learning* (pp. 12843–12856). PMLR (*Vol. 162*).

Li, M., Sun, H., Huang, Y., & Chen, H. (2024b). Shapley value: From cooperative game to explainable artificial intelligence. *Autonomous Intelligent Systems*, *4*(1), 2.

Li, P., Chen, S., Yuan, W., Hu, Z., & Chen, J. (2024c). VAOS: Enhancing the stability of cooperative multi-agent policy learning. *Knowledge-Based Systems*, *304*, 112474.

Li, Z., Wang, Q., Wang, J., & He, Z. (2024d). A flexible cooperative MARL method for efficient passage of an emergency CAV in mixed traffic. *IEEE Transactions on Intelligent Transportation Systems*, *25*(8), 8898–8912.

Liu, H., Chen, L., Qiao, Y., Lv, C., & Li, H. (2024). Reasoning multi-agent behavioral topology for interactive autonomous driving. In *Advances in neural information processing systems* (pp. 92605–92637). (*Vol. 37*).

Liu, K., Zhang, T., Xu, X., & Zhao, Y. (2025). Counterfactual value decomposition for cooperative multi-agent reinforcement learning. *Neural Networks*, *190*, 107692.

Mai, T., Nguyen, T. et al. (2024). Mimicking to dominate: Imitation learning strategies for success in multiagent games. In *Advances in neural information processing systems* (pp. 84669–84697). (*Vol. 37*).

Moldagalieva, A., Ortiz-Haro, J., Toussaint, M., & Hönig, W. (2024). db-cbs: Discontinuity-bounded conflict-based search for multi-robot kinodynamic motion planning. In *Proceedings of the international conference on robotics and automation* (pp. 14569–14575).

Naderializadeh, N., Hung, F. H., Soleyman, S., & Khosla, D. (2020). Graph convolutional value decomposition in multi-agent reinforcement learning. arXiv preprint arXiv:2010.04740.

Nie, Y., Liu, J., Liu, X., Zhao, Y., Ren, K., & Chen, C. (2024). Asynchronous multi-agent reinforcement learning-based framework for bi-level noncooperative game-theoretic demand response. *IEEE Transactions on Smart Grid*, *15*(6), 5622–5637.

Oliehoek, F. A., Spaan, M. T. J., & Vlassis, N. (2008). Optimal and approximate Q-value functions for decentralized POMDPs. *Journal of Artificial Intelligence Research*, *32*, 289–353.

Parada, L., Yu, K., & Angeloudis, P. (2025). IntNet: A communication-driven multi-agent reinforcement learning framework for cooperative autonomous driving. *IEEE Robotics and Automation Letters*, *10*(3), 2478–2485.

Peters, L., Bajcsy, A., Chiu, C.-Y., Fridovich-Keil, D., Laine, F., Ferranti, L., & Alonso-Mora, J. (2024). Contingency games for multi-agent interaction. *IEEE Robotics and Automation Letters*, *9*(3), 2208–2215.

Phan, T., Belzner, L., Gabor, T., Sedlmeier, A., Ritz, F., & Linnhoff-Popien, C. (2021). Resilient multi-agent reinforcement learning with adversarial value decomposition. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 11308–11316). (*Vol. 35*).

Rashid, T., Farquhar, G., Peng, B., & Whiteson, S. (2020a). Weighted QMIX: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning. In *Advances in neural information processing systems* (pp. 10199–10210). (*Vol. 33*).

Rashid, T., Samvelyan, M., De Witt, C. S., Farquhar, G., Foerster, J., & Whiteson, S. (2020b). QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. *The Journal of Machine Learning Research*, *21*(1), 7234–7284.

Samvelyan, M., Rashid, T., Schroeder de Witt, C., Farquhar, G., Nardelli, N., Rudner, T. G. J., Hung, C.-M., Torr, P. H. S., Foerster, J., & Whiteson, S. (2019). The StarCraft multi-agent challenge. In *Proceedings of the international conference on autonomous agents and multiagent systems* (pp. 2186–2188).

Shen, S., Qiu, M., Liu, J., Liu, W., Fu, Y., Liu, X., & Wang, C. (2022). ResQ: A residual Q function-based approach for multi-agent reinforcement learning value factorization. In *Advances in neural information processing systems* (pp. 5471–5483). (*Vol. 35*).

Son, K., Kim, D., Kang, W. J., Hostallero, D. E., & Yi, Y. (2019). QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *Proceedings of the international conference on machine learning* (pp. 5887–5896).

Su, K., Zhou, S., Jiang, J., Gan, C., Wang, X., & Lu, Z. (2024). Multi-agent alternate q-learning. In *Proceedings of the international conference on autonomous agents and multiagent systems* (pp. 1791–1799).

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K. et al. (2018). Value-decomposition networks for cooperative multi-agent learning based on team reward. In *Proceedings of the international conference on autonomous agents and multiagent systems* (pp. 2085–2087). (*Vol. 3*).

Wang, D., Liu, H., Wang, N., Wang, Y., Wang, H., & McLoone, S. (2022a). SEEM: A sequence entropy energy-based model for pedestrian trajectory all-then-one prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(1), 1070–1086.

Wang, J., Ren, Z., Liu, T., Yu, Y., & Zhang, C. (2021). QPLEX: Duplex dueling multi-agent Q-learning. In *International conference on learning representations*.

Wang, J., Zhang, Y., Gu, Y., & Kim, T.-K. (2022b). SHAQ: Incorporating shapley value theory into multi-agent Q-learning. In *Advances in neural information processing systems* (pp. 5941–5954). (*Vol. 35*).

Wang, J., Zhang, Y., Kim, T.-K., & Gu, Y. (2020). Shapley Q-value: A local reward approach to solve global reward games. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 7285–7292). (*Vol. 34*).

Wang, N., Zhang, B., Chi, H., Wang, H., McLoone, S., & Liu, H. (2024). DUEL: Depth visual ego-motion learning for autonomous robot obstacle avoidance. *The International Journal of Robotics Research*, *43*(3), 305–329.

Wei, Q., Li, Y., Zhang, J., & Wang, F.-Y. (2024). VGN: Value decomposition with graph attention networks for multiagent reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, *35*(1), 182–195.

Ye, D., Liu, Z., Sun, M., Shi, B., Zhao, P., Wu, H., Yu, H., Yang, S., Wu, X., Guo, Q. et al. (2020). Mastering complex control in moba games with deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 6672–6679). (*Vol. 34*).

Zhang, B., Bai, Y., Xu, Z., Li, D., & Fan, G. (2022). Efficient policy generation in multi-agent systems via hypergraph neural network. In *Advances in neural information processing systems* (pp. 219–230). Springer.

Zhang, J., Zhang, Y., Zhang, X. S., Zang, Y., & Cheng, J. (2024). Intrinsic action tendency consistency for cooperative multi-agent reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 17600–17608). (*Vol. 38*).

Zhang, X., Chen, J., Gan, M.-G., & Chen, H. (2025a). Influence enhanced sparse coordination graphs for multi-agent reinforcement learning. *Neural Networks*, *188*, 107454.

Zhang, Z., He, B., Cheng, B., & Li, G. (2025b). Bridging training and execution via dynamic directed graph-based communication in cooperative multi-agent systems. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 23395–23403). (*Vol. 39*).

Zhu, T., Shi, X., Xu, X., Gui, J., & Cao, J. (2024). HyperComm: Hypergraph-based communication in multi-agent reinforcement learning. *Neural Networks*, *178*, 106432.