GEORGE BROWN PRESS
JOURNALS, BOOKS & PROCEEDINGS

Journal of Computer, Signal, and System Research

Article

Research on Q-Learning and Motion Control of Multi-Robot Systems Based on Community-Aware Networks

Kexuan Shen [1,*]

[1] Texas A&M University, College Station, Texas, USA
* Correspondence: Kexuan Shen, Texas A&M University, College Station, Texas, USA

Abstract: Collaborative perception and autonomous motion control in complex environments are key challenges for multi-robot systems. As system scale increases, traditional centralized control and conventional distributed learning methods suffer from high communication overhead, low learning efficiency, and reduced stability. To address these limitations, this paper introduces a community-aware network framework that partitions the communication topology of a multi-robot system into communities with strong internal interactions. Based on this structure, a community-aware Q-learning and motion control method is proposed. The communication relationships among robots are modeled from the perspective of complex networks, and a community-based state representation is designed to capture local cooperation while reducing state dimensionality. Community-level information is further incorporated into the Q-value update through an improved reward mechanism, enhancing learning efficiency and convergence stability. In addition, learning decisions are mapped to continuous control inputs using robot kinematic models, enabling coordinated motion within communities and effective obstacle avoidance between communities. Simulation results demonstrate that the proposed method outperforms traditional multi-robot Q-learning approaches in convergence speed, path efficiency, and overall cooperative performance.

Keywords: multi-robot systems; community-aware networks; reinforcement learning; Q-learning; motion control

Received: 22 December 2025
Revised: 01 February 2026
Accepted: 11 February 2026
Published: 15 February 2026

1. Introduction

Multi-robot systems have attracted increasing attention in recent years due to their potential advantages in efficiency, robustness, and scalability when performing complex tasks in dynamic environments. Compared with single-robot systems, multiple robots can cooperate to accomplish tasks such as exploration, surveillance, and transportation more effectively. However, as the number of robots increases, the complexity of coordination, communication, and motion control grows rapidly, posing significant challenges to system design and algorithm performance. Traditional centralized control strategies rely on global information and a central decision unit, which often leads to high communication overhead and limited scalability. Distributed control and learning methods alleviate some of these issues, but they frequently suffer from slow learning convergence and unstable cooperation, especially in large-scale systems. Reinforcement learning, particularly Q-learning, has been widely applied to multi-robot decision-making because of its model-free nature and adaptability. Nevertheless, directly applying conventional Q-learning to multi-robot systems can result in state space explosion and poor coordination among agents. In many practical scenarios, robot communication networks exhibit clear structural patterns, where groups of robots interact more frequently within local regions. Inspired by complex network theory, this paper introduces a community-aware perspective to multi-robot learning and control. By exploiting community structures in communication networks, the proposed approach aims to improve learning efficiency, enhance cooperative behavior, and achieve stable and effective motion control in multi-robot systems.

2. Related Theory and Methods

2.1. Multi-Robot Systems and Network-Based Modeling Foundations

A multi-robot system consists of multiple robot agents equipped with sensing, decision-making, and execution capabilities, which achieve cooperative behavior through communication mechanisms. As illustrated in Figure 1, each robot acts both as an autonomous intelligent agent and as a functional node within the overall system, typically comprising perception, decision and control, and execution and feedback modules. The overall system performance is not a simple sum of individual capabilities but is highly dependent on interaction patterns and cooperation structures among robots. Consequently, network-based modeling provides a fundamental basis for analyzing cooperative control and learning in multi-robot systems [1].

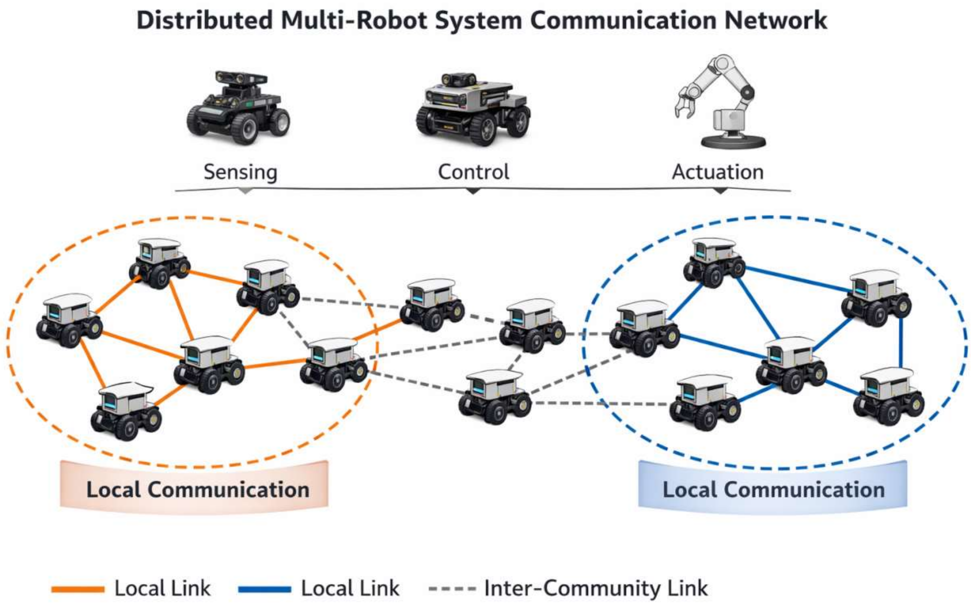

Figure 1. Distributed Multi-Robot System Communication Network.

From a network perspective, a multi-robot system can be abstracted as a communication graph, where nodes represent robots and edges denote information exchange relationships. According to communication range and connection mechanisms, common models include fully connected, local neighborhood, and event-triggered communication. Although fully connected models simplify theoretical analysis, they suffer from high communication costs and limited scalability. Local communication models, in which robots interact only with nearby neighbors, are more realistic and widely adopted, resulting in sparse and dynamically evolving networks. Due to spatial distribution and communication constraints, multi-robot networks often exhibit distinct topological characteristics, such as local clustering and sparse inter-region connections. This naturally motivates the application of complex network theory and community structures. Network-based modeling further enables hierarchical abstraction and information aggregation, revealing structural heterogeneity caused by spatial, task-related, and communication differences. Treating a multi-robot system as a topology-constrained network therefore lays the theoretical foundation for community-aware modeling and efficient cooperative learning and motion control [2].

2.2. Community-Aware Network Theory

Community-aware network theory originates from complex network analysis and focuses on identifying groups of nodes that are densely connected internally but sparsely connected externally, known as communities. These structures capture intrinsic relationships among nodes based on spatial proximity, functional similarity, or interaction frequency, allowing complex systems to be represented in a modular form. In multi-robot systems, variations in robot locations, task roles, and communication ranges naturally give rise to communication networks with pronounced community characteristics, making community-aware modeling particularly suitable for analyzing cooperative behaviors [3].

The formation and evolution of community structures are typically dynamic and hierarchical. As illustrated in Figure 2, a network gradually evolves from an initially unstructured state into a stable community configuration. In the early stage, connections are scattered and community boundaries are unclear. As interactions intensify, certain nodes emerge as hubs, revealing modular patterns. Eventually, the network becomes clearly partitioned into multiple communities with dense intra-community connections and sparse inter-community links maintained by key bridging nodes. Structurally, community-aware networks exhibit high modularity, short intra-community communication paths, and strong local consistency. As shown in Figure 2, information propagates efficiently within communities, while cross-community communication relies on a limited number of bridging nodes. This structure maintains global connectivity while reducing redundant information exchange. When applied to multi-robot systems, communities act as intermediate organizational units that support local coordination and global cooperation, providing a strong theoretical foundation for community-aware reinforcement learning and motion control strategies [4].

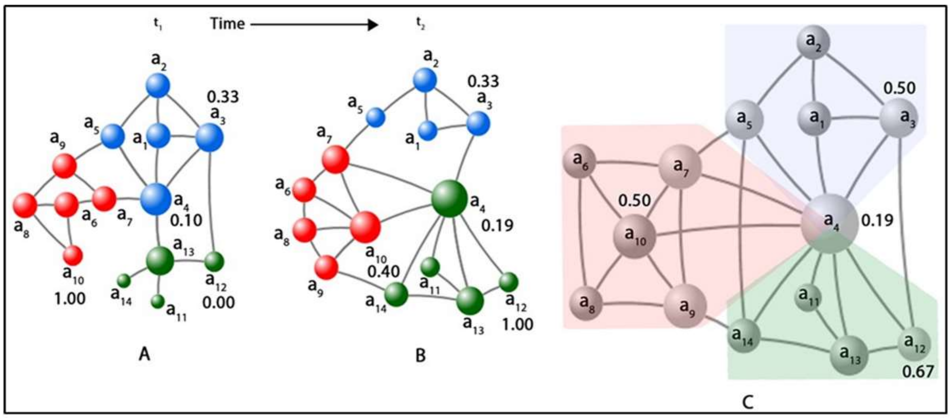

Figure 2. Community-Aware Network.

2.3. Fundamentals of Reinforcement Learning and Q-Learning

Reinforcement learning is a trial-and-error decision-making framework in which agents learn optimal behavior through continuous interaction with the environment. Unlike supervised or unsupervised learning, it relies on reward signals rather than labeled data to guide policy improvement. At each discrete time step, an agent observes the current state, selects an action, and receives a reward as the environment transitions to a new state. By maximizing long-term cumulative rewards, the agent gradually refines its decision-making strategy [5]. This process is commonly modeled as a Markov Decision Process (MDP), defined by states, actions, transition dynamics, and a reward function. In multi-robot systems, each robot functions as an independent learning agent influenced by its own state, neighboring robots, and environmental dynamics. Reinforcement learning is therefore well suited to problems such as path planning, cooperative control, and task allocation. However, increasing system scale leads to rapid growth in state information, posing challenges for learning efficiency and computational cost. Q-learning is a classical value-based reinforcement learning algorithm that directly estimates state-action values without requiring a model of the environment. By iteratively updating Q-values using immediate rewards and discounted future returns, agents can derive optimal or near-optimal policies. Despite its simplicity, Q-learning faces challenges in multi-robot settings due to environmental non-stationarity and state-space explosion. Integrating community-aware structures provides an effective way to introduce information abstraction and coordination, forming the basis for the proposed community-aware Q-learning approach [6].

3. Community-Aware Multi-Robot System Modeling

Building on network-based modeling of multi-robot systems and community-aware network theory, this section presents a unified modeling framework for community-aware multi-robot systems. The primary goal is to introduce community structures as intermediate organizational units while preserving the distributed nature of multi-robot systems, thereby providing a scalable and structured representation for subsequent reinforcement learning and motion control. As illustrated in the figure, the system is modeled as a hierarchical network composed of three layers: the robot individual layer, the community layer, and the system layer, which captures global coordination relationships [7].

As the Figure 3 shown, at the individual layer, each robot is modeled as an autonomous agent with perception, decision-making, and execution capabilities. Its state includes pose, velocity, and locally sensed environmental information, and it communicates with neighboring robots through local, range-limited links. This sparse communication structure supports scalability but also increases the complexity of cooperative decision-making. The community layer acts as a key intermediate structure linking individual robots to system-level coordination [8]. Robots are grouped into communities based on spatial proximity, communication frequency, or task relevance. Dense intra-community connections enable rapid information sharing and local consensus, while sparse inter-community links maintain global connectivity and reduce communication overhead. Community awareness is further embedded in the state representation by incorporating community-level features such as average pose or task progress. This hierarchical modeling framework effectively mitigates state dimensionality growth and provides a unified foundation for the community-aware Q-learning and motion control methods developed in subsequent chapters [9].

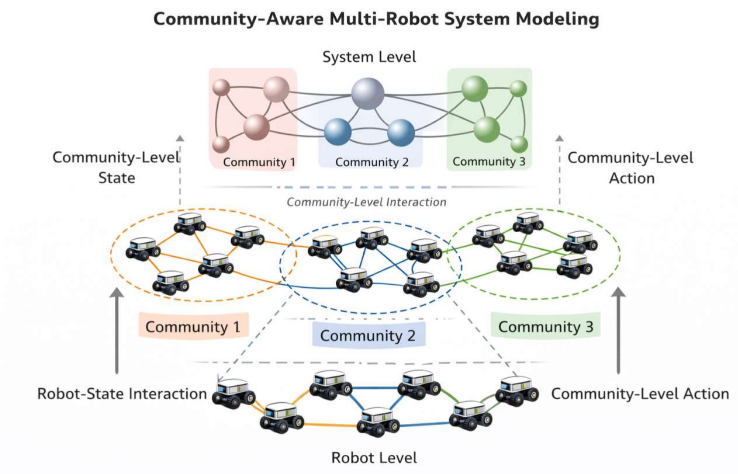

Figure 3. Community-Aware Multi-Robot System Modeling.

4. Community-Aware Q-Learning Algorithm Design

Building on this foundation, this chapter proposes a community-aware multi-robot Q-learning algorithm aimed at improving learning efficiency, cooperative consistency, and scalability. The core idea of the proposed method is to treat community information as an intermediate structural constraint and information aggregation unit. By incorporating community-level state features and community-based rewards into the Q-value update process, robots can maintain distributed learning while forming more stable and consistent cooperative policies. This provides reliable discrete decision outputs for subsequent motion control mapping and coordinated obstacle avoidance [10].Under the community-aware framework, the learning state of robot i at time t is no longer determined solely by its own observation o_i^t or neighborhood information. Instead, it integrates aggregated information from its associated community c(i) to form a structured state representation. To ensure consistency with the modeling , the community aggregation feature $\bar{x}_c^t$ is used to describe overall motion trends or task progress at the community level, and it is concatenated with the individual state to form an extended state s_i^t as shown in Formula 1:

$$s_i^t = [x_i^t,\ \bar{x}_{c(i)}^t], \bar{x}_c^t = \frac{1}{|V_c|}\sum_{j \epsilon V_c} x_j^t \quad (1)$$

Here, x_i^t denotes the local state of robot i , V_c is the set of robots in community ccc, and $|V_c|$ is the community size. The Formula 1 highlights the first role of community-aware modeling: embedding local consistency information into individual learning in a low-dimensional form, allowing robots to perceive their relative position within community behavior and avoiding strategy oscillations caused by purely independent learning [11]. During the value learning stage, a standard Q-learning framework with a discrete action set A_i is adopted to match discrete multi-robot behaviors such as direction selection, speed level adjustment, formation switching, and communication actions. Unlike traditional Q-learning, the reward function is decomposed into an individual reward and a community cooperation reward, balanced by a weighting factor. The individual reward reflects task achievement at the robot level, while the community reward captures cooperative quality within the community, such as formation maintenance or coverage performance. The community-aware reward is defined as shown in Formula 2:

$$r_i^t = r_i^{ind}(t) + \lambda\, r_{c(i)}^{com}(t) \quad (2)$$

where r_i^t is the individual reward, $r_{c(i)}^{com}(t)$ is the community-level cooperative reward, and $\lambda \geq 0$ controls the trade-off between individual efficiency and community consistency. This reward decomposition is intuitive and flexible: larger λvalues emphasize global formation or cooperative coverage, while smaller values favor rapid individual task completion or local obstacle avoidance. Importantly, it provides a stable policy foundation for mapping learning decisions to continuous motion control [12].

For Q-value updates, the classical temporal-difference (TD) learning rule is adopted, with both states and rewards defined in a community-aware form. After robot iii executes action a_i^t in state s_i^t, receives reward r_i^t, and transitions to state s_i^{t+1}, the Q-value is updated as shown in Formula 3:

$$Q_i(s_i^t, a_i^t) \leftarrow (1-\alpha)\, Q_i(s_i^t, a_i^t) + \alpha\, [r_i^t + \gamma \max_{a' \in A_i} Q_i(s_i^{t+1}, a')] \quad (3)$$

where $\alpha \in (0,1]$ is the learning rate and $\gamma \in [0,1)$ is the discount factor. Although Formula 3 has the same form as standard Q-learning, its key differences lie in the community-aware state representation, the inclusion of cooperative rewards, and the stabilizing effect of smooth community aggregation features. Together, these factors reduce non-stationarity in multi-agent learning and improve convergence speed and policy stability [13].

In summary, the proposed community-aware Q-learning algorithm injects community structure into the learning process through community-embedded state representation, cooperative reward design, and context-aware value updates. This approach preserves the simplicity, interpretability, and deployability of Q-learning while significantly enhancing cooperative consistency and learning efficiency in large-scale multi-robot systems. The next chapter builds on these discrete decision policies and discusses how they are mapped to continuous motion control under community-level coordination and inter-community obstacle avoidance constraints [14].

5. Motion Control and Coordination

5.1. Robot Kinematics and Control Model

The community-aware Q-learning framework developed in the previous chapter generates optimal or near-optimal action decisions in a discrete state space. To ensure that these decisions can be effectively executed in real or simulated environments, discrete actions must be mapped to continuous control inputs that satisfy robot kinematic constraints [15]. Accordingly, this section establishes a robot kinematic and control model that defines a clear interface between the learning layer and the control layer, enabling executable cooperative motion as show in figure 4.

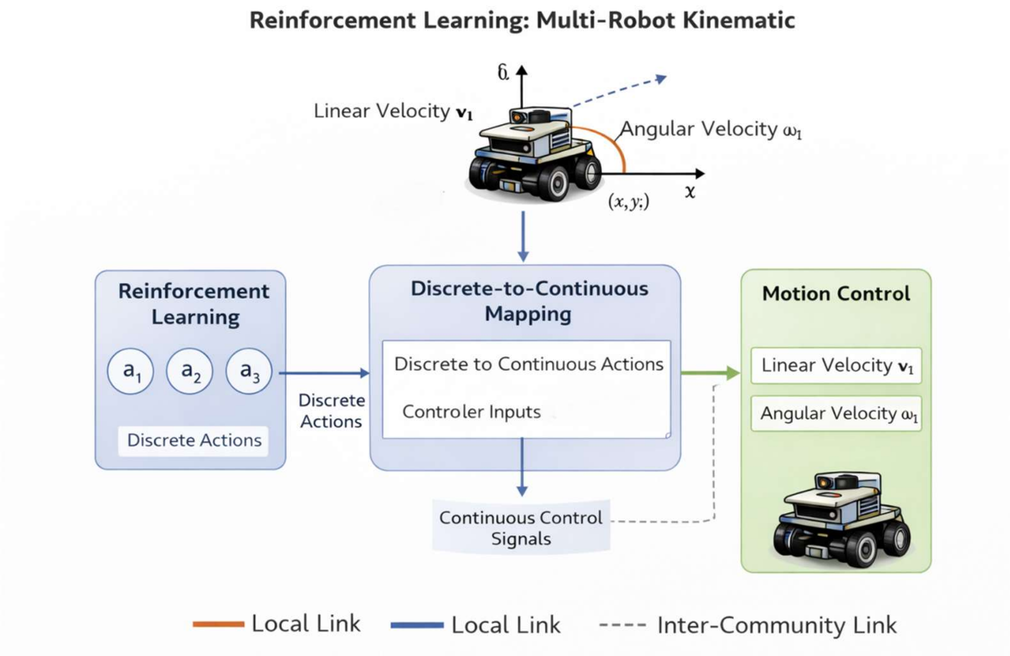

Figure 4. Reinforcement Learning: Multi-Robot Kinematic.

A typical ground mobile robot is considered, modeled as a rigid body moving in a two-dimensional plane. The robot's motion state is described by its position and orientation. To achieve a balance between modeling simplicity and practical applicability, a nonholonomic kinematic model is adopted, in which motion is governed by linear and angular velocities. Control inputs directly drive the robot's low-level actuators, while the high-level learning algorithm outputs abstract motion decisions. This formulation accurately captures the motion characteristics of wheeled robots and facilitates the transformation of discrete actions into continuous control signals. Within this framework, cooperative motion is realized through a layered process comprising decision generation, control mapping, and physical execution. At each decision step, the community-aware Q-learning algorithm selects a discrete action, such as forward motion or turning. These actions are then converted into velocity commands through predefined mapping rules, ensuring policy stability and physical feasibility. By incorporating community information, the control model supports both intra-community coordination and inter-community alignment, achieving local cooperation with global consistency.

5.2. Intra-Community Cooperative Control Strategy

Based on the established robot kinematic model and the mapping from learning decisions to control inputs, this section focuses on intra-community cooperative control strategies. The main objective is to enable robots within the same community to form stable, orderly, and task-oriented collective motion patterns while maintaining individual safety and controllability. By strengthening local cooperation, the overall coordination burden of the system is effectively reduced. Within the community-aware framework, a community is regarded as the basic organizational unit for cooperative motion. Robots in the same community usually exhibit strong spatial proximity or task correlation, and their motion behaviors are expected to maintain a certain degree of consistency. To achieve this, community-level coordination constraints are incorporated into the control layer. When generating continuous control inputs, each robot considers not only its own discrete action selected by the learning algorithm but also aggregated community-level information, such as the average position, velocity, or formation center of the community. These shared features guide robots toward coherent motion directions while preserving individual autonomy. Formation maintenance is an important aspect of intra-community cooperation. By imposing relative position constraints, desired inter-robot distances and orientations are specified according to task requirements. Robots continuously adjust their control inputs based on deviations from these desired relationships, allowing formation errors to gradually converge. This distributed strategy enables stable formation control without centralized coordination. Overall, the proposed approach combines learning-driven decision-making with control-level constraints to ensure efficient and robust intra-community cooperation.

5.3. Inter-Community Interaction and Obstacle Avoidance Mechanism

While intra-community cooperation improves local coordination efficiency, multi-robot systems operating across multiple communities must also manage interactions and conflicts between communities. Different communities often pursue distinct local objectives or occupy separate spatial regions, resulting in variations in motion direction, speed, and formation patterns. Without effective inter-community coordination and obstacle avoidance, conflicts may occur in overlapping regions, potentially compromising system stability. Therefore, this section develops an inter-community interaction and obstacle avoidance mechanism to ensure safe and stable system operation. In the proposed framework, each community is treated as a high-level coordination unit. Inter-community information exchange is realized through a limited number of cross-community links. Unlike frequent intra-community communication, inter-community interactions follow a low-frequency, key-information-sharing strategy, where only aggregated community-level states-such as centroid position, average velocity, and task priority-are transmitted when necessary. This approach preserves global connectivity while reducing communication overhead and avoiding unnecessary interference. Robots utilize this information to adjust their motion trends in advance, enabling coordinated movement among neighboring communities. To mitigate collision risks, a community-boundary-based obstacle avoidance mechanism is incorporated into the control layer. When robots approach inter-community regions, their control inputs are influenced by both intra-community coordination constraints and safety constraints derived from neighboring communities. By monitoring relative distances and velocities, potential collisions are detected early and control commands are adjusted to maintain safe separation. In addition, reward modulation in the community-aware Q-learning framework reinforces smooth and collision-free interactions, encouraging proactive and cooperative avoidance behavior in complex multi-community environments.

6. Simulation and Performance Evaluation

6.1. Simulation Environment and Parameter Settings

To verify the effectiveness and stability of the proposed community-aware multi-robot Q-learning and cooperative motion control method in complex environments, a unified simulation platform was constructed for systematic performance evaluation. The simulation environment was designed following the principles of reproducibility, comparability, and scalability, ensuring that experimental results objectively reflect the advantages of the community-aware mechanism in learning efficiency, cooperative stability, and system safety.

A two-dimensional simulation framework was adopted to model wheeled mobile robots, each satisfying the nonholonomic kinematic constraints described in the previous section. Robots execute continuous control inputs generated by reinforcement learning policies within a bounded planar workspace containing static obstacles and accessible regions, representing typical cooperative task scenarios in constrained environments. Robots belonging to different communities are initialized in separate regions, and both community size and the number of communities can be adjusted to evaluate system performance under multi-community parallel operation. The experimental scenarios emphasize the influence of community structure on system behavior. Robots first perform intra-community cooperative tasks such as formation maintenance and coordinated movement. When community trajectories intersect or overlap, inter-community interaction and obstacle avoidance mechanisms are activated to test system safety and stability in dense environments. To ensure fairness and repeatability, key learning and control parameters were fixed across experiments. As summarized in Table 1, these parameters provide a consistent baseline for subsequent comparative analysis.

Table 1. Simulation Environment and Main Parameter Settings.

Parameter	Value	Description
Simulation area size	50 m × 50 m	Two-dimensional workspace
Number of robots	20-40	Scale of the multi-robot system
Number of communities	3-5	Number of network communities
Robots per community	Variable	Size of each community
Maximum linear velocity	1.0 m/s	Motion constraint of robots
Maximum angular velocity	1.0 rad/s	Turning constraint of robots
Learning rate	0.1	Step size of Q-learning updates
Discount factor	0.95	Weight of future rewards
Community reward weight	0.3-0.6	Trade-off between individual and community cooperation
Decision interval	0.1 s	Reinforcement learning decision period
Simulation steps	5,000-10,000	Duration of a single experiment

6.2. Comparative Experiment Design

To quantitatively assess the performance advantages of the proposed community-aware multi-robot Q-learning method, a set of comparative experiments was designed focusing on learning efficiency, cooperative stability, and system safety. The core objective is to compare multi-robot Q-learning algorithms with and without community awareness under identical simulation conditions, thereby isolating the effect of community structures on cooperative learning and motion control. A conventional distributed multi-robot Q-learning method was selected as the baseline approach. In this method, each robot independently updates its Q-values based solely on its own state and local neighborhood information, without introducing community-level state representations or cooperative rewards. This baseline represents a widely used implementation in multi-robot reinforcement learning research. To ensure a fair comparison, both methods share the same robot models, action definitions, kinematic constraints, and learning parameters. The only difference lies in whether community-aware state aggregation and community-level rewards are incorporated. For each experimental run, robot initial positions and community assignments are randomly generated within predefined ranges, as shown in Figure 5, it illustrates a typical initialization layout in the 2D simulation environment, where distinct colors denote different community assignments, and shaded boundary regions represent the predefined deployment ranges for each community. Furthermore, random initial heading vectors (indicated by arrows) are assigned to each robot to simulate diverse motion intentions. Both algorithms are executed in the same environment, and each experiment is repeated multiple times to reduce randomness. During execution, key performance metrics are recorded over time to analyze convergence behavior and cooperative dynamics. Experimental scenarios include intra-community cooperative motion, inter-community collision avoidance, and complex multi-community interactions.

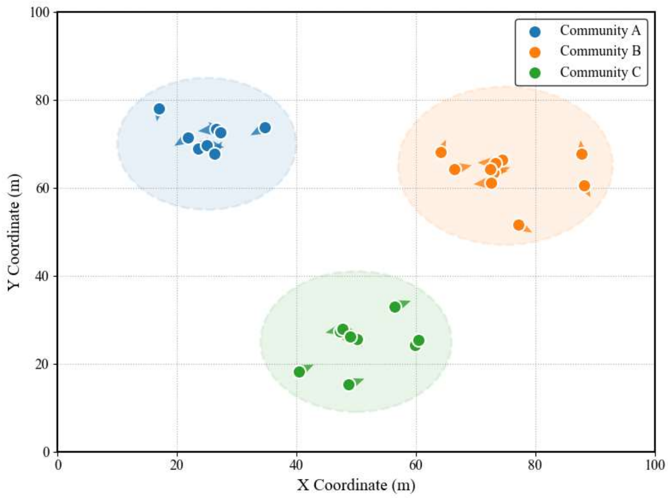

Figure 5. Initial Spatial Distribution and Community Assignments.

6.3. Experimental Results and Analysis

Based on the simulation environment and comparative experiment design, experimental results are analyzed from three key perspectives: learning convergence speed, path efficiency, and cooperative performance. All reported results are averaged over multiple independent runs to mitigate the effects of random initialization and exploration strategies, ensuring reliable conclusions.

From the perspective of learning convergence, performance is evaluated by analyzing the evolution of cumulative rewards and the number of training steps required to reach stable performance. As shown in Figure 6, the community-aware Q-learning method converges significantly faster than the traditional approach. By incorporating community-level state features and cooperative rewards, learning oscillations caused by environmental non-stationarity are effectively reduced, resulting in smoother and more stable value updates.

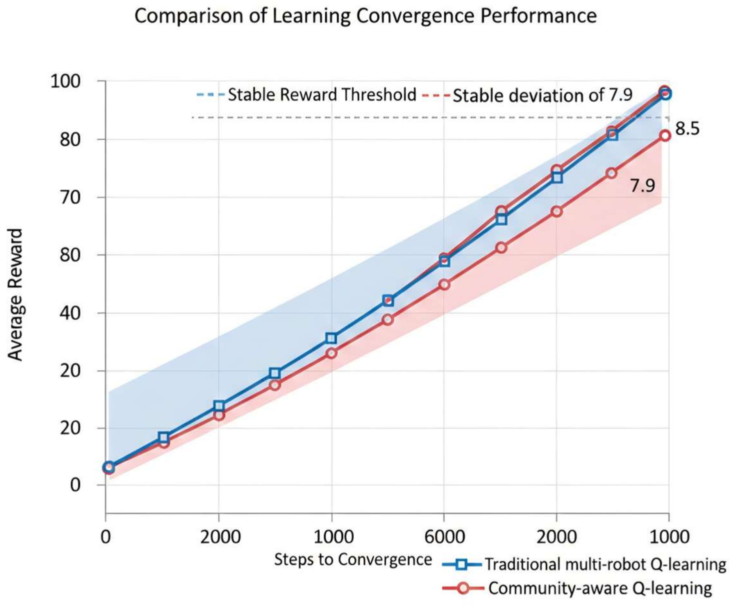

Figure 6. Comparison of Learning Convergence Performance.

Path efficiency is assessed using average path length and task completion time. The baseline method often exhibits redundant movements and local congestion due to the lack of explicit cooperation modeling. In contrast, the community-aware approach guides robots toward coordinated motion patterns within communities, leading to shorter paths and faster task completion. These improvements are clearly reflected in Figure 7.

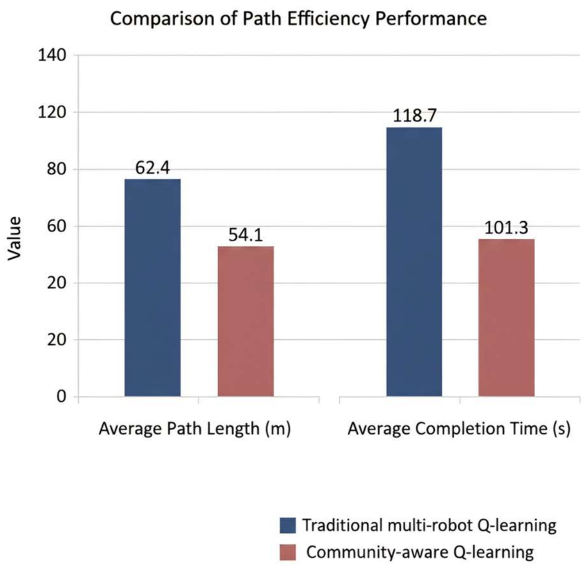

Figure 7. Comparison of Path Efficiency Performance.

Finally, cooperative performance and system safety are evaluated using intra-community formation error and inter-community conflict frequency. As reported in Figure 8, the proposed method significantly reduces formation deviation and collision events, demonstrating improved stability in dense multi-community scenarios. Overall, the results confirm that introducing community-aware mechanisms enhances learning efficiency, coordination quality, and system robustness.

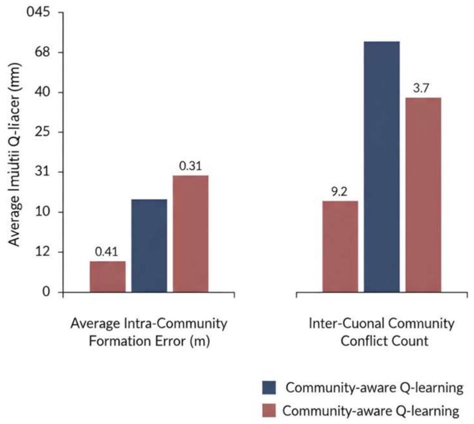

Figure 8. Comparison of Cooperative Performance and System Safety.

7. Conclusions

This study explores multi-robot Q-learning and motion control within a community-aware network framework. By introducing community structures as intermediate organizational units, the proposed approach addresses the scalability and stability limitations of conventional multi-robot reinforcement learning. Community-level information is incorporated into state representation and reward design, improving learning convergence and policy stability while preserving distributed learning properties. In addition, intra- and inter-community coordination mechanisms are integrated with robot kinematic models to ensure smooth and safe motion execution. Simulation results show that the proposed method outperforms traditional approaches in convergence speed, path efficiency, and cooperative performance, particularly in dense multi-community environments, demonstrating its effectiveness and practical applicability.

References

1. P. Tsiotras, M. Gombolay, and J. Foerster, "Decision-making and planning for multi-agent systems," *Frontiers in Robotics and AI*, vol. 11, p. 1422344, 2024.
2. E. Sebastián, T. Duong, N. Atanasov, E. Montijano, and C. Sagüés, "Physics-informed multi-agent reinforcement learning for distributed multi-robot problems," *IEEE Transactions on Robotics*, 2025.
3. X. Zhou, X. Shi, L. Zhang, C. Chen, H. Li, L. Ma, and J. Chen, "Scalable Hierarchical Reinforcement Learning for Hyper Scale Multi-Robot Task Planning," *arXiv preprint arXiv:2412.19538*, 2024.
4. Y. Liu, D. Wu, and Y. Liang, "Survey on Graph-Based Reinforcement Learning for Networked Coordination and Control," *Automation*, vol. 6, no. 4, p. 65, 2025. doi: 10.3390/automation6040065
5. S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, and C. Finn, "Robonet: Large-scale multi-robot learning," *arXiv preprint arXiv:1910.11215*, 2019.
6. N. Xie, Y. Hu, and L. Chen, "A distributed multi-agent formation control method based on deep Q learning," *Frontiers in Neurorobotics*, vol. 16, p. 817168, 2022. doi: 10.3389/fnbot.2022.817168
7. M. G. Quiles, L. Zhao, R. L. Alonso, and R. A. Romero, "Particle competition for complex network community detection," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 18, no. 3, 2008.
8. Y. Du, B. Liu, V. Moens, Z. Liu, Z. Ren, J. Wang, and H. Zhang, "Learning correlated communication topology in multi-agent reinforcement learning," In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, May, 2021, pp. 456-464. doi: 10.65109/fxam2191
9. S. Devaraju, S. Garg, A. Ihler, E. S. Bentley, and S. Kumar, "Pipe routing with topology control for decentralized and autonomous UAV networks," *Drones*, vol. 9, no. 2, p. 140, 2025. doi: 10.3390/drones9020140
10. G. S. Mahalakshmi, and T. V. Geetha, "Multi-robot learning using non-deterministic argument games," *International Journal of Autonomous and Adaptive Communications Systems*, vol. 3, no. 4, pp. 439-463, 2010.

11. F. Venturini, "Multi-Agent Reinforcement Learning of Swarm Behaviours with Graph Neural Networks: prototype and first experiments," .
12. P. Singh, R. Tiwari, and M. Bhattacharya, "Navigation in Multi Robot system using cooperative learning: A survey," In *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, March, 2016, pp. 145-150. doi: 10.1109/icctict.2016.7514569
13. G. E. Setyawan, P. Hartono, and H. Sawada, "Cooperative multi-robot hierarchical reinforcement learning," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 9, 2022. doi: 10.14569/ijacsa.2022.0130904
14. L. Zhou, P. Yang, C. Chen, and Y. Gao, "Multiagent reinforcement learning with sparse interactions by negotiation and knowledge transfer," *IEEE transactions on cybernetics*, vol. 47, no. 5, pp. 1238-1250, 2016. doi: 10.1109/tcyb.2016.2543238
15. J. Yin, W. Rao, Y. Xiao, and K. Tang, "Cooperative Path Planning With Asynchronous Multiagent Reinforcement Learning," *IEEE Transactions on Mobile Computing*, 2025. doi: 10.1109/tmc.2025.3526979