

Emergence of Cooperation in Multi-Agent Reinforcement Learning via Coalition Labeling and Structural Entropy

Dingli Su* Hao Peng[†] Guangjie Zeng* Pu Li[‡] Angsheng Li* Yicheng Pan* 

Abstract

Multi-agent cooperation is essential for tasks that require collaboration to achieve optimal performance or cannot be completed by individual agents alone. These tasks often necessitate a divide-and-conquer strategy, where sub-goals are allocated to individual agents or groups. By integrating coalition formation concepts from cooperative game theory, we demonstrate the implicit learning of coalition formation and task assignments, resulting in emergent cooperative behavior. We propose a novel **COaLition LABELing** technique for Multi-Agent Reinforcement Learning (**COLLAB-MARL**) to encourage coalition formation and introduce a structural entropy measure to detect the emergence of coalitions and cooperative behavior. Compared to classical MARL methods, COLLAB-MARL is more effective, explainable, and easier to implement. Experiments on state-of-the-art cooperative MARL benchmarks show that our method’s mean return outperforms the strongest baselines by 8.4% on average. Additionally, visualization and structural entropy analysis reveal that COLLAB-MARL effectively learns meaningful cooperative behavior. The source code is available at <https://github.com/SELGroup/collab>.

Keywords: Cooperative AI, Multi-Agent Reinforcement Learning, Coalition Formation, Structural Entropy, Drone Swarm

1 Introduction

The emergence and evolution of cooperation among intelligent agents is a fundamental question that has intrigued researchers across various fields, from biology to artificial intelligence. Key questions remain unanswered: Why do agents cooperate rather than act individually? What environmental factors encourage or hinder collaboration? What mechanisms ensure the balance between personal gain and collective success? [15] These challenges in understanding cooperation have prompted spirited debates, reminiscent of Charles Darwin’s own struggles to explain selflessness in social insects, which he once admitted posed a serious challenge to his theory of natural selection. Recent research

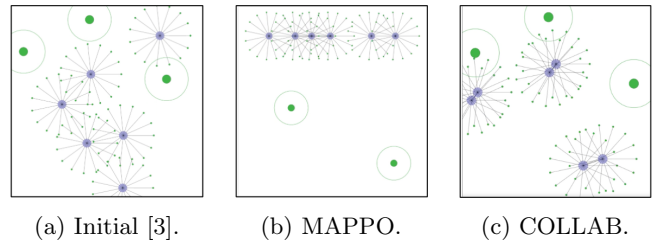


Figure 1: Comparison of agent behaviors in the discovery task: (a) Initial random movement, (b) MAPPO-trained agents exhibit homogeneous behavior, and (c) COLLAB-trained (ours) agents exhibiting cooperative behavior through coalition formation.

suggests that rather than cooperation simply emerging from intelligent agents, intelligence itself is shaped within environments characterized by these dynamics. In such settings, competition fosters innovation, while social interaction and cooperation are crucial for the emergence of intelligence [2, 6].


The rapid advancements in artificial intelligence, particularly in multi-agent reinforcement learning (MARL), offer fresh perspectives on the long-standing debate surrounding cooperation. Existing MARL approaches focus on improving reward assignment [7], value function approximation [16], and policy optimization [11, 14]. Despite the progress made in multi-agent reinforcement learning, current state-of-the-art methods, such as QMIX [16] and MADDPG [14], have several limitations when applied to cooperative tasks. These methods often exhibit only marginal improvements over basic approaches like MAPPO [24] and suffer from poor performance, as shown in the (b) of figure 1. Although they enhance policy optimization or value function approximation techniques, they do not effectively differentiate between agents’ membership in cooperative subgroups, which is crucial for modeling collaborative behavior. Another significant drawback is the lack of a subjective method to detect the presence of cooperative actions within learned policies, making it difficult to interpret the emerging cooperative strategies.

In this paper, we take initial steps toward addressing these limitations using the **COaLition LABELing**

*State Key Laboratory of Software Development Environment, Beihang University, Beijing, China. {sudingli, zengguangjie, yichengp, angsheng}@buaa.edu.cn

[†]School of Cyber Science and Technology, Beihang University. penghao@buaa.edu.cn

[‡]Faculty of Information Engineering and Automation, Kunming University of Science and Technology. lip@stu.kust.edu.cn

 Corresponding author.

(**COLLAB**) technique, which augments MAPPO and improves their performance in cooperative tasks, as shown in figure 1(c). Our approach is systematic: first, we select a set of multi-agent benchmark tasks that necessitate cooperation; next, we design a framework to improve the cooperative learning algorithms applied to these tasks; and finally, we analyze whether the agents successfully learn cooperative behavior by applying structural entropy measurements, which help us detect the emergence of coalition structures.

We focus on fully cooperative, partially observable, multi-agent tasks with shared rewards, requiring agents to coordinate under communication constraints. These tasks often follow a divide-and-conquer approach, necessitating altruistic strategies for a common objective that individual agents can't achieve alone. Our approach involves centralized training with decentralized execution (CTDE), where communication is unrestricted during training but limited during execution, as commonly used in multi-agent planning [7].

Inspired by graph representation learning, particularly the Labeling Trick [31], we propose the **COLLAB** technique to enhance the representation of agent coalitions. **COLLAB** assigns labels to agents' observations, transforming them into enriched inputs for policy learning. Our experiments show that simple aggregation of observations fails to capture complex structures, echoing findings from recent graph representation studies like GLASS [21]. To the best of our knowledge, this is the first application of coalition labeling in MARL.

Structural entropy is a measure that quantifies the complexity of dynamic interactions in networks and has been successfully applied in community detection [13, 18]. Since cooperation is a type of dynamic behavior in social systems and multi-agent systems (MAS), structural entropy is well-suited for identifying emergent cooperation and coalition structures. By measuring the uncertainty in agents' interactions, structural entropy reveals how well agents coordinate to achieve shared goals. This approach builds a graph from agents' interactions across environment frames, offering a dynamic representation of cooperation and capturing the complexity of these coordinated behaviors.

Our experimental results demonstrate the effectiveness of **COLLAB-MARL** on a well-established benchmark task, underscoring the novelty of utilizing the **COLLAB** technique in reinforcement learning to facilitate the learning of cooperative behaviors in a more explainable and expressive manner. Furthermore, we introduce the innovative use of structural entropy as a tool to detect and quantify cooperative behaviors. **COLLAB-MARL** achieves either the best or near-best scores across multiple tasks, demonstrating an 8.4% av-

erage improvement over the strongest competing baseline in terms of mean return. Additionally, our approach exhibits greater training efficiency when compared to existing methods. The experimental outcomes also validate the utility of structural entropy as a robust metric for evaluating the emergence of coalitions and cooperation in multi-agent systems, providing a systematic and reliable framework for understanding cooperative dynamics.

2 Related Work

Our work builds upon key research areas, including MARL, and coalition formation. Below, we provide an overview of relevant literature and identify gaps our approach addresses.

2.1 Multi-Agent Reinforcement Learning.

MARL has become a robust framework for coordination in multi-agent systems [29]. MARL algorithms are often categorized as centralized training with decentralized execution (CTDE) or fully decentralized approaches. CTDE approaches like MAPPO [24], MADDPG [14], and QMIX [16] use a central controller during training, while execution is decentralized. In contrast, fully decentralized approaches, such as Independent Q-Learning [20] and Decentralized Actor-Critic [30], rely solely on local observations.

2.2 Coalition Formation with Structural Entropy.

Coalition formation is a key aspect of multi-agent coordination that enables agents to form temporary teams to tackle specific objectives more effectively [10]. Research in multi-agent systems has focused on developing algorithms for forming stable coalitions, often relying on game-theoretic concepts like the core or the Shapley value [5]. However, in swarm robotics and multi-agent reinforcement learning (MARL), the explicit integration of coalition formation has been largely unexplored.

Jiang and Lu [11] introduced attentional communication mechanisms for multi-agent cooperation, implicitly allowing sub-group formation within a swarm. However, these approaches do not explicitly address task allocation or leverage structural entropy to enhance coalition formation.

Structural entropy, derived from structural information theory [12], measures the complexity of a system by quantifying uncertainty in its structure [13]. In multi-agent systems, structural entropy can be used to optimize coalition formation by minimizing structural uncertainty [26] and maximizing task efficiency [27, 28]. Despite its success in other areas, the use of structural entropy in swarm robotics is less explored.

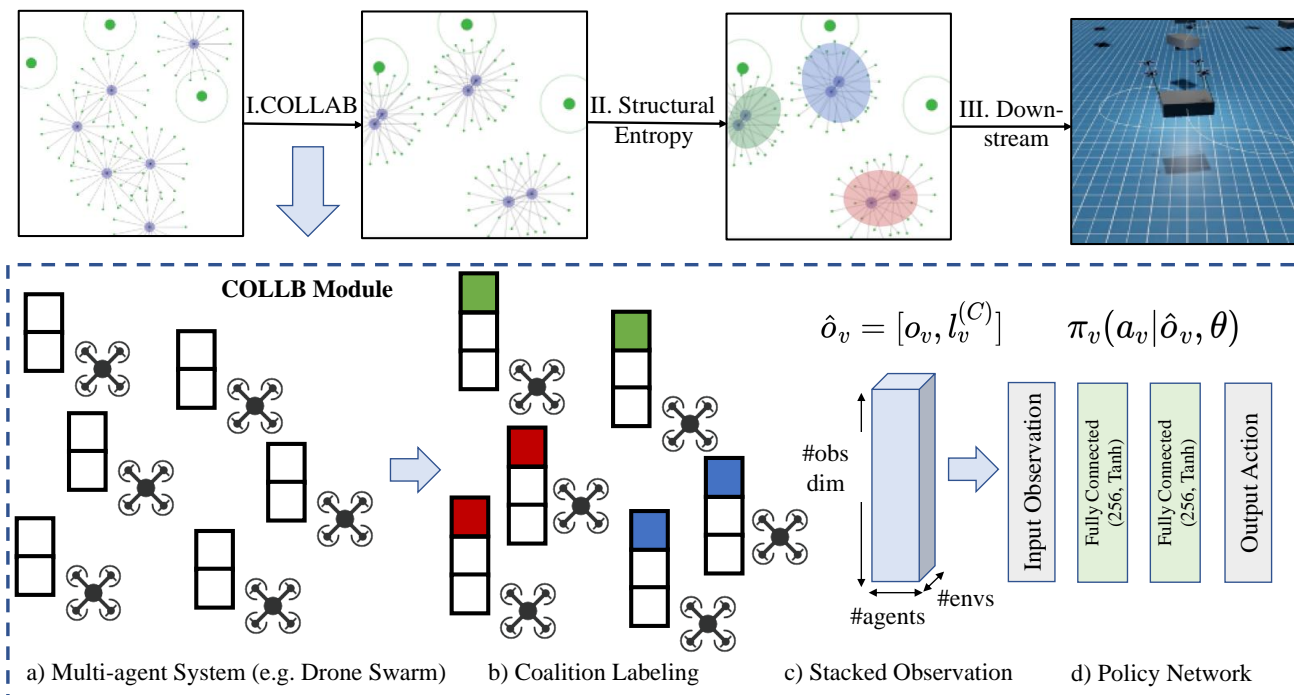


Figure 2: Overview of the proposed method. Step **I. COLLAB**: Apply coalition labeling. **II. Structural Entropy** captures interaction complexity. **III. Down-stream** tasks use coalition labels to optimize cooperative performance. The **COLLAB Module** includes: (a) **Multi-agent System** with drone swarms as an example (b) **Coalition Labeling**, (c) **Stacked Observation**, and (d) **Policy Network** mapping observations to actions.

Our approach aims to bridge these gaps by combining MARL with coalition formation for task assignment in multi-agent systems like drone swarms, leveraging structural entropy. This enables the swarm to adapt coalition structures and task allocation strategies in a decentralized manner, balancing structural uncertainty and coalition efficiency. This integration addresses the limitations of existing approaches, providing a novel framework for coalition formation and task allocation based on structural entropy principles.

3 Methodology

This section presents two complementary components of our approach (Figure 2). First, the COLLAB-MARL framework enhances the baseline MARL algorithm by incorporating coalition labeling, enriching agents' observation space with coalition-specific meta-information, and allowing strategy adjustments. Second, structural entropy quantifies coordination among agents by analyzing interaction graphs over time, providing a clear measure of collaboration effectiveness.

3.1 COLLAB-MARL Framework. The COLLAB-MARL framework extends the existing Multi-Agent Proximal Policy Optimization (MAPPO) algorithm by augmenting agent observations with coalition information. The primary aim of this framework is to enhance coordination and cooperation among agents belonging to the same coalition in environments. By incorporating coalition-specific information into each agent's state representation, COLLAB-MARL enables more sophisticated intra-coalition collaboration, leading to improved collective performance.

3.1.1 Coalition Labeling Technique. In a multi-agent system (MAS), agents operate within the same environment, sharing a common policy network in scenarios where cooperation between subsets of agents (coalitions) is necessary for optimizing heterogeneous reward functions. However, agents cannot inherently perceive their coalition membership from their environmental observations, as coalition structure is typically not embedded in the environment itself. To address this issue, we introduce the coalition labeling technique, which appends a coalition-specific identifier to each agent's ob-

servation vector.

Definition 1. Given a multi-agent system $S = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ represents the set of agents and E denotes the set of agent interactions, let $C = \{C_1, C_2, \dots, C_k\}$ be the set of coalitions within S . Each agent $v \in V$ belongs to one coalition $C_i \in C$. The integer coalition label of agent v is defined as:

$$(3.1) \quad l_v^{(C)} = \begin{cases} i & \text{if } v \in C_i. \end{cases}$$

The scalar label $l_v^{(C)}$ indicates the coalition to which agent v belongs and is appended directly to the agent’s observation vector o_v , transforming it into an augmented observation $\hat{o}_v = [o_v, l_v^{(C)}]$. By including $l_v^{(C)}$ in the agent’s observation, the policy network can recognize coalition membership as part of the agent’s state representation, enabling the shared policy to learn behaviors specific to each coalition and improve coordination among agents within the same coalition. This approach eliminates the need for agents to perceive coalition membership through environmental cues, thus avoiding modifications to the environment or sensor model. Mathematically, the policy $\pi_v(a_v | \hat{o}_v, \theta)$ for each agent v is conditioned on the augmented observation \hat{o}_v , where a_v represents the action taken by agent v , and θ represents the shared policy parameters.

The coalition labeling technique is simple to implement, requiring only a scalar label to be appended to the observation vector, without necessitating any changes to the policy architecture. Since it integrates easily into most MARL frameworks, especially those using shared policies like MAPPO, it is highly compatible with existing setups. Additionally, the label provides a form of **status awareness**, allowing agents to differentiate coalition membership, which would otherwise be difficult for the policy network to infer directly from raw environmental data. This awareness fosters more strategic and cooperative behavior, enhancing coordination within the coalition.

3.2 Policy Framework. Our policy framework is compatible with various baseline methods, such as MAPPO, MADDPG, QMIX, VDN, and MASAC. Depending on the specific algorithm, different architectures (e.g., shared or individual policies) may be used to suit each method. The shared network processes an augmented observation \hat{o}_v , which combines the original observation o_v with coalition information. The policy maps \hat{o}_v to a distribution over actions, parameterized by mean μ_v and variance σ_v :

$$(3.2) \quad \pi_v(a_v | \hat{o}_v, \theta) = \mathcal{N}(\mu_v(\hat{o}_v), \sigma_v(\hat{o}_v)),$$

where θ represents the shared parameters. The policy network is implemented as a multi-layer perceptron (MLP) with L hidden layers:

$$(3.3) \quad h_v^{(l)} = \phi(W^{(l)}h_v^{(l-1)} + b^{(l)}).$$

The value function is modeled by a critic network, which can be centralized or decentralized. For actor-critic methods (e.g., MAPPO, MADDPG), a centralized critic is used, while value-based methods (e.g., QMIX, VDN) use decentralized critics:

$$(3.4) \quad V(\hat{o}_v) = f(W_c, \hat{o}_v).$$

The Training uses collected batches of observations, actions and rewards. For actor-critic methods, Proximal Policy Optimization (PPO) is applied, with Generalized Advantage Estimation (GAE) used for advantage calculation. Value-based methods like QMIX and VDN minimize tailored loss functions to improve value predictions. This framework enables coordinated learning through shared parameters and supports multiple multi-agent reinforcement learning algorithms, enhancing scalability and generalizability in cooperative tasks.

3.3 Cooperation Detection Using Structural Entropy

3.3.1 Graph Building. To detect cooperation among agents in a multi-agent system, we construct a dynamic graph $G = (V, E, W)$, where V represents the set of agents, E denotes the cooperative interactions, and W quantifies the strength of these interactions. Each agent $v_i \in V$ is represented as a node, and an edge $(v_i, v_j) \in E$ exists between agents v_i and v_j if they demonstrate cooperative behavior.

We consider a scenario where agents’ full dynamics are described by their positions \mathbf{x}_i , velocities $\dot{\mathbf{x}}_i$, and accelerations $\ddot{\mathbf{x}}_i$. The cooperation weight $W(v_i, v_j)$ is defined based on the spatial distance $|\mathbf{x}_i - \mathbf{x}_j|$, velocity difference $|\dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j|$, and acceleration difference $|\ddot{\mathbf{x}}_i - \ddot{\mathbf{x}}_j|$ between agents v_i and v_j . The cooperation weight $W(v_i, v_j)$ is given by:

$$(3.5) \quad W(v_i, v_j) = e^{-\alpha(|\mathbf{x}_i - \mathbf{x}_j|^2 + \beta|\dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j|^2 + \gamma|\ddot{\mathbf{x}}_i - \ddot{\mathbf{x}}_j|^2)},$$

where α , β , and γ are hyper-parameters controlling the influence of each factor. Alternatively, it can be expressed as:

$$(3.6) \quad W(v_i, v_j) = \frac{1}{|\mathbf{x}_i - \mathbf{x}_j| + \beta|\dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j| + \gamma|\ddot{\mathbf{x}}_i - \ddot{\mathbf{x}}_j| + \epsilon},$$

where ϵ is a small constant to avoid division by zero.

To avoid constructing a dense and potentially noisy graph G_t at each time step, we sparsify the graph using a threshold hyperparameter τ . Specifically, an edge (v_i, v_j, t) is included in the graph G_t if the cooperation weight $W(v_i, v_j, t)$, computed based on the agents' positions \mathbf{x}_i , velocities $\dot{\mathbf{x}}_i$, and accelerations $\ddot{\mathbf{x}}_i$, exceeds this threshold. Formally, the edge is included if $W(v_i, v_j, t) > \tau$.

3.3.2 Structural Entropy and Encoding Tree Enumeration. To analyze the complexity and regularity of cooperation patterns within the dynamic graph G constructed in the previous section, we introduce the concept of structural entropy. The structural entropy quantifies the information content of the agent interaction network, providing insights into the underlying community structure. To minimize this entropy, we construct an encoding tree that hierarchically organizes agents based on their interactions, optimizing the tree to reduce the description length of the interaction patterns.

Enumerating possible encoding tree structures for the dynamic graph G is closely related to **Schröder's Fourth Problem** [17], which enumerates series-reduced rooted trees, also known as number of total partitions. In our context, each partitioning corresponds to a recursive clustering of agents, equivalent to the encoding tree structure. The total number of hierarchical partitionings follows a recurrence relation from Schröder's Fourth Problem:

$$(3.7) \quad a(n+1) = (n+2) \cdot a(n) + 2 \cdot \sum_{k=2}^{n-1} \binom{n}{k} \cdot a(k) \cdot a(n-k+1).$$

The recurrence relation from Schröder's Fourth Problem, with $a(n)$ denoting the number of hierarchical partitionings for n agents, captures the recursive subdivision of subsets. For multi-agent systems with up to 6 agents, we enumerate all possible encoding trees, each representing a distinct hierarchical clustering based on the cooperation weights $W(i, j, t)$ at time step t . The structural entropy, calculated by summing the local entropy at each tree node determined by the interaction weights, is minimized to obtain the optimal tree, revealing the most compact and informative representation of the interaction dynamics.

4 Experiments

In this section, we compare COLLAB with state-of-the-art multi-agent reinforcement learning methods, specifically IPPO, MAPPO, MADDPG, IDDPG, VDN, QMIX, and MASAC, across 2D physical environments and additional demonstration tasks. The aim is to

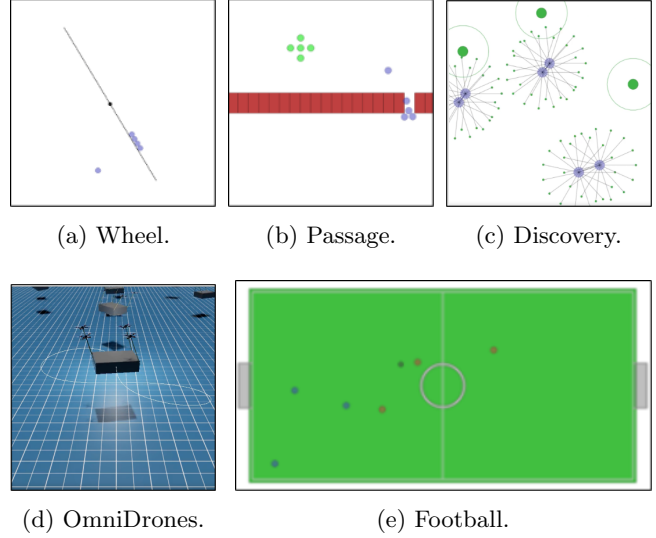


Figure 3: Images illustrating different experimental tasks. The first row shows (a) the Wheel task, (b) the Passage task, and (c) the Discovery task. The second row presents (d) the OmniDrones platform and (e) the Football task.

demonstrate that COLLAB achieves superior performance and fosters a higher degree of cooperation, while showcasing its adaptability to various cooperative scenarios.

4.1 Experimental Settings.

4.1.1 Testbeds. We use four 2D physics-based tasks (Balance, Discovery, Passage, and Wheel) from the VMAS benchmark suite [3] as shown in figure 3. These tasks evaluate various cooperative behaviors, including target coverage, obstacle navigation, and synchronized control. Additionally, we use two other tasks for demonstration purposes: a competitive Football task and a 3D drone navigation task in the OmniDrones platform [22]. The Football task is used to showcase competitive behaviors, while the OmniDrones platform demonstrates the adaptability of our methods in a realistic simulation environment.

4.1.2 Baselines. We compare our approach against seven multi-agent reinforcement learning baselines: IPPO, MAPPO [25], MADDPG [14], IDDPG, VDN [19], QMIX [16], and MASAC [9]. IPPO and MAPPO utilize centralized value functions for cooperative task optimization, with MAPPO having additional centralized components. MADDPG and IDDPG are designed for continuous action spaces, with decentralized critics. VDN and QMIX factorize joint action-value func-

tions to facilitate coordinated task assignment, with QMIX employing a more complex factorization structure. MASAC handles exploration-exploitation in continuous action settings effectively with a centralized critic during training.

4.1.3 Evaluation Metric. We evaluate using metrics from **MARL-eval** [8] and **rliable** [1] for cooperative multi-agent reinforcement learning (MARL): **S1 (Median)**: The middle value of performance, showing typical task results. **S2 (IQM)**: The Interquartile Mean, calculating the average of the middle 50% of data to reduce outlier impact and reflect typical performance. **S3 (Mean)**: The arithmetic mean, representing overall average performance. **S4 (Optimality Gap)**: The gap between optimal return and observed return, measuring how close a policy is to ideal performance.

Sample Efficiency is also plotted to illustrate how efficiently each algorithm learns over the training data, with curves plotted against the number of timesteps to provide insights into the learning progress over time. Additionally, we employ **Structural Entropy** to measure the cooperative dynamics of the complex multi-agent system, where entropy progression over steps indicates the system’s global dynamics. This suite of metrics ensures a robust and standardized comparison of the evaluated MARL algorithms.

4.2 Main Result: Overall Performance Analysis. Our COLLAB approach leads to significant performance improvements across all evaluated reinforcement learning (RL) methods, as reflected in both Table1 and the sample efficiency curves depicted in Figure4. The evaluated methods include IPPO, MAPPO, MADDPG, IDDPG, VDN, QMIX, and MASAC, and their respective COLLAB-enhanced versions.

As shown in Table1, the COLLAB-enhanced versions of QMIX (C-QMIX), MAPPO (C-MAPPO), and IPPO (C-IPPO) achieved either the best or near-best scores across multiple tasks. Specifically, the COLLAB variations consistently exhibited high median (S1), interquartile mean (IQM, S2), and mean (S3) scores, as well as lower Optimality Gap (S4) across the four benchmark tasks: BALANCE, DISCOVERY, PASSAGE, and WHEEL. For instance, C-QMIX attained the highest performance in the WHEEL task, with a perfect score across S1, S2, and S3, while also minimizing the Optimality Gap to zero, thus demonstrating its superior convergence. Similarly, C-IPPO and C-MAPPO performed notably well, achieving competitive results, with consistently highlighted improvements over their original versions.

Figure4 provides further insight into the sample ef-

iciency of these algorithms. The COLLAB-enhanced versions, particularly C-QMIX, C-MAPPO, and C-IPPO, consistently showed superior learning progress, with their curves positioned significantly above those of the other algorithms. This indicates that these COLLAB-enhanced methods achieved faster convergence, requiring fewer samples to reach high performance levels. The significant separation of the COLLAB curves from the baseline suggests that our approach facilitates quicker learning and a more efficient use of resources, which is crucial for complex multi-agent tasks.

The consistent reduction in Optimality Gap (S4) across all tasks demonstrates that the COLLAB method not only improves performance metrics but also reduces variability, leading to more reliable outcomes. Overall, our COLLAB method enhances subgroup cooperation and improves the robustness of the system in solving cooperative tasks effectively.

4.3 Visualizing Cooperation with Structural Entropy. Figure5 illustrates the utility of our Structural Entropy metric in visualizing and interpreting the dynamics of cooperation within multi-agent systems. The entropy progression over time captures the systems global evolution, reflecting key stages of agent organization and community formation. Initially, the rapid increase in entropy indicates a phase of exploratory interactions among agents as they begin to form coalitions. As time progresses, the plateau in entropy signifies a stabilization in the system, where distinct and persistent subgroups or communities emerge.

The snapshots in the figure provide additional clarity by visually depicting the agent community structures at different time steps. Early in the process, agents exhibit sparse and weakly defined connections, whereas later stages reveal well-established, tightly connected communities. These visualizations not only validate our Structural Entropy metric as an effective tool for capturing the emergence of cooperative subgroups but also demonstrate the successful implementation of our labeling method, which facilitates both interpretability and the identification of stable agent coalitions.

These results underscore the dual advantages of our approach: optimizing agent performance through efficient subgroup formation, while simultaneously offering an interpretable understanding of how cooperation and community dynamics evolve in complex environments.

4.4 Ablation Study: Impact of Observation Labeling and Aggregation on Reward Performance. Observation labeling (Lbl.) and observation aggregation (Aggr.) play a significant role in influenc-

Table 1: Performance comparison of different RL methods across various tasks. The table combines overall performance comparisons between baseline and their variations. The best results are highlighted in **bold**, and the second-best results are underlined. Metrics include S1 (Median), S2 (IQM), S3 (Mean), and S4 (Optimality Gap) over all tasks.

RL Method	BALANCE				DISCOVERY				PASSAGE				WHEEL				COMBINED
	S1 (†)	S2 (†)	S3 (†)	S4 (‡)	S1 (†)	S2 (†)	S3 (†)	S4 (‡)	S1 (†)	S2 (†)	S3 (†)	S4 (‡)	S1 (†)	S2 (†)	S3 (†)	S4 (‡)	
IPPO	0.88	0.88	0.88	0.12	0.45	0.45	0.45	0.55	<u>0.99</u>	<u>0.99</u>	<u>0.99</u>	<u>0.01</u>	1.00	1.00	1.00	0.00	0.94 / <u>0.94</u> / 0.83 / 0.17
MAPPO	0.93	0.93	0.93	0.07	0.47	0.47	0.47	0.53	<u>0.99</u>	<u>0.99</u>	<u>0.99</u>	<u>0.01</u>	0.76	0.76	0.76	0.24	0.84 / 0.84 / 0.79 / 0.21
MADDPG	0.91	0.91	0.91	0.09	0.35	0.35	0.35	0.65	<u>0.99</u>	<u>0.99</u>	<u>0.99</u>	<u>0.01</u>	0.73	0.73	0.73	0.27	0.82 / 0.82 / 0.75 / 0.25
IDDPG	0.83	0.83	0.83	0.17	<u>0.80</u>	<u>0.80</u>	<u>0.80</u>	<u>0.20</u>	0.98	0.98	0.98	0.02	0.74	0.74	0.74	0.26	0.82 / 0.84 / <u>0.84</u> / <u>0.16</u>
VDN	0.86	0.86	0.86	0.14	0.77	0.77	0.77	0.23	1.00	1.00	1.00	0.00	0.74	0.74	0.74	0.26	0.82 / 0.84 / <u>0.84</u> / <u>0.16</u>
QMIX	0.56	0.56	0.56	0.44	0.85	0.85	0.85	0.15	<u>0.99</u>	<u>0.99</u>	<u>0.99</u>	<u>0.01</u>	<u>0.92</u>	<u>0.92</u>	<u>0.92</u>	<u>0.08</u>	0.89 / 0.96 / 0.83 / 0.17
MASAC	0.89	0.89	0.89	0.11	0.45	0.45	0.45	0.55	1.00	1.00	1.00	0.00	0.70	0.70	0.70	0.30	0.80 / 0.80 / 0.76 / 0.24
C-MADDPG	0.96	0.96	0.96	0.04	0.40	0.40	0.40	0.60	<u>0.99</u>	<u>0.99</u>	<u>0.99</u>	<u>0.01</u>	0.75	0.75	0.75	0.25	0.86 / 0.86 / 0.78 / 0.22
C-IDDPG	0.83	0.83	0.83	0.17	0.55	0.55	0.55	0.45	<u>0.99</u>	<u>0.99</u>	<u>0.99</u>	<u>0.01</u>	0.78	0.78	0.78	0.22	0.80 / 0.80 / 0.79 / 0.21
C-VDN	0.83	0.83	0.83	0.17	0.75	0.75	0.75	0.25	1.00	1.00	1.00	0.00	0.72	0.72	0.72	0.28	0.79 / 0.80 / 0.82 / 0.18
C-QMIX	0.87	0.87	0.87	0.13	0.72	0.72	0.72	0.28	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	<u>0.93</u> / 0.93 / 0.90 / 0.10
C-MASAC	<u>0.94</u>	<u>0.94</u>	<u>0.94</u>	<u>0.06</u>	0.32	0.32	0.32	0.68	1.00	1.00	1.00	0.00	0.81	0.81	0.81	0.19	0.87 / 0.87 / 0.77 / 0.23

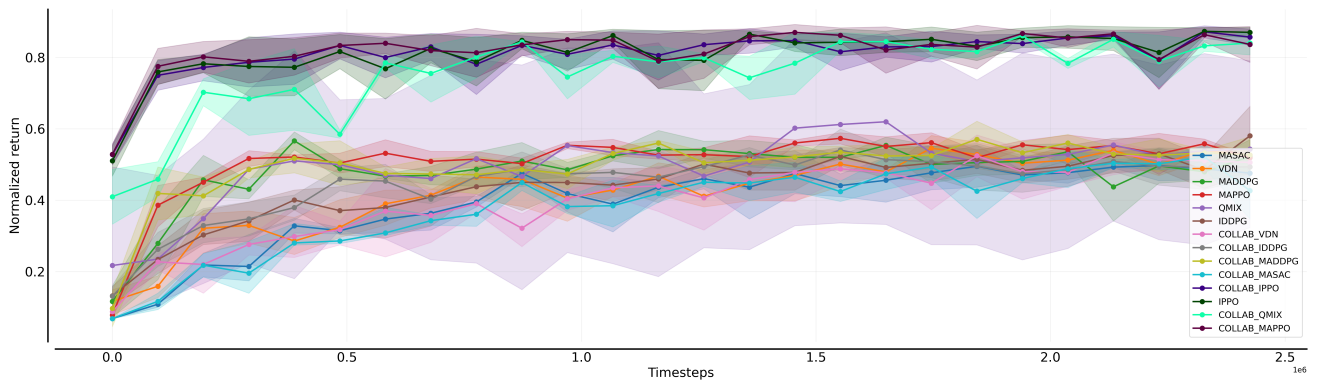


Figure 4: Sample Efficiency curves for all algorithms across all tasks. Algorithm names with the prefix **COLLAB**-indicate enhancements by our **COLLAB** technique. Note that **COLLAB-QMIX**, **COLLAB-MAPPO**, and **COLLAB-IPPO** are among the top four curves.

Table 2: Impact of Observation Labeling and Aggregation on Evaluation Performance. Abbreviations: Lbl - Observation Labeling, Aggr. - Observation Aggregation.

Lbl	Aggr.	MR (x1000)	Iter T (s)
✓	✓	3.09	5.22
×	✓	3.09	7.88
✓	×	3.14	7.82
×	×	2.97	6.91

ing reward performance and computational efficiency in multi-agent systems. Observation labeling involves adding a common label for agents within the same coalition, allowing them to identify their coalition membership, while observation aggregation extends each agent’s observation by appending an aggregated observation vector derived from all members within the same coalition.

Our results align with previous findings that aggregating node representations alone to capture high-order structures may not be expressive enough, and that using labeling can effectively address this limitation [21]. Specifically, the combination of observation labeling and aggregation yielded the highest performance improvement in terms of reward, as indicated in Table 2. Moreover, observation labeling also demonstrated improvements in the interpretability of agent behaviors, as it provided a clearer indication of coalition membership and dynamics. The results further suggest that labeling is a critical factor in enhancing both performance and explainability in cooperative multi-agent systems.

4.5 Computational Efficiency Analysis. Figure 4 presents iteration time comparisons for various algorithms, broken down into collection, evaluation, and training phases. Algorithms with observation labels are positioned alongside unlabeled variants for direct comparison. In general, labeled versions have higher iteration

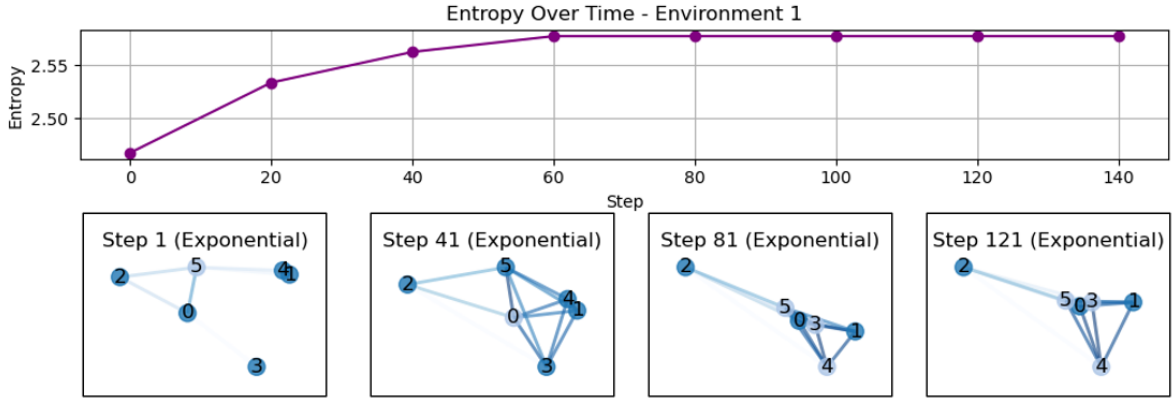


Figure 5: Experiment results demonstrating the evolution of agent communities over time. The top plot shows the entropy progression over steps, indicating the system’s global dynamics. Below, four snapshots capture the community structure of agents at different steps, with node colors representing community memberships. Each snapshot corresponds to a key stage in the evolution process, illustrating how agent communities form and evolve.

ation times, primarily due to increased training time, while collection and evaluation differences are minimal. The bar chart highlights computational bottlenecks, providing insights for algorithmic efficiency.

enumeration feasible only for small n (e.g., $n \leq 10$). The overall time complexity, dominated by $O(n^n)$, underscores the necessity of limiting the number of agents to maintain computational feasibility.

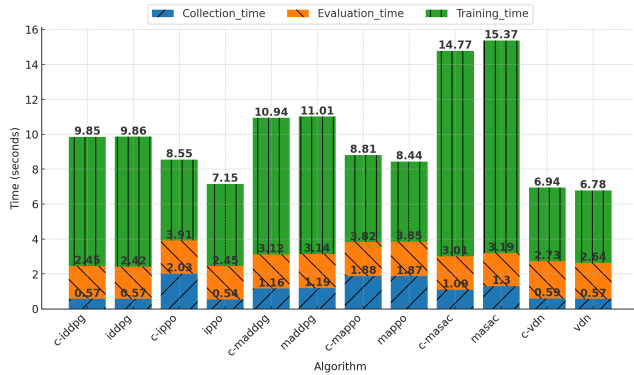


Figure 6: Breakdown of Iteration Time by Algorithm with Collection, Evaluation, and Training Components

The structural entropy and encoding tree enumeration phase is the most computationally intensive aspect of the algorithm. Enumerating all possible encoding tree structures for a dynamic graph with n agents involves generating all hierarchical partitions, corresponding to series-reduced rooted trees. The number of such encoding trees grows super-exponentially, approximately as:

$$(4.8) \quad a(n) \sim \frac{n^{n-1}}{\sqrt{2} \cdot e \cdot (2 \ln 2 - 1)^{n-1/2}}$$

This growth follows an order of $O(n^n)$, making

5 Conclusion

We introduce COLLAB, an innovative and effective model for multi-agent cooperation learning. Our study demonstrates the superiority of the coalition labeling technique over traditional MARL algorithms and highlights the efficacy of structural entropy as a metric for detecting cooperative behavior. COLLAB consistently achieves either the best or near-best performance across multiple tasks, surpassing previous state-of-the-art methods by an average of 8.4%. Our analysis reveals that structural entropy strongly correlates with overall cooperation performance, and we validate its effectiveness as a cooperation indicator. While our coalition labeling technique has proven successful, it currently relies on manually set labels, which may not be feasible in real-world cooperative learning scenarios. Future research could explore more advanced techniques, like dynamic graph modeling [23] or multi-relational cooperative interaction [4], to enable the adaptive learning of these labels.

Acknowledgments

This work is supported by the NSFC through grants 62441612, 62322202, 62432006, and 62476163, and the Guangdong Basic and Applied Basic Research Foundation through grant 2023B1515120020, and Hebei Natural Science Foundation through grant F2024210008.

References

- [1] R. AGARWAL, M. SCHWARZER, P. S. CASTRO, A. COURVILLE, AND M. G. BELLEMARE, *Deep reinforcement learning at the edge of the statistical precipice*, Advances in Neural Information Processing Systems, (2021).
- [2] B. BAKER, I. KANITSCHIEDER, T. M. MARKOV, Y. WU, G. POWELL, B. MCGREW, AND I. MORDATCH, *Emergent tool use from multi-agent autocurricula*, (2020).
- [3] M. BETTINI, R. KORTVELESY, J. BLUMENKAMP, AND A. PROROK, *Vmas: A vectorized multi-agent simulator for collective robot learning*, International Symposium on Distributed Autonomous Robotic Systems, (2022).
- [4] Y. CAO, H. PENG, A. LI, C. YOU, Z. HAO, AND S. Y. PHILIP, *Multi-relational structural entropy*, in The 40th Conference on Uncertainty in Artificial Intelligence.
- [5] G. CHALKIADAKIS, E. ELKIND, AND M. WOOLDRIDGE, *Cooperative game theory: Basic concepts and computational challenges*, IEEE Intelligent Systems, (2012).
- [6] E. A. DUÉÑEZ-GUZMÁN, S. SADEDIN, J. X. WANG, K. R. MCKEE, AND J. Z. LEIBO, *A social path to human-like artificial intelligence*, Nat. Mac. Intell., (2023).
- [7] J. FOERSTER, G. FARQUHAR, T. AFOURAS, N. NARDELLI, AND S. WHITESON, *Counterfactual multi-agent policy gradients*, (2018).
- [8] R. GORSANE, O. MAHJOUR, R. J. DE KOCK, R. DUBB, S. SINGH, AND A. PRETORIUS, *Towards a standardised performance evaluation protocol for cooperative marl*, Advances in Neural Information Processing Systems, (2022).
- [9] T. HAARNOJA, A. ZHOU, P. ABBEEL, AND S. LEVINE, *Soft actor-critic algorithms and applications*, arXiv preprint arXiv:1812.05905, (2018).
- [10] B. HORLING AND V. LESSER, *A survey of multi-agent organizational paradigms*, The Knowledge Engineering Review, (2004).
- [11] J. JIANG AND Z. LU, *Learning attentional communication for multi-agent cooperation*, Advances in Neural Information Processing Systems, (2018).
- [12] A. LI, *Science of Artificial Intelligence: Mathematical Principles of Intelligence* (In Chinese), Science Press, Beijing, 2024.
- [13] A. LI AND Y. PAN, *Structural information and dynamical complexity of networks*, IEEE Trans. Inf. Theory, (2016).
- [14] R. LOWE, Y. WU, A. TAMAR, J. HARB, P. ABBEEL, AND I. MORDATCH, *Multi-agent actor-critic for mixed cooperative-competitive environments*, Advances in neural information processing systems, (2017).
- [15] E. PENNISI, *On the origin of cooperation*, (2009).
- [16] T. RASHID, M. SAMVELYAN, C. DE WITT, G. FARQUHAR, J. FOERSTER, AND S. WHITESON, *Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning*, (2018).
- [17] J. RIORDAN, *The blossoming of schröder's fourth problem*, Acta Mathematica, (1976).
- [18] M. ROSVALL AND C. T. BERGSTROM, *An information-theoretic framework for resolving community structure in complex networks*, Proceedings of the National Academy of Sciences, (2007).
- [19] P. SUNEHAG, G. LEVER, A. GRUSLYS, W. M. CZARNECKI, V. ZAMBALDI, M. JADERBERG, M. LANCTOT, N. SONNERAT, J. Z. LEIBO, K. TUYLS, ET AL., *Value-decomposition networks for cooperative multi-agent learning*, arXiv preprint arXiv:1706.05296, (2017).
- [20] M. TAN, *Multi-agent reinforcement learning: Independent vs. cooperative agents*, Proceedings of the tenth international conference on machine learning, (1993).
- [21] X. WANG AND M. ZHANG, *GLASS: GNN with labeling tricks for subgraph representation learning*, The Tenth International Conference on Learning Representations, ICLR, (2022).
- [22] B. XU, F. GAO, C. YU, R. ZHANG, Y. WU, AND Y. WANG, *Omnidrones: An efficient and flexible platform for reinforcement learning in drone control*, IEEE Robotics and Automation Letters, (2024).
- [23] R. YANG, H. PENG, C. LIU, AND A. LI, *Incremental measurement of structural entropy for dynamic graphs*, Artificial Intelligence, 334 (2024), p. 104175.
- [24] C. YU, A. VELU, E. VINITSKY, J. GAO, Y. WANG, A. BAYEN, AND Y. WU, *The surprising effectiveness of ppo in cooperative multi-agent games*, Advances in Neural Information Processing Systems, (2022).
- [25] C.-H. YU, P. MACALPINE, J. HANNA, AND P. STONE, *The surprising effectiveness of mappo in cooperative multi-agent games*, arXiv preprint arXiv:2103.01955, (2021).
- [26] X. ZENG, H. PENG, AND A. LI, *Effective and stable role-based multi-agent collaboration by structural information principles*, Thirty-Seventh AAAI Conference on Artificial Intelligence, (2023).
- [27] X. ZENG, H. PENG, AND A. LI, *Adversarial social-bots modeling based on structural information principles*, Thirty-Eighth AAAI Conference on Artificial Intelligence, (2024).
- [28] X. ZENG, H. PENG, AND A. LI, *Effective exploration based on the structural information principles*, in Proceedings of the 38th Conference on Neural Information Processing Systems, 2024.
- [29] K. ZHANG, Z. YANG, AND T. BAŞAR, *Multi-agent reinforcement learning: A selective overview of theories and algorithms*, arXiv preprint arXiv:1911.10635, (2021).
- [30] K. ZHANG, Z. YANG, H. LIU, T. ZHANG, AND T. BAŞAR, *Fully decentralized multi-agent reinforcement learning with networked agents*, International Conference on Machine Learning, (2018).
- [31] M. ZHANG, P. LI, Y. XIA, K. WANG, AND L. JIN, *Labeling trick: A theory of using graph neural networks for multi-node representation learning*, (2021).