Communication-efficient Multi-Agent Reinforcement Learning with Spatiotemporal Information Hub

Ling Ding[1], Tianbai Lyu[1], Zhiliang Bi[1], Hao Wang[1*], Shanshan Feng[1], Wei Yu[2*]

[1]School of Computer Science, Wuhan University, China
[2]School of Artificial Intelligence, Wuhan University, China
{lingding, lyutianbai, bizhiliang, wanghao.cs, victor_fengss, yuwei}@whu.edu.cn

Abstract

Centralized training with decentralized execution (CTDE) is a framework for MARL with wide applications. In the CTDE paradigm, agents leverage global state information during training to mitigate the non-stationarity of the MARL environment, but must rely solely on partial observations during execution. Recent work has highlighted the growing importance of inter-agent communication for more effective learning and coordination. However, most existing methods overlook the fact that real-world communication channels are often bandwidth-constrained and imperfectly reliable. Toward more communication-efficient and robust MARL, we extend the conventional CTDE framework with an information hub. The hub collects local observations from the agents to restore the global state, which is then delivered to the agents on demand. To this end, technical mechanisms are designed to enable effective global reconstruction with incomplete observations, as well as agent-specific attention to the reconstructed global information. Experiments on multiple cooperative MARL benchmarks demonstrate that our method achieves state-of-the-art performance compared to popular MARL algorithms while substantially reducing communication overhead and exhibiting strong robustness under imperfect communication channels.

1 Introduction

Multi-Agent Reinforcement Learning (MARL) has recently garnered widespread attention, achieving impressive results across various complex domains, such as traffic signal control (Bie, Ji, and Ma 2024), droplet routing (Liang 2021), active voltage regulation (Wang et al. 2021), dynamic algorithm configuration (Xue et al. 2022). In practice, multi-agent systems are typically driven by the joint decisions of agents with limited perception and are thus essentially non-stationary. To implement effective multi-agent learning, the *Centralized Training with Decentralized Execution* (CTDE) paradigm has been widely adopted (Kraemer and Banerjee 2016; Rashid et al. 2020; Sunehag et al. 2017), where agents may access the global (total) information during training but should rely solely on individual (partial) observations during execution (Jianye et al. 2022; Rashid et al. 2020; Wang et al. 2020). The basic idea of CTDE is quite straightforward: Using the global information provided during training, each agent learns a policy to react to its observations, which is then applied independently during execution. The joint action of all agents is effective if the policy of each agent serves the system goal well via local reactions.

Despite the success of CTDE in various applications, it has been noted that execution-time information exchanges via proper *communication* mechanisms could still be beneficial toward more effective learning (Wang et al. 2019; Guan et al. 2024). NDQ (Wang et al. 2019) broadcasts the observation histories of the agents to ensure expressive and succinct communication. To mitigate communication overhead, TMC (Zhang, Zhang, and Lin 2020) broadcasts messages only when the deviation from previous transmissions is significant, otherwise reusing historical information. In MAIC (Yuan et al. 2022), each agent models its teammates to generate tailored incentive messages, which are then broadcast to influence the Q-value updates of others. Recently, MASIA (Guan et al. 2024) enables all agents to broadcast their local observations, and introduces self-supervised objectives to ensure that the aggregated representation is both compact and sufficient. TGCNet (Zhang et al. 2025) leverages a multi-key gated mechanism to dynamically construct a directed communication graph, and leverages this structure to reduce reliance on global state information.

Although effective in terms of the learned policies, most existing solutions are *inefficient* in communication as they rely on the *broadcasting* mechanism. For example, MASIA (Guan et al. 2024) relies on message broadcasting from all agents to estimate the global state at each timestep. TGCNet (Zhang et al. 2025) requires access to all agents' hidden representations at each timestep to construct the communication graph. Assuming a constant cost for each message transmission, these methods incur a substantial quadratic communication overhead at each timestep.

In many real-world applications, such as traffic signal control (Bie, Ji, and Ma 2024) and real-time strategy games (Samvelyan et al. 2019), communication channels are *bandwidth-constrained*. Excessive message passing may congest the system, preventing the agents from making timely decisions. In addition, the communication is often *imperfectly reliable* due to various factors, subject to message losses and delays. Ignoring the imperfections of communi-

*Corresponding authors
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

cation may lead to the ultimate failure of decision-making or multi-agent coordination. *These practical constraints highlight a pressing need for multi-agent communication mechanisms that are both cost-efficient and failure-tolerant.*

In response to the above needs, we propose a novel communication-efficient MARL framework by augmenting CTDE with an information hub. The framework, named Multi-Agent spatioTemporal Communication Hub (MATCH), can effectively capture spatiotemporal dependencies among agents for robust state reconstruction. Specifically, the information hub collects agent-centric observations and uses a Transformer-based spatiotemporal interpolator to restore the global information. The agents may request the global information from the hub. To keep the agents focused on their local circumstances and tasks, the hub employs a position-aware attention module to personalize global information for each agent. The information hub operates independently and asynchronously with the agents. In the worst case, if there is a fatal failure in the information hub during execution, the entire multi-agent system can still operate following the conventional CTDE paradigm. Thus, our proposed method is both efficient (with linear communication costs) and robust.

In summary, our main contributions are as follows:

- First, we propose a novel paradigm for cooperative MARL with a centralized information hub, which can significantly reduce communication overhead and remain robust to unstable communication channels.
- Second, we design a communication mechanism between agents and the information hub, which enables effective global information restoration and personalization. The techniques are based on recent advances in Transformer and attention mechanisms.
- Last but not least, we evaluate our method using various tasks in the benchmarks of SMAC (Samvelyan et al. 2019) and MPE (Lowe et al. 2017). Our method outperforms state-of-the-art approaches, achieving up to 12.9× lower communication volume while maintaining robust performance under message loss.

2 Related Work

Centralized Training with Decentralized Execution (CTDE) has become the dominant framework in cooperative multi-agent reinforcement learning (MARL). CTDE-based approaches can be broadly categorized into policy-based and value-based methods. Representative policy-based methods include COMA (Foerster et al. 2018), MADDPG (Lowe et al. 2017), and MAPPO (Yu et al. 2022), while value-based methods such as VDN (Sunehag et al. 2017), QMIX (Rashid et al. 2020), and QPLEX (Wang et al. 2020) have demonstrated strong empirical performance across a range of cooperative tasks. While CTDE offers a strong foundation for training, introducing communication allows agents to compensate for their limited local observations and promotes more effective collaboration (Wang et al. 2019).

Early works explored neighbor-based communication to address limited local observability (Guo, Shi, and Fan 2023; Jiang et al. 2018). For instance, ATOC (Jiang and Lu 2018) allows agents to dynamically form communication groups and share information. But the similarity of local observations often limits the diversity. Broadcasting enables more thorough information exchange among agents (Guan et al. 2024; Yuan et al. 2022; Zhang, Zhang, and Lin 2020). RIAL and DIAL (Foerster et al. 2016) enhance communication by employing a broadcasting mechanism that shares messages across time steps. TarMAC (Das et al. 2019) leverages key-value broadcasting to differentiate the importance of incoming messages. To further reduce communication overhead, the concept of dynamic communication topology has been introduced (Meng and Tan 2024; Wang and Sartoretti 2022). For example, MAGIC (Niu, Paleja, and Gombolay 2021) constructs communication graphs to determine message targets, G2ANet (Liu et al. 2020) builds an interaction graph via a two-stage attention mechanism, and TGCNet (Zhang et al. 2025) designs a multi-key gated network to learn a dynamic directed graph structure. Nevertheless, peer-to-peer communication still incurs substantial bandwidth consumption and redundant message processing.

Besides, centralized communication enables agents to transmit information to and from a central unit, resulting in lower communication overhead (Meng and Tan 2023; Liu et al. 2021). For instance, CommNet (Sukhbaatar, Fergus et al. 2016) aggregates messages from all agents at each timestep to construct a centralized global signal. Gated-ACML (Mao et al. 2020) and IC3Net (Singh, Jain, and Sukhbaatar 2018) introduce gating mechanisms that allow agents to selectively transmit information to the central module. Nevertheless, most centralized communication approaches assume that all messages can be stably collected within a single timestep, which is hardly attainable under real-world communication constraints.

In real-world communication environments, where bandwidth is limited and connections are inherently unstable, existing methods, to the best of our knowledge, fail to fully adapt to such imperfect conditions. MATCH employs a centralized communication module that captures spatiotemporal dependence to simulate the global state, thereby reducing communication overhead and improving robustness.

3 Dec-POMDP with Communication

In this work, we model cooperative MARL as a slightly extended Decentralized Partially Observable Markov Decision Process (Dec-POMDP) (Oliehoek, Amato et al. 2016):

$$\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \Omega, \mathcal{O}, \mathcal{R}, \gamma; \mathcal{M}, \mathcal{C} \rangle.$$

Here, $\mathcal{N} = \{1, 2, \cdots, n+m\}$ denotes the set of n agent entities $(1, 2, \cdots, n)$ and m non-agent entities $(n+1, n+2, \cdots, n+m)$ in the system; $\mathcal{S}$ is the global state space; $\mathcal{A} = \prod_{i=1}^{n} A_i$ is the joint action space, spanned by the individual action spaces A_i of the agents. The transition probability function $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ defines the dynamics of the environment; Ω is the set of possible observations; $\mathcal{O} : \mathcal{S} \times \mathcal{N} \to \Omega$ is the observation function mapping the global state into individual observations of the agents; $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a shared reward function of the multi-agent system; and $\gamma \in (0, 1)$ is the discount factor.

Our extension to Dec-POMDP is the explicit inclusion of the message space $\mathcal{M}$ and the communication mechanism $\mathcal{C}$ for inter-agent information exchange. At each timestep t, agent $i \in \mathcal{N}$ works in the following routine:

- **Observe** $o_t^i = \mathcal{O}(s_t, i) \in \Omega$ based on the current environment state $s_t \in \mathcal{S}$.
- **Inquire** message $m_t^i \in \mathcal{M}$, if necessary, via the communication mechanism $\mathcal{C}$.
- **Decide** action a_t^i using a policy function $\pi^i\left(a_t^i \left| \tau_t^i, m_t^i\right.\right)$, where $\tau_t^i = \left(o_1^i, a_1^i, o_2^i, a_2^i, \cdots, o_{t-1}^i, a_{t-1}^i, o_t^i\right)$ is the local interaction history.
- **Send** message $z_t^i \in \mathcal{M}$ via the communication mechanism $\mathcal{C}$, if necessary.

The joint action of all the agents $a_t = \left\langle a_t^1, a_t^2, \cdots, a_t^n \right\rangle$ leads to a state transition from s_t to s_{t+1} with probability $\mathcal{P}\left(s_{t+1} \left| s_t, a_t \right.\right)$, for which the agents receive a global reward of $\mathcal{R}(s_t, a_t)$. The primary objective of the agents is to learn a joint policy $\pi = \left\langle \pi^1, \pi^2, \cdots, \pi^n \right\rangle$ that maximizes the expected cumulative reward through the global action-value function

$$Q_{\text{tot}}^{\pi,\mathcal{C}}(\tau, a) = \mathbb{E}_{s,a}\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t) \middle| s_0 = s, a_0 = a, \pi, \mathcal{C}\right],$$

where $\tau = \langle \tau^1, \tau^2, \cdots, \tau^n \rangle$ denotes the joint interaction history of all agents.

Considering real-world application scenarios, we want the learning process to be *communication-efficient* and *robust*. Specifically, we assume that the cost of sending/receiving a message m is proportional to the size of the message, i.e., $\text{cost}(m) = O(|m|)$. In addition, to model message losses, we introduce an *information loss rate* $\kappa \in [0, 1]$, denoting the probability that a message fails to reach its destination. The learning agents are expected to be capable of dealing with such real-world imperfections at a relatively low cost.

4 Method

In this section, we describe our proposed method, MATCH (Multi-Agent SpatioTemporal Communication Hub), an extension to the CTDE paradigm with efficient and robust communication. We first introduce the design methodology and overall framework of MATCH, and then explain the main modules in detail.

4.1 Methodology and Overall Framework

To reduce the quadratic communication overhead while keeping the effectiveness of learning, we extend the CTDE framework with a centralized *information hub*, as shown in Fig. 1. The information hub, as its name suggests, is a centralized mechanism to coordinate communication between agents. The main function is to receive local observations from the agents and aggregate them into estimations of the global state, which are then redistributed to the agents for more effective learning and decision-making.

Following the routine of the extended Dec-POMDP, at each timestep t, agent i observes o_t^i. In this work, we consider *entity-based observations*, i.e.,

$$o_t^i = \left\{\left(j, \boldsymbol{p}_t^{ij}, \boldsymbol{v}_t^j\right)\right\}_{j \in \mathcal{N}},$$

where j is the entity ID, $\boldsymbol{p}_t^{ij}$ the relative position of entity j with respect to agent i, and $\boldsymbol{v}_t^j$ the feature data of entity j. Agent i also packs the observation and sends it to the hub. The hub works asynchronously with the agents. After collecting a batch of messages $B_t = \left\{o_t^i\right\}_{i \in \mathcal{N}}$, the hub attempts to restore (or, more generally, estimate) a global representation of the entities $\hat{\boldsymbol{s}}_t$ from B_t, i.e.,

$$\hat{\boldsymbol{s}}_t = \textsc{Estimate}\left(B_t\right). \tag{1}$$

Since the global representation $\hat{\boldsymbol{s}}_t$ is often rich in information and large in size, a *query* mechanism enables each agent to fetch self-related data from $\hat{\boldsymbol{s}}_t$, avoiding information overload and unnecessary network communication. Specifically, the query $\boldsymbol{q}_t^i$ is generated with τ_t^i, the historical observations of agent i at time t:

$$\boldsymbol{q}_t^i = \textsc{Query}\left(\tau_t^i\right). \tag{2}$$

On receiving a query $\boldsymbol{q}_t^i$, the information hub answers the query to generate a reply message

$$\boldsymbol{m}_t^i = \textsc{Answer}\left(\hat{\boldsymbol{s}}_t, \boldsymbol{q}_t^i\right). \tag{3}$$

The message $\boldsymbol{m}_t^i$ is then used by the agent in the action-value function $Q^i\left(\tau_t^i, \boldsymbol{m}_t^i, a_t^i\right)$ for action selection.

4.2 Information Hub

As depicted in Fig. 1, the hub receives messages in a batch $B_t = \left\{\boldsymbol{o}_t^1, \boldsymbol{o}_t^2, \cdots, \boldsymbol{o}_t^n\right\}$, where

$$\boldsymbol{o}_t^i = \left(\boldsymbol{o}_t^{i1}, \boldsymbol{o}_t^{i2}, \cdots, \boldsymbol{o}_t^{i(n+m)}\right),$$

with $\boldsymbol{o}_t^{ij} = \left(\boldsymbol{p}_t^{ij}, \boldsymbol{v}_t^j\right)$ referring to the relative position (concerning agent i, the observer) and feature of entity j.

Information Aggregation. Since the same entity may be observed by multiple agents from different perspectives, the received messages often contain redundant information and viewpoint inconsistencies. The hub employs a *self-attentive pooling* mechanism to integrate multi-view features and resolve conflicts. Specifically, the aggregated embedding of entity j is computed as

$$\boldsymbol{e}_t^j = \sum_{i=1}^{n} \alpha_t^{ij} \cdot \left(\boldsymbol{W}_{\text{pool}} \boldsymbol{o}_t^{ij}\right),$$

where

$$\alpha_t^{ij} = \frac{\exp\left(\boldsymbol{w}_{\text{pool}}^\top \boldsymbol{W}_{\text{pool}} \boldsymbol{o}_t^{ij}\right)}{\sum_{k=1}^{n} \exp\left(\boldsymbol{w}_{\text{pool}}^\top \boldsymbol{W}_{\text{pool}} \boldsymbol{o}_t^{kj}\right)}$$

are softmax weights and $(\boldsymbol{w}_{\text{pool}}, \boldsymbol{W}_{\text{pool}})$ are parameters.

Through pooling, we obtain an initial representation of entities $\boldsymbol{e}_t = \left(\boldsymbol{e}_t^1, \boldsymbol{e}_t^2, \cdots, \boldsymbol{e}_t^{n+m}\right)$. Nonetheless, $\boldsymbol{e}_t$ may still be incomplete due to restricted observation scope or communication failures. Thus, we investigate how missing information can be effectively reconstructed. The key insight here is that *messages from agents are strongly connected in both space and time*. Spatially, the local observations of agents are correlated as they depict the same global state; Temporally, missing information of an entity can often be inferred from its past due to the temporal continuity of intention.

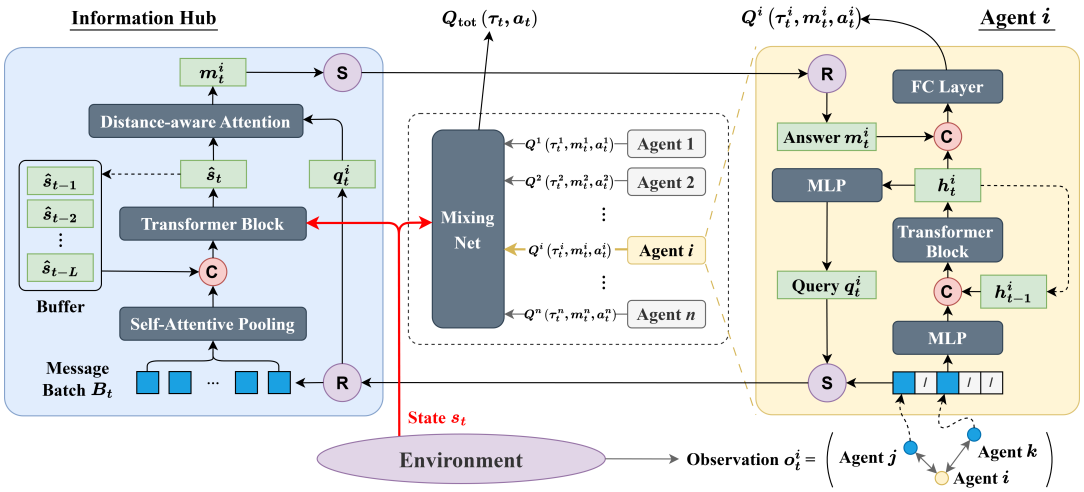


Figure 1: An illustration of our proposed framework. The framework is CTDE augmented with a shared information hub. The information hub functions independently and asynchronously with the agents.

Spatiotemporal Reconstruction. To exploit these patterns, we introduce a Transformer-based interpolation module that captures both spatial and temporal correlations to reconstruct the global information $\hat{\boldsymbol{s}}_t \in \mathbb{R}^{(n+m)\times d}$, consisting of a d-dimensional representation of each of the $n+m$ entities. This reconstruction module concretely implements the estimation step defined in Equation 1. To incorporate temporal context, we buffer a list of the past L reconstructed global information $\boldsymbol{H}_t = [\hat{\boldsymbol{s}}_{t-L}, \hat{\boldsymbol{s}}_{t-L+1}, \cdots, \hat{\boldsymbol{s}}_{t-1}]$. At time t, we reshape the concatenated $\boldsymbol{X}_t = (\boldsymbol{H}'_t \,\|\, \boldsymbol{e}_t)$ into a sequence of $(L+1)\times(n+m)$ tokens, where $\boldsymbol{H}'_t$ is a lightly projected form of $\boldsymbol{H}_t$ matching the dimensionality of $\boldsymbol{e}_t$. To preserve structural priors, we incorporate position encodings using separable 2D embeddings (temporal and entity indices). The resulting token sequence is then fed into a stack of Transformer blocks for spatiotemporal reconstruction

$$\tilde{\boldsymbol{X}}_t = \text{Transformer}\left(\boldsymbol{X}_t\right).$$

After such a process, the missing information is recovered in $\tilde{\boldsymbol{X}}_t$. By passing $\tilde{\boldsymbol{X}}_t$ through a lightweight decoder, the global information $\hat{\boldsymbol{s}}_t = \left(\hat{\boldsymbol{s}}_t^1, \hat{\boldsymbol{s}}_t^2, \cdots, \hat{\boldsymbol{s}}_t^{n+m}\right)$ is reconstructed, with component $\hat{\boldsymbol{s}}_t^j \in \mathbb{R}^d$ for entity j.

Distance-aware Attention. Directly distributing an identical global information $\hat{\boldsymbol{s}}_t$ to all agents makes it difficult for them to focus on individually relevant content (Chen et al. 2022). Intuitively, the attention of an agent often exhibits *locality*, focusing primarily on nearby entities. Therefore, we use a *distance-aware attention* module, serving as a concrete implementation of Equation 3, to generate agent-specific global information, which assigns higher weights to nearby entities and attenuates the influence of those farther away. Specifically, given a query $\boldsymbol{q}_t^i$ from agent i and a global representation $\hat{\boldsymbol{s}}_t$, the hub incorporates inter-entity distances as bias terms into the attention scores, i.e.,

$$a_t^{ij} = \lambda \cdot \left(\frac{\boldsymbol{q}_t^{i\top} \boldsymbol{W}_{\text{attn}} \hat{\boldsymbol{s}}_t^j}{\sqrt{d}}\right) \cdot \text{sigmoid}\left(-D_t^{ij}\right),$$

where $(\lambda, \boldsymbol{W}_{\text{attn}})$ are network parameters and D_t^{ij} the normalized Euclidean distance between agent i and entity j at time t. The final message $\boldsymbol{m}_t^i$ answering query $\boldsymbol{q}_t^i$ is thus $\boldsymbol{m}_t^i = \sum_{j=1}^{n+m} \beta_t^{ij} \cdot \hat{\boldsymbol{s}}_t^j$, where β_t^{ij} is the softmax weight derived from a_t^{ij}.

4.3 Communication-enhanced Agent

Given the above-described information hub, it remains to be clarified how agent i generates the query $\boldsymbol{q}_t^i$ and exploits the answer $\boldsymbol{m}_t^i$.

As depicted in Fig. 1, agent i first converts its entity-based observation into a fixed-size representation $\boldsymbol{o}_t^i = \left(\boldsymbol{o}_t^{i1}, \boldsymbol{o}_t^{i2}, \cdots, \boldsymbol{o}_t^{i(n+m)}\right)$ with padding zeros for missing information. To align the incoming message from the hub and historical observations, these entity features are then processed by an *entity-level Transformer* module (Gallici, Martin, and Masmitja 2023). Specifically, the historical hidden state $\boldsymbol{h}_{t-1}^i$ from time $t-1$ is concatenated with the current entity features $\boldsymbol{o}_t^i$. The resulting vector is then fed into the Transformer blocks. Through multi-head self-attention, the Transformer produces a new hidden state $\boldsymbol{h}_t^i$ that encodes the attention relationships among entities.

Note that *an agent's attention also exhibits continuity in time*. For example, if an entity was previously observed but found to be irrelevant to decision-making, the agent will tend to assign it lower attention until the situation significantly changes. Therefore, we model the query $\boldsymbol{q}_t^i$ as a function of the history $\boldsymbol{h}_t^i$, i.e.,

$$\boldsymbol{q}_t^i = \Phi\left(\boldsymbol{h}_t^i; \boldsymbol{\theta}_{\text{query}}^i\right),$$

representing the agent-specific focus on global information, where $\boldsymbol{\theta}^i_{\text{query}}$ are learnable parameters, and the function Φ concretely implements the QUERY operation in Equation 2.

Agent i sends the query vector $\boldsymbol{q}^i_t$ along with the observation $\boldsymbol{o}^i_t$ to the information hub. The hub answers the query $\boldsymbol{q}^i_t$ with a message $\boldsymbol{m}^i_t$. The final Q-value for agent i to take action a^i_t at time t is computed by combining the message $\boldsymbol{m}^i_t$ with the hidden state $\boldsymbol{h}^i_t$, i.e.,

$$Q^i_t\left(\tau^i_t, \boldsymbol{m}^i_t, \cdot\right) = \Psi\left(\boldsymbol{h}^i_t, \boldsymbol{m}^i_t; \boldsymbol{\theta}^i_{\text{value}}\right),$$

where $\boldsymbol{\theta}^i_{\text{value}}$ are learnable network parameters.

4.4 Centralized Training

The training process of our proposed framework is centralized, supervised by the true global state $\boldsymbol{s}_t$ over time.

State Reconstruction Loss. For the information hub to obtain an accurate estimate $\hat{\boldsymbol{s}}_t$ of the global state $\boldsymbol{s}_t$, we introduce a *state reconstruction loss* $\mathcal{L}_{\text{state}}\left(\hat{\boldsymbol{s}}_t, \boldsymbol{s}_t\right)$, which is essentially a distance measure between $\hat{\boldsymbol{s}}_t$ and $\boldsymbol{s}_t$. Technically, since the observation is agent-centric, directly using it to infer the global state often leads to inconsistencies. Therefore, we decompose both the true and reconstructed states following the same structure as the agent observations: $\boldsymbol{s}^i_t = \left(\boldsymbol{p}^i_t;\ \boldsymbol{v}^i_t\right)$, $\hat{\boldsymbol{s}}^i_t = \left(\hat{\boldsymbol{p}}^i_t;\ \hat{\boldsymbol{v}}^i_t\right)$, where $\boldsymbol{p}^i_t$ (resp. $\hat{\boldsymbol{p}}^i_t$) denotes the relative position of entity i, and $\boldsymbol{v}^i_t$ (resp. $\hat{\boldsymbol{v}}^i_t$) represents its feature vector. Then, we define two losses to measure the difference between $\boldsymbol{s}^i_t$'s and $\hat{\boldsymbol{s}}^i_t$'s:

$$\mathcal{L}_{\text{pos}} = \frac{1}{n+m}\sum_{i=1}^{n+m}\sum_{t=1}^{T}\left\|\left(\boldsymbol{p}^i_t - \boldsymbol{c}_t\right) - \left(\hat{\boldsymbol{p}}^i_t - \hat{\boldsymbol{c}}_t\right)\right\|_2^2,$$

$$\mathcal{L}_{\text{attr}} = \frac{1}{n+m}\sum_{i=1}^{n+m}\sum_{t=1}^{T}\left\|\boldsymbol{v}^i_t - \hat{\boldsymbol{v}}^i_t\right\|_2^2,$$

where $\boldsymbol{c}_t$ is the average of $\boldsymbol{p}^i_t$ over $i \in \mathcal{N}$ and $\hat{\boldsymbol{c}}_t$ defined analogously. The overall loss for reconstructing the state is defined as the sum of the above losses: $\mathcal{L}_{\text{state}} = \mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{attr}}$.

Overall Loss. Together with state reconstruction loss, the overall learning objective of our proposed framework is

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_{\text{TD}}(\boldsymbol{\theta}) + \eta\mathcal{L}_{\text{state}},$$

where $\mathcal{L}_{\text{TD}}(\boldsymbol{\theta})$ is the standard temporal difference loss used in CTDE and η is a tunable hyperparameter that controls the weight of the state reconstruction loss.

5 Experiments

5.1 Experimental Settings

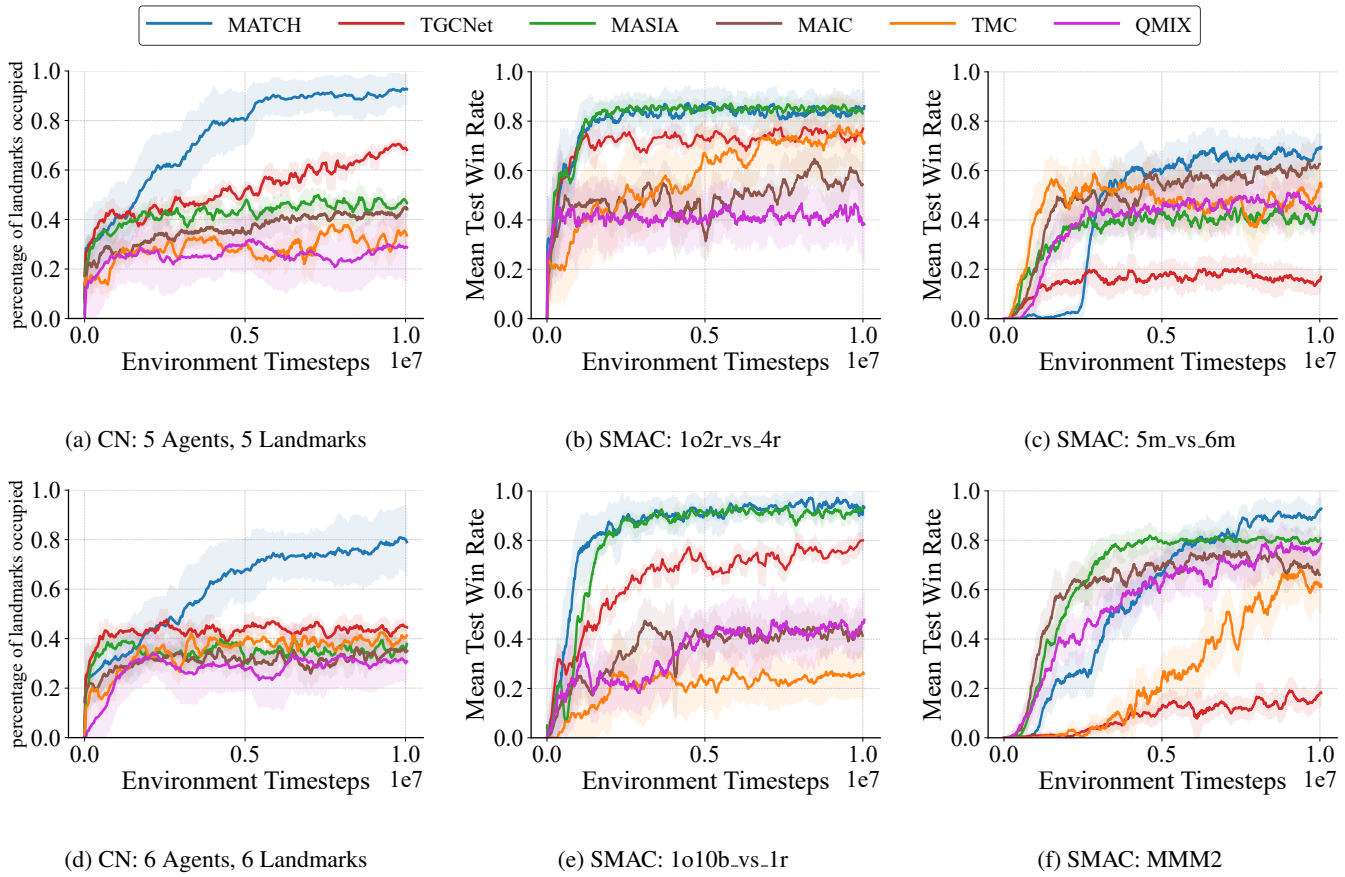
In this section, we conduct experiments on various benchmarks with different communication conditions to evaluate our method. Specifically, we aim to answer the following questions in this section: 1) How does our method perform when compared with multiple baselines in various scenarios (Section 5.2)? 2) Does our approach effectively reduce communication overhead while maintaining coordination performance (Section 5.3)? 3) How robust is it under varying levels of message loss in the communication channel (Section 5.4)? We compare MATCH against a variety of baselines, including a non-communication method and several CTDE-based communication methods. QMIX (Hu et al. 2021) serves as a widely adopted non-communication baseline, achieving strong performance on diverse multi-agent benchmarks. TMC (Zhang, Zhang, and Lin 2020) reduces communication overhead by allowing agents to broadcast messages only when their content changes significantly. MAIC (Yuan et al. 2022) enhances coordination by letting agents model their teammates and broadcast incentive messages to guide Q-value updates. MASIA (Guan et al. 2024) improves communication by broadcasting local observations and applying self-supervised objectives to maintain compact and informative representations. TGCNet (Zhang et al. 2025) constructs a dynamic directed communication graph via a multi-key gated mechanism, and leverages this structure to reduce reliance on global state information.

We evaluate MATCH against multiple state-of-the-art baselines on two tasks: Cooperative Navigation (CN) in Multi-agent Particle Environment (MPE) (Lowe et al. 2017) and SMAC (Samvelyan et al. 2019). The Cooperative Navigation task involves N agents and N landmarks, where each agent aims to reach a landmark while avoiding collisions with others. In SMAC, we select maps that require communication, including 1o2r_vs_4r and 1o10b_vs_1r (Wang et al. 2019), as well as the hard map 5m_vs_6m and the super hard map MMM2. We further increase the difficulty of coordination by reducing the agents' sight range from 9 to 2. For fair evaluation, all experiments are conducted with five random seeds, and the results are reported as means with 95% confidence intervals.

5.2 Communication Performance

We first compare MATCH against multiple baselines to evaluate its communication performance across a range of benchmarks. In the CN task, we adopt the percentage of landmarks occupied at the end of an episode (POL) as our evaluation metric (Gallici, Martin, and Masmitja 2023), where a landmark is considered occupied if an agent is within a distance of 0.3 units. As shown in Figure 2a and Figure 2d, MATCH significantly outperforms all baseline methods across the evaluated scenarios. This highlights the effectiveness of decomposing agent and landmark information into entity representations and capturing the structure.

For scenarios requiring communication in SMAC (Figure 2b and Figure 2e), methods lacking communication capabilities, such as QMIX, fail. Other communication-based baselines, such as TGCNet, MAIC, and TMC, struggle to effectively integrate messages, which leads to degraded performance. In contrast, MATCH and MASIA achieve superior performance thanks to their strong state reconstruction capabilities. On the hard and super hard maps (Figure 2c and Figure 2f), under reduced sight setting, the communication-free QMIX performs poorly. Other communication-based approaches fail to perform well as they transmit overly redundant information. In contrast, our method achieves superior performance by leveraging a query-driven and distance-aware attention mechanism, which enables personalized and relevant information acquisition for each agent.



(a) CN: 5 Agents, 5 Landmarks (b) SMAC: 1o2r_vs_4r (c) SMAC: 5m_vs_6m

(d) CN: 6 Agents, 6 Landmarks (e) SMAC: 1o10b_vs_1r (f) SMAC: MMM2

Figure 2: Performance comparison with baselines on CN and SMAC benchmarks.

5.3 Communication Overhead

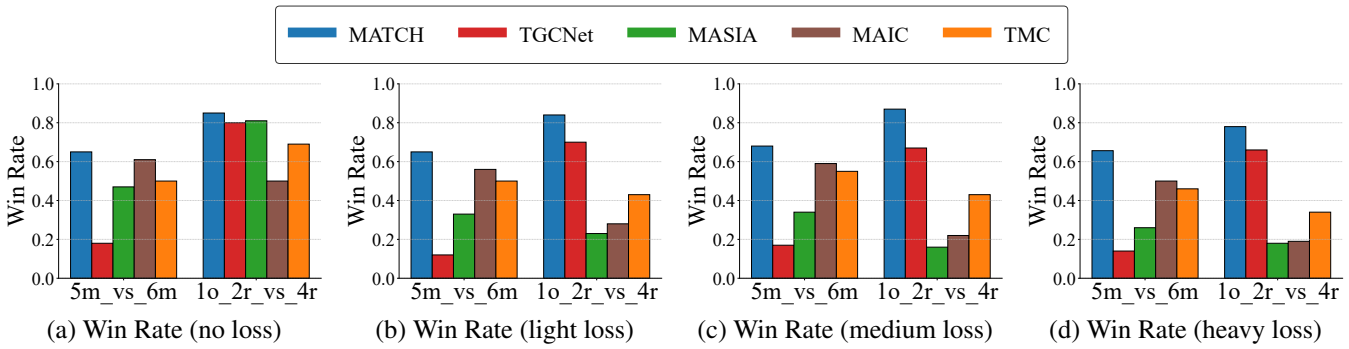
We now evaluate the communication overhead of MATCH by comparing it against all communication-based baselines, excluding the non-communicative QMIX, under the SMAC super hard map MMM2. Specifically, for each method, we run 100 test episodes among 5 random seeds.

We quantify communication overhead using two metrics: *communication frequency* and *communication volume*. Let x_t denote the number of directed communications that occur at timestep $t \in \mathcal{T}$, and let m_t represent the transmitted message. The communication frequency is defined as: $\frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} x_t$, and the communication volume is defined as: $\frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} x_t \cdot \|m_t\|$, where $|\mathcal{T}|$ is the total number of timesteps, and $\|m_t\|$ denotes the dimensionality of the message at timestep t.

Thanks to its centralized communication design and selective transmission of visible entities, MATCH achieves significantly lower communication overhead in both frequency and volume as shown in Figure 4. Compared with TGCNet and MASIA, MATCH reduces the communication volume by factors of 12.9× and 9.8×, respectively. Although MAIC and TMC achieve substantial communication volume reduction by transmitting low-dimensional modified Q-values, they achieve relatively less promising win rates. Moreover, our method exhibits the lowest communication frequency compared to all baseline approaches.

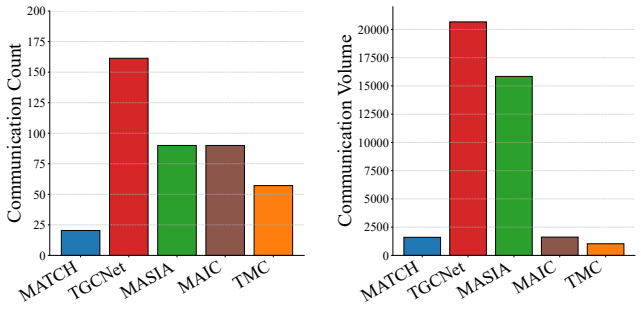
5.4 Performance Under Transmission Loss

Next we evaluate the algorithm's performance under communication loss. Specifically, for each method, we run 100 test episodes among 5 random seeds under three loss patterns corresponding to light, medium, and heavy levels, which are reasonable for wireless loss conditions (Zhang, Zhang, and Lin 2020; Sheth et al. 2007; Xylomenos and Polyzos 1999). Figure 3 depicts the win rates of MATCH, TGCNet, MASIA, MAIC, and TMC under three loss patterns across two SMAC maps, with all methods trained for 10M timesteps. The win rate of MASIA drops significantly, as it relies on receiving complete information from all agents to reconstruct state. TGCNet and MAIC exhibit higher robustness than MASIA, due to their use of dynamic communication topology and teammate modeling, respectively. TMC exhibits more robust communication performance owing to its mechanism of caching recently received messages. Under all transmission loss settings, MATCH consistently outperforms all baseline methods by a significant margin, with its winning rate remaining nearly stable. This demonstrates that our approach can effectively reconstruct the global state even in the presence of communication loss.



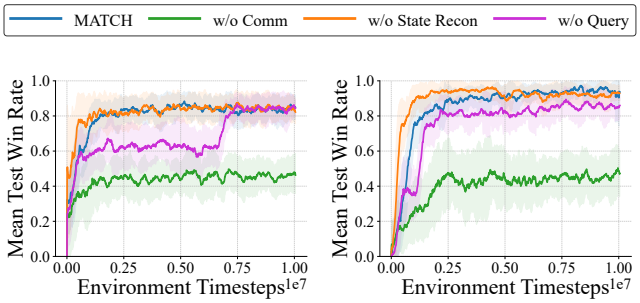
(a) Win Rate (no loss) (b) Win Rate (light loss) (c) Win Rate (medium loss) (d) Win Rate (heavy loss)

Figure 3: Win rate comparisons under different levels of message loss.



(a) Communication Count (b) Communication Volume

Figure 4: Comparison of communication frequency and volume across methods on the SMAC super-hard scenario.



(a) 1o2r_vs_4r (b) 1o10b_vs_1r

Figure 5: Ablation study results of MATCH w/o Comm, MATCH w/o State Recon, and MATCH w/o Query.

5.5 Ablation Studies

To understand the superior performance of MATCH, we conduct ablation studies to evaluate the contribution of its three main components, addressing the following questions: (1) Is communication through the hub indeed necessary for effective coordination? (2) Can the spatiotemporal reconstruction module accurately approximate the global state? (3) Does the Query mechanism significantly contribute to improved agent-specific message extraction?

To investigate the first question, we construct a variant called MATCH w/o Comm, which disables communication by removing the message passing between agents and the hub, thereby isolating the effect of centralized communication. To address the second question, we develop MATCH w/o State Recon, which replaces the spatiotemporal reconstruction module with access to the true global state. For the third question, we introduce MATCH w/o Query, which removes the query mechanism and instead broadcasts the same shared message to all agents.

We conduct ablation experiments on the communication-based SMAC scenarios. As shown in Figure 5, MATCH w/o Comm suffers a noticeable drop in performance, verifying the importance of communication. MATCH performs similarly to MATCH w/o State Recon, indicating that the spatiotemporal reconstruction module can effectively approximate the true global state. Finally, MATCH w/o Query underperforms due to all agents receiving identical messages, highlighting the importance of tailoring information to each agent's needs.

6 Conclusion and Future Work

In this work, we propose MATCH, a method that integrates a centralized information hub into the CTDE framework to facilitate more effective agent cooperation. Existing methods often overlook real-world communication environments, which are typically bandwidth-limited and imperfectly reliable. Our method collects messages centrally and reconstructs a global state through spatiotemporal modeling. After that, we further introduce a query mechanism allowing agents to selectively access information most relevant to their decision-making. Empirical results demonstrate that our method delivers strong performance, significantly reduces communication overhead, and remains highly robust under various levels of message loss. In the future, we aim to extend MATCH to large-scale multi-agent environments with hundreds of agents. This raises new challenges for communication scalability and capacity. Techniques such as hierarchical coordination and further exploration of scalable communication architectures are promising directions.

Acknowledgments

The authors would like to acknowledge the support from National Key R&D Program of China (No. 2024YFE0111800), the Science and Technology Development Fund (SKL-IOTSC(UM)-2024-2026) of Macau, and the State Key Laboratory of Internet of Things for Smart City (University of Macau) (Ref. No.: SKL-IoTSC(UM)-2024-2026/ORP/GA05/2023).

References

Bie, Y.; Ji, Y.; and Ma, D. 2024. Multi-agent deep reinforcement learning collaborative traffic signal control method considering intersection heterogeneity. *Transportation Research Part C: Emerging Technologies*, 164: 104663.

Chen, Y.; Mao, H.; Mao, J.; Wu, S.; Zhang, T.; Zhang, B.; Yang, W.; and Chang, H. 2022. PTDE: Personalized training with distilled execution for multi-agent reinforcement learning. *arXiv preprint arXiv:2210.08872*.

Das, A.; Gervet, T.; Romoff, J.; Batra, D.; Parikh, D.; Rabbat, M.; and Pineau, J. 2019. Tarmac: Targeted multi-agent communication. In *International Conference on machine learning*, 1538–1546. PMLR.

Foerster, J.; Assael, I. A.; De Freitas, N.; and Whiteson, S. 2016. Learning to communicate with deep multi-agent reinforcement learning. *Advances in neural information processing systems*, 29.

Foerster, J.; Farquhar, G.; Afouras, T.; Nardelli, N.; and Whiteson, S. 2018. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Gallici, M.; Martin, M.; and Masmitja, I. 2023. TransfQMix: Transformers for leveraging the graph structure of multi-agent reinforcement learning problems. *arXiv preprint arXiv:2301.05334*.

Guan, C.; Chen, F.; Yuan, L.; Zhang, Z.; and Yu, Y. 2024. Efficient Communication via Self-Supervised Information Aggregation for Online and Offline Multiagent Reinforcement Learning. *IEEE Transactions on Neural Networks and Learning Systems*.

Guo, X.; Shi, D.; and Fan, W. 2023. Scalable communication for multi-agent reinforcement learning via transformer-based email mechanism. *arXiv preprint arXiv:2301.01919*.

Hu, J.; Jiang, S.; Harding, S. A.; Wu, H.; and Liao, S.-w. 2021. Rethinking the implementation tricks and monotonicity constraint in cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2102.03479*.

Jiang, J.; Dun, C.; Huang, T.; and Lu, Z. 2018. Graph convolutional reinforcement learning. *arXiv preprint arXiv:1810.09202*.

Jiang, J.; and Lu, Z. 2018. Learning attentional communication for multi-agent cooperation. *Advances in neural information processing systems*, 31.

Jianye, H.; Hao, X.; Mao, H.; Wang, W.; Yang, Y.; Li, D.; Zheng, Y.; and Wang, Z. 2022. Boosting multiagent reinforcement learning via permutation invariant and permutation equivariant networks. In *The eleventh international conference on learning representations*.

Kraemer, L.; and Banerjee, B. 2016. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing*, 190: 82–94.

Liang, T.-C. 2021. Parallel droplet control in MEDA biochips using multi-agent reinforcement learning. In *International Conference on Machine Learning*.

Liu, B.; Liu, Q.; Stone, P.; Garg, A.; Zhu, Y.; and Anandkumar, A. 2021. Coach-player multi-agent reinforcement learning for dynamic team composition. In *International Conference on Machine Learning*, 6860–6870. PMLR.

Liu, Y.; Wang, W.; Hu, Y.; Hao, J.; Chen, X.; and Gao, Y. 2020. Multi-agent game abstraction via graph attention neural network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 7211–7218.

Lowe, R.; Wu, Y. I.; Tamar, A.; Harb, J.; Pieter Abbeel, O.; and Mordatch, I. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30.

Mao, H.; Zhang, Z.; Xiao, Z.; Gong, Z.; and Ni, Y. 2020. Learning agent communication under limited bandwidth by message pruning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 5142–5149.

Meng, X.; and Tan, Y. 2023. Learning group-level information integration in multi-agent communication. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, 2601–2603.

Meng, X.; and Tan, Y. 2024. Pmac: Personalized multi-agent communication. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17505–17513.

Niu, Y.; Paleja, R. R.; and Gombolay, M. C. 2021. Multi-Agent Graph-Attention Communication and Teaming. In *AAMAS*, volume 21, 20th.

Oliehoek, F. A.; Amato, C.; et al. 2016. *A concise introduction to decentralized POMDPs*, volume 1. Springer.

Rashid, T.; Samvelyan, M.; De Witt, C. S.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2020. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178): 1–51.

Samvelyan, M.; Rashid, T.; De Witt, C. S.; Farquhar, G.; Nardelli, N.; Rudner, T. G.; Hung, C.-M.; Torr, P. H.; Foerster, J.; and Whiteson, S. 2019. The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043*.

Sheth, A.; Nedevschi, S.; Patra, R.; Surana, S.; Brewer, E.; and Subramanian, L. 2007. Packet loss characterization in WiFi-based long distance networks. In *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*, 312–320. IEEE.

Singh, A.; Jain, T.; and Sukhbaatar, S. 2018. Learning when to communicate at scale in multiagent cooperative and competitive tasks. *arXiv preprint arXiv:1812.09755*.

Sukhbaatar, S.; Fergus, R.; et al. 2016. Learning multiagent communication with backpropagation. *Advances in neural information processing systems*, 29.

Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W. M.; Zambaldi, V.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo,

J. Z.; Tuyls, K.; et al. 2017. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*.

Wang, J.; Ren, Z.; Liu, T.; Yu, Y.; and Zhang, C. 2020. Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062*.

Wang, J.; Xu, W.; Gu, Y.; Song, W.; and Green, T. C. 2021. Multi-agent reinforcement learning for active voltage control on power distribution networks. *Advances in Neural Information Processing Systems*, 34: 3271–3284.

Wang, T.; Wang, J.; Zheng, C.; and Zhang, C. 2019. Learning nearly decomposable value functions via communication minimization. *arXiv preprint arXiv:1910.05366*.

Wang, Y.; and Sartoretti, G. 2022. Fcmnet: Full communication memory net for team-level cooperation in multi-agent systems. *arXiv preprint arXiv:2201.11994*.

Xue, K.; Xu, J.; Yuan, L.; Li, M.; Qian, C.; Zhang, Z.; and Yu, Y. 2022. Multi-agent dynamic algorithm configuration. *Advances in Neural Information Processing Systems*, 35: 20147–20161.

Xylomenos, G.; and Polyzos, G. C. 1999. TCP and UDP performance over a wireless LAN. In *IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)*, volume 2, 439–446. IEEE.

Yu, C.; Velu, A.; Vinitsky, E.; Gao, J.; Wang, Y.; Bayen, A.; and Wu, Y. 2022. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in neural information processing systems*, 35: 24611–24624.

Yuan, L.; Wang, J.; Zhang, F.; Wang, C.; Zhang, Z.; Yu, Y.; and Zhang, C. 2022. Multi-agent incentive communication via decentralized teammate modeling. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 9466–9474.

Zhang, S. Q.; Zhang, Q.; and Lin, J. 2020. Succinct and robust multi-agent communication with temporal message control. *Advances in neural information processing systems*, 33: 17271–17282.

Zhang, Z.; He, B.; Cheng, B.; and Li, G. 2025. Bridging training and execution via dynamic directed graph-based communication in cooperative multi-agent systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 23395–23403.