DOI:
10.1007/s10458-024-09665-6

Team-wise Effective Communication in Multi-Agent Reinforcement Learning

Ming Yang[1,2], Kaiyan Zhao[1,4], Yiming Wang[1,2], Renzhi Dong[1], Yali Du[5], Furui Liu[6], Mingliang Zhou[7], Leong Hou U[1,2,3*]

[1]SKL of Internet of Things for Smart City, University of Macau, Macao, China.
[2]Department of Computer Information Science, University of Macau, Macao, China.
[3]Centre for Data Science, University of Macau, Macao,China.
[4]School of Computer Science, Wuhan University, Wuhan, China.
[5] King's College London, London, UK.
[6]Zhejiang Lab, Hangzhou, China.
[7]College of Computer Science, Chongqing University, Chongqing, China.

*Corresponding author(s). E-mail(s): ryanlhu@um.edu.mo;

Abstract

Effective communication is crucial for the success of multi-agent systems, as it promotes collaboration for attaining joint objectives and enhances competitive efforts towards individual goals. In the context of multi-agent reinforcement learning, determining "whom", "how" and "what" to communicate are crucial factors for developing effective policies. Therefore, we propose TeamComm, a novel framework for multi-agent communication reinforcement learning. First, it introduces a dynamic team reasoning policy, allowing agents to dynamically form teams and adapt their communication partners based on task requirements and environment states in cooperative or competitive scenarios. Second, TeamComm utilizes heterogeneous communication channels consisting of intra- and inter-team to achieve diverse information flow. Lastly, TeamComm leverages the information bottleneck principle to optimize communication content, guiding agents to convey relevant and valuable information. Through experimental evaluations on three popular environments with seven different scenarios, we empirically demonstrate the superior performance of TeamComm compared to existing methods.

Keywords: Reinforcement Learning, Multi-agent System, Communication, Cooperation, Competition

1 Introduction

Citizen-centric AI systems are critical for addressing societal challenges, with Multi-Agent Reinforcement Learning (MARL) playing a key role in developing intelligent, collaborative solutions. In sectors like smart transportation and disaster response, MARL is vital for understanding and meeting the diverse needs of citizens [1]. Effective communication among agents is crucial, particularly in partially observable settings, to ensure policies learned are effective and aligned with citizen-centric goals [2]. Consequently, several key questions emerge regarding multi-agent communication: whom agents should communicate with, how message passing should be processed, and what information should be exchanged. These questions constitute critical factors in the development of effective MARL policies.

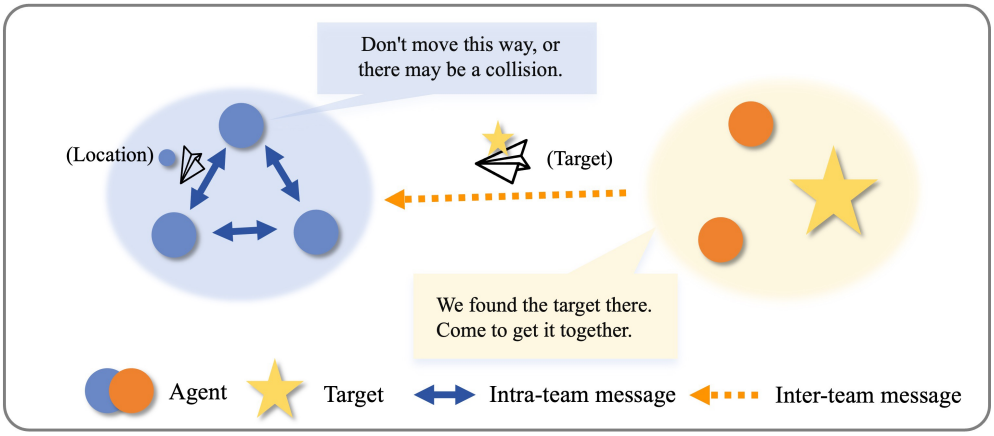


Fig. 1 This figure depicts a multi-agent scenario where the team-wise communication structure and heterogeneous communication strategy enable the agents to accomplish the task.

Previous research on multi-agent communication has recognized the importance of the aforementioned three questions from various perspectives. Existing approaches for determining "whom" to communicate with rely on communication topology to select linked agents. Some choose static topologies, such as fully connected networks (DIAL[3]), star topology (CommNet[4]), and tree topology (ATOC[5]), while others learn dynamic topologies, such as ([6–10]). However, learning novel and dynamic communication topologies presents a significant challenge, as the space of possible topologies grows exponentially with the number of agents. With n agents in the environment, the number of potential topologies can reach up to $2^{n(n-1)}$, severely limiting the quality of the learned topology. Regarding "how" to communicate, the common approach is to use an isomorphism channel, where all messages are passed using the same way. TieComm ([11]) tackles this challenge by leveraging tie theory to learn a hierarchical communication structure. For "what" to communicate, approaches either use raw observations of the communication targets or utilize attention mechanisms to focus on weighted content.

In this work, we propose TeamComm, a novel hierarchical communication protocol that determines "whom", "how", and "what" to communicate through an end-to-end framework.

First, we propose learning a team assignment matrix that dynamically assigns agents to teams, forming a team-wise communication structure that enables agents to communicate with their teammates and other teams. As illustrated in Fig. 1, individuals naturally tend to form teams and engage in communication within and between these teams to facilitate collaboration and achieve shared objectives in cooperative tasks. Furthermore, sociological studies [12] have observed that individuals naturally belong to communities and can acquire novel information from other communities for individual goals. Therefore, the team-wise structure can serve as a potentially effective communication structure in a multi-agent system.

Moreover, we construct heterogeneous communication channels consisting of inter- and intra-team message passing, inspired by the Tie Theory, a well-established concept in social network analysis and mathematical sociology [13]. Tie Theory highlights the significance of recognizing the differences in information exchanged within teams and between teams, underscoring the importance of incorporating diverse communication channels [14]. Given the dynamic formation of teams and the potential variation in the number of teams and members over time, we propose a team pooling method to ensure permutation-equivariant and permutation-invariant.

Lastly, to enhance effective communication, we aim to increase the conciseness of received messages by applying the information bottleneck principle in our team-wise communication structure. By leveraging this approach, we aim to pass messages that contain valuable and informative content while minimizing unnecessary redundancy or noise.

To validate the effectiveness of our proposed method, we conducted experiments on three popular benchmark environments: Cooperative Navigation, Predator Prey, and Traffic Junction. The results demonstrate that TeamComm outperforms existing state-of-the-art methods, thereby verifying the effectiveness of our approach. Our contribution can be summarized in three main aspects:

- We propose a novel framework to learn a dynamic team-wise communication structure along with its associated heterogeneous communication channels. This framework consists of intra-team and inter-team message passing, enabling effective communication within and between teams.
- We integrate the information bottleneck principle into our communication framework with the aim of encouraging concise, relevant, and informative communication. This approach strives to improve the effectiveness of the overall communication process by prioritizing the transmission of valuable information.
- We propose an iterative learning framework that concurrently updates both the dynamic team reasoning policy and conditional action policy in both cooperative and competitive tasks.

2 Related Work

In multi-agent reinforcement learning, the environment can be non-stationary due to the agents considering other agents as part of the environment dynamics. Therefore, communication becomes crucial for agents to capture meaningful information and make informed decisions, especially when the environment offers only limited observations. Unlike some MARL studies [15] that explore communication methods in model-based settings, where the dynamics of the environment are known, our work focuses on the model-free setting. Based on the communication mechanism, existing model-free MARL methods can be classified into three groups.

2.1 MARL without Communication

Some studies utilize a centralized value estimation network to stabilize training and learn a decentralized execution policy without explicit communication channels. Representative works in this category include COMA [16], MADDPG [17], LIIR [18], and GridNet [19]. These methods extend variants of actor-critic algorithms to multi-agent settings, addressing the problem of credit assignment of policy gradient. Another set of works, based on Q-learning, includes VDN [20], QMIX [21], QTRAN [22], and MAVEN [23]. These methods aim to learn a value function that decomposes the global value into individual credits, enabling effective coordination among agents.

2.2 MARL with Static Communication

Some studies directly utilize static topology as a communication protocol based on prior knowledge. For instance, DIAL [3], [24], [25] [26] MAIC[27] select a fully connected graph, assuming agents need to communicate with all others. DIAL [3] uses one round of fixed-size communication, while [24] aims to minimize communication length and [25] adopts two rounds of communication to deliver personalized messages. [26] proposes the [26] proposed FCMNet and FCMTran, which are built upon Long Short-Term Memory and Transformer, to achieve diferentiable communication. [27] propose MAIC [27] firstly learns targeted agent models, with which each agent can anticipate the teammate's action selection and generate tailored messages to specific agents. TMC[28] enhances communication efficiency and robustness by minimizing message variation over time and enhancing confidence in action selection decisions. CommNet [4] adopts a star topology, wherein agents send messages to a virtual central agent, and average pooling is employed to filter the hidden state representations of all agents. ATOC [5] employs a tree topology, where certain agents act as initiators to aggregate and exchange messages from different groups of agents. Another set of methods, including [29–31], directly utilize the static graph of the environments, connecting agents with neighborhoods in networked systems (e.g., traffic signal, distributed sensing).

2.3 MARL with Dynamic Communication

Recent studies have focused on learning dynamic communication to determine agents with whom to communicate instead of using predefined communication structures.

Some notable methods in this category include: IC3Net [6]: Learns a hard binary gating mechanism to decide when to communicate. The agent broadcasts its information to all other agents when the gate is open. TarMAC [7]: Allows each individual agent to actively select which other agents to address messages to via a simple signature-based soft attention mechanism. SchedNet [32]: Exploits the attention mechanism to filter the necessary agents to communicate. GAComm [33]: Uses hard attention to decide who to communicate with and soft attention to allocate weights to the messages of the agents communicated with. LSC [34]: Considers hierarchical communication topology, considering only local geometrical relationships and the neighbor agent policy performance to establish groups. MAGIC [9]: Utilizes a Scheduler to generate a directed graph to decide when to communicate and to whom to address messages. A Message Processor using GATs is fed by the generated graph to integrate and process messages. a method attempts to learn a communication topology from zero and utilizes graph neutral network for communication. MAGIC [9] utilizes a scheduler to generate a directed graph, deciding when to communicate and to whom to address messages, and a message processor using graph neural networks to integrate and process messages. FlowComm [8]: Attempts to learn a dynamic directed graph for message passing priority. GCS [35]: Learns dynamic decision structures and allows actions to be passed as messages among agents. DGN [10] uses the spatial locations of agents as the dynamic graph and leverages homogeneous graph networks for communication. HetNet [36] uses heterogeneous graph attention networks to learn efficient and diverse communication models for coordinating cooperative heterogeneous teams. Tiecomm [11] utilizes tie theory to establish team-wise communication by learning a hierarchical communication structure based on prior knowledge. IMAC[37] proposes introducing an information bottleneck to enhance agent communication effectiveness under limited bandwidth constraints. MGAI [38] proposes a robust communication learning mechanism for multi-agent reinforcement learning (MARL) that leverages graph information bottleneck optimization to achieve optimal robustness and effectiveness.

Our proposed method, TeamComm, falls under the category of model-free MARL with dynamic communication. Unlike existing methods, such as IC3Net[6], TarMac[7], and FlowComm[8], which focus on a single aspect of communication, TeamComm takes a unified, end-to-end approach to address the "whom", "how", and "what" questions of communication. While we also leverage graphs for communication, unlike DGN[10], TieComm[11], and MGAI[11], which rely on prior knowledge of graphs, TeamComm has the ability to directly learn dynamic hierarchical graphs that adapt to environmental changes. Moreover, our method facilitates the transmission of effective communication content through heterogeneous communication channels.

3 Problem Setup

3.1 Dec-POMDP

In this manuscript, we study the problem of multi-agent communication in reinforcement learning with decentralized control. This problem can be formulated as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) [39], which is represented by a tuple $\{N, S, A^{tot}, O^{tot}, p, r, \gamma\}$. N denotes the set of agents,

indexed from 1 to n. S represents the set of possible global states where agents exist at each time step. $O^{tot} := \left\{O^i\right\}_{i=1}^n$ denotes the observation space for the agents. Please note that it is common to utilize observation O^{tot} as an approximation for the underlying true state S in practice. $A^{tot} := \left\{A^i\right\}_{i=1}^n$ refers to the action space available to the agents. $p\left(s_{t+1} \mid s_t, a_t^{tot}\right) : S \times A^{tot} \times S \to \mathbb{R}^+$ represents the probability distribution of state transition given the current global state s_t and joint action $a_t^{tot} := \left\{a_t^i\right\}_{i=1}^n$. $r^i\left(s_t, a_t^{tot}\right) : S \times A^{tot} \to \mathbb{R}$ indicates the reward function provided by the environment for agent i. The discount factor $\gamma \in [0, 1)$ is used to discount future rewards.

During an episode in a multi-agent communication system, agents make decisions by aggregating messages $\hat{o}^i$ from their own local partial observations $o^i \in O^i$ and received messages m^i, using the communication function $\Psi(\hat{o}^i \mid o^{tot})$. At each time step t, each agent selects an action $a_t^i \in A^i$ according to its individual stochastic policy $\pi^i\left(a_t^i \mid \hat{o}_t^i\right)$, thereby forming a joint action a_t^{tot} and The joint policy $\pi^{tot} := \left\{\pi^i\right\}_{i=1}^n$. The system then transitions randomly to the next state $s_{t+1} \in S$ following the state transition probability distribution $p\left(s_{t+1} \mid s_t, a_t^{tot}\right)$. Then, each agent receives an individual reward r^i based on the reward function $r\left(s_t, a_t^{tot}\right)$.

3.2 Competitive or Cooperative task

In a general-sum competitive task[40], each individual agent i, operates in a decentralized manner with the primary objective of maximizing its own cumulative discounted reward R^i, denoted as

$$\eta^i\left(\pi^{tot}\right) = \mathbb{E}\left[R_t^i\right] = \mathbb{E}_{\pi^{tot}} \sum_{l=0}^{\infty} \gamma^l r^i\left(s_{t+l}, a_{t+l}^{tot}\right) \tag{1}$$

Consequently, the state-action Q-function and the value function for each agent are defined as :

$$Q_{\pi^{tot}}^i(s, a^{tot}) = \mathbb{E}\left[R_t^i \mid s_t = s, a_t = a^{tot}\right] \tag{2}$$

and

$$V_{\pi^{tot}}^i(S) = \mathbb{E}\left[R_t^i \mid s_t = s\right] \tag{3}$$

respectively.

The cooperative task focuses on encouraging agents to collaborate and work together toward achieving a common goal. In this task, the primary goal is to maximize the global cumulative discounted reward R^{tot}, denoted as:

$$\eta\left(\pi^{tot}\right) = \mathbb{E}\left[\sum_{i=1}^{n} R_t^i\right] = \mathbb{E} \sum_{i=1}^{n} \sum_{l=0}^{\infty} \gamma^l r^i\left(s_{t+l}, a_{t+l}^{tot}\right) \tag{4}$$

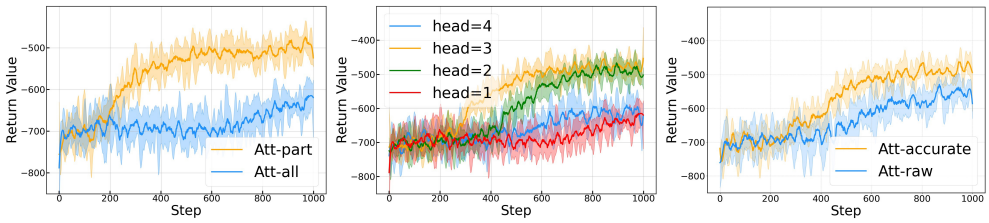
Correspondingly, the global state-action Q-function and global state value function are defined as $Q^{tot} = \sum_{i=1}^n Q^i$ and $V^{tot} = \sum_{i=1}^n V^i$.

4 Methodology

4.1 Investigating the Communication Mechanism of Agents

In the context of multi-agent reinforcement learning, constructing effective communication entails a comprehensive exploration of three key questions: "whom" to communicate with, "how" to communicate, and "what" information to convey. To study these three aspects, we introduce a naïve method, which employs attention weights to determine the exchange of messages, enabling selective and adaptive communication based on the relevance and importance of the information shared. It is a widely used approach in existing communication studies ([8–10, 33]). The following toy experiments are designed with certain prior knowledge in place, and our primary objective was to isolate and confirm the significance of specific factors in a controlled experiment setting.

We conducted experiments on a modified version of the Cooperative Navigation (CN) environment, where agents operate under partial observability and require communication to achieve better performance. Specifically, agents in this environment need to navigate towards landmarks matching their colors but lack direct perception of their own and other agents' colors. They can only observe their own position, velocity, and the relative positions of the landmarks. Agents are rewarded for proximity to the correct landmarks and penalized for collisions. Thus, effective communication plays a vital role in navigating towards the correct landmarks and avoiding collisions.



(a) Exp on "whom" (b) Exp on "how" (c) Exp on "what"

Fig. 2 Experiment Results: All experiments were conducted using 3 random seeds, and the shaded regions represent the standard deviation. Additional results in various settings can be accessed in the public repository.

Firstly, we investigate whom to communicate with. In our exploration, we conducted experiments using two approaches: "Att-all" where agents communicate with all other agents, and "Att-part" where agents selectively communicate with a subset of agents containing relevant landmark information. Results in Fig.2(a) show that global communication performs worse compared to partial communication. This indicates that unnecessary communication introduces noise, potentially impacting the agent policy's performance adversely. Selectively communicating with relevant subsets of agents can enhance the effectiveness of communication, leading to improved overall performance.

Secondly, we investigate how to communicate. In our experiments, we varied the number of attention heads for communication channels. The results, as depicted in the Fig.2(b), indicate that a single communication channel is not sufficient, and having too many channels does not necessarily improve effectiveness. This finding aligns with research in natural language processing [41]. The article also highlights that simply increasing the number of heads may not enable effective focusing on different content. Therefore, adopting heterogeneous channels shows the potential to enhance multi-agent communication.

Lastly, we investigate what to communicate. We conducted experiments using two approaches: "Att-raw" which communicated raw observations, and "Att-accurate" which only conveyed landmark information. Fig.2(c) showed that the "Att-raw" approach decreased communication effectiveness even when communicating with the correct agent, highlighting the importance of communicating precise and relevant content. Reducing redundant information is therefore crucial to enhance effective communication.

These toy experiments described above provide valuable insights into multi-agent communication. They highlight the importance of not only communicating with the correct agent but also conveying concrete content. Furthermore, our results imply that simply increasing the number of communication channels may not be effective. This is because merely adding more channels without proper separation or organization may not work. Instead, we hypothesize that heterogeneous communication channels could be a more promising approach. These findings can potentially contribute to the development of improved multi-agent communication strategies.

Therefore, we propose a team-wise communication structure with heterogeneous communication channels. As shown in Fig.3, agents are organized into teams and engage in two levels of communication. Firstly, within each team, agents communicate to exchange information and coordinate their actions. Secondly, between teams, agents selectively exchange novel information while focusing on their own team's objectives. In an ideal scenario, if there are n agents in the environment and they are evenly grouped, each agent only needs to pay attention to $\sqrt{n}$ agents and $\sqrt{n}$ teams. This significantly reduces the scope of attention required. While our method has the capability to accommodate multiple hierarchical levels as the number of agents increases, this manuscript primarily focuses on demonstrating the effectiveness of utilizing two levels of hierarchy.

We represent the hierarchical structure of our method as $c := (c_{\text{intra}}, c_{\text{inter}})$, where c_{intra} denotes the communication topology within each team, and c_{inter} represents the communication topology between teams. During an episode, each agent i should utilize not only its own local observation o^i but also the received messages m^i, consisting of messages from both intra-team m^i_{intra} and inter-team m^i_{inter}, to select an action a^i.

To achieve this, we propose the communication function $\Psi(\hat{o}^i \mid o^{tot}, c)$ as follows:

$$\Psi(\hat{o}^i \mid o^{tot}, c) := o^i \cup m^i(o^{tot}, c) \tag{5}$$

where $m^i(o^{tot}, c) = m^i_{\text{intra}}(o^{tot}, c_{\text{intra}}) \cup m^i_{\text{inter}}(o^{tot}, c_{\text{inter}})$.

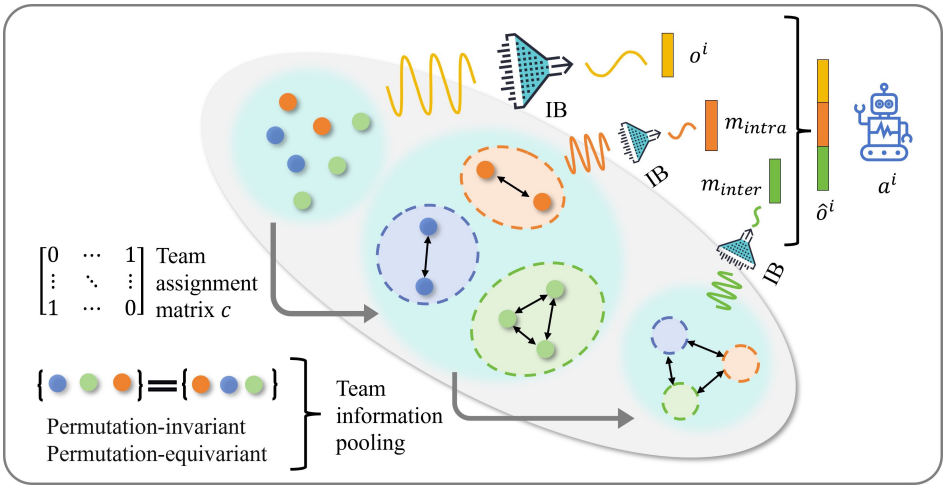


Fig. 3 Framework of TeamComm: Agents are grouped by a team assignment matrix. We then perform team information pooling to generate higher-level messages, ensuring permutation-invariance and permutation-equivariance. The agent selects actions based on multi-level information, including intra- and inter-messages, after applying information bottleneck (IB) filtering.

In the subsequent section, we initially generate a hierarchical structure c to group agents, in order to determine whom to communicate with. Next, we introduce heterogeneous message channels to address how to communicate. Then, we propose using an information bottleneck to tackle what to communicate effectively. Finally, we present the practical implementation and pseudo-code of our method.

4.2 Learning Dynamic Teaming

In this section, we aim to develop a team detection method that relies solely on the local features of individual agents. The observation of the homophily property in real-world networks indicates that agent features can serve as a reliable initialization for the clustering process [42, 43]. Hence, our approach focuses on learning team detection directly from the raw global observations o^{tot}, without any prior knowledge about the network topology, such as the approach used in TieComm [11].

To be specific, we employ an agent team assignment matrix $c \in \{0,1\}^{n\times n}$ to represent the communication structure. This matrix indicates whether agent i belongs to the team j through $c^{i,j}$. From this matrix, we can derive specific agent sets c_{intra} and team sets c_{inter}, capturing communication topology within and between teams, respectively. We propose a team reasoning policy $\rho(c \mid o^{tot})$, a multi-layer perceptron (MLP) ρ with softmax activation on the output layer, parameterized by ϱ, to reason about c:

$$c \sim \rho(o^{tot}; \varrho) \tag{6}$$

The team assignment matrix c is sampled from the probability distribution governed by ρ.

For the joint policy of multi-agent reinforcement learning, we need to modify it accordingly to adapt to the proposed hierarchical communication structure. The

joint policy π^{tot} of multi-agent system now is factorised into conditional action policy $\pi^i\left(a^i \mid \hat{o}^i;\theta\right)$ of each agent and a teaming reasoning policy $\rho(c \mid o^{tot})$:

$$\pi^{tot}(a^{tot}, c \mid o^{tot};\theta,\varrho) = \Pi_{i=1}^{n}\pi^i\left(a^i \mid \phi^i(o^{tot}, c);\theta\right)\rho(c \mid o^{tot};\varrho) \tag{7}$$

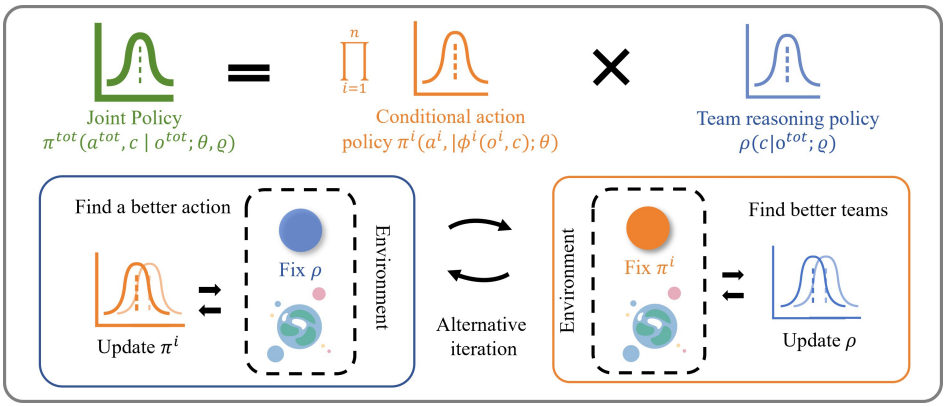


Fig. 4 Iterative Optimization: The joint policy is factorized into two kinds of policies. When we update one of the policies, the other policy remains fixed and is treated as a part of the environment.

As shown in Fig. 4, we iteratively optimize the two policies to achieve an optimal joint policy. In the following, we first introduce the loss of team reasoning policy. The optimization of conditional action policy will be introduced later.

MARL generally involves two scenarios: cooperative tasks and competitive tasks, each of which demands a distinct optimization approach for team reasoning policy.

4.2.1 Optimization in Cooperative Tasks

The objective in cooperative tasks is to find an optimal joint policy to maximize the global Q^{tot} value. By introducing new policy $\rho(c \mid \varrho)$, the new objective of the cooperative MARL problem is defined as:

$$\pi^{tot}(a^{tot}, c \mid o^{tot};\theta,\varrho) = \underset{\{\pi\}_{i=1}^{n},\rho}{\text{argmax}}\mathbb{E}_{\pi^{tot}\sim\theta,\varrho}Q^{tot}\left(o^{tot}, a^{tot}\right) \tag{8}$$

To optimize the parameter ϱ in the team reasoning policy ρ, we utilize the following proposition:

Proposition 1. *Given Eq.(7) and Eq.(8), the update rule for the team reasoning policy gradient in cooperative tasks can be devised as follows:*

$$\nabla_{\varrho}J(\rho) = \mathbb{E}_{s\sim p,c\sim\rho}\left[\nabla_{\varrho}\log\rho_{\varrho}(c \mid s)\sum_{i}^{n}\int_{A}\bar{\pi}\left(a^{tot} \mid s, c\right)Q^i(s, a^{tot})\,da^i\right] \tag{9}$$

where $\bar{\pi}\left(a^{tot} \mid s, c\right) = \prod_i^n \pi^i(a^i \mid s, c)$.

Proof. Following the single agent Policy Gradient Theorem [44, 45] and VDN framework [20] in a multi-agent cooperative task, we decompose joint objective function J (based on Eq.(8)) into a single policy concatenation (based on Eq.(7)), which is the integral of the expectation value of all the agents' the policy π^i and the action-value function Q^i under observation o^{tot}, action a^{tot}, and communication structure c:

$$\begin{aligned} J &= \int_{S,A,C} \pi^{tot}(a^{tot}, c \mid s) Q^{tot}(s, a^{tot})\, ds\, da^{tot}\, dc \\ &= \sum_i^n \int_S \int_C \rho(c) \int_A \bar{\pi}\left(a^{tot} \mid s, c\right) Q^i(s, a^{tot})\, da^i\ dc\, ds \end{aligned} \tag{10}$$

Please note that the uppercase letters (e.g. C) represent the corresponding spaces, while the lowercase letters (e.g. c) represent specific value within those spaces. Suppose $\rho(c)$ is parameterized by ϱ, $\bar{\pi}$ is parameterized by θ. Assuming that we fix the parameters θ in policy $\bar{\pi}$ to a constant value, we can then apply the gradient over ϱ.

$$\begin{aligned} \nabla_\varrho J(\rho) &= \sum_i^n \int_S \int_C \nabla_\varrho \rho_\varrho(c) \int_A \bar{\pi}\left(a^{tot} \mid s, c\right) Q^i(s, a^{tot})\, da^i\, dc\, ds \\ &= \mathbb{E}_{s\sim p, c\sim\rho}\left[\nabla_\varrho \log \rho_\varrho(c \mid s) \sum_i^n \int_A \bar{\pi}\left(a^{tot} \mid s, c\right) Q^i(s, a^{tot})\, da^i\right] \end{aligned} \tag{11}$$

It is important to note that in practice, we typically utilize the observation o^{tot} instead of the state s. Besides, due to the decentralized setting, we use local value to estimate global value and employ $Q^i(o^i, a^i)$ to estimate $Q^i(s, a^{tot})$. Finally, we utilize the reply buffer $\mathcal{D}$ to estimate expectations. Hence, Proposition 1 can be revised to in practice:

$$\nabla_\varrho \rho = \mathop{\mathbb{E}}_{(o^{tot}, a^{tot}, C)\sim\mathcal{D}}\left[\nabla_\varrho \ln \rho_\varrho(\varrho \mid o^{tot}, c) \sum_i^n Q^i(o^i, a^i)\right]$$

The poof completes. □

Proposition 1 indicates that the team reasoning policy should consider all possible actions of each agent to encourage cooperation among agents and improve their policies toward the optimal global return.

4.2.2 Optimization in Competitive Tasks

In cooperative tasks, a global Q-value Q^tot guides the optimization of the teaming reasoning policy. However, in competitive environments, this global Q-value is not available. Hence, we need to introduce a new metric to achieve a similar effect.

Definition 1 (Modularity [46]). *Modularity $\mathcal{M}$ is a commonly used metric for measuring the strength of team structure c, defined as follows:*

$$\mathcal{M}(c) = \frac{1}{2w} \sum_{ij} \left[e^{ij} - \frac{k^i k^j}{2w} \right] \delta\left(c^i, c^j\right) \tag{12}$$

where e^{ij} represents the edge weight between agent i and j, k^i and k^j are the degrees of nodes i and j respectively (i.e., the sums of the weights of the edges connected to each agent), w is the sum of all edge weights in the graph, c^i is the team which agent i belongs to and δ is the kronecker delta function which equals 1 if c^i is equal to c^j, and 0 otherwise.

The calculation of modularity $\mathcal{M}$ requires a graph, so we need firstly build a fully connected undirected weighted graph by computing the cosine similarity between agents:

$$e^{ij} = \frac{o^i \cdot o^j}{|o^i| \cdot |o^j|} \tag{13}$$

Therefore, given a global observation o^{tot} and a team assignment matrix c, we can calculate the corresponding modularity value $\mathcal{M}(o^{tot}, c)$.

Given Eq.(12), we propose the update rule for the team reasoning policy gradient in competitive tasks can be devised as follows:

$$\nabla_{\varrho} J(\rho) = \mathbb{E}_{o^{tot} \sim p, c \sim \rho} \left[\mathcal{M}(o^{tot}, c) \nabla \varrho \log \rho_{\varrho}(c \mid o^{tot}) \right] \tag{14}$$

Eq.(14) suggests that the team reasoning policy should be optimized for maximum modularity of each time step. This ensures that agents with similar information are grouped together on the same team, allowing intra-team message channels to propagate differences among teammates, while inter-team message channels can pass novel information. Consequently, our heterogeneous channels are able to maintain distinct roles and functions.

4.3 Heterogeneous Message Channels

In this section, we tackle the problem of how to communicate. We address this by exploring two key aspects: Firstly, we propose Team Information Pooling methods to generate higher-level expressions. Secondly, we design Differentiable Message Passing to facilitate information exchange across all levels.

4.3.1 Team Information Pooling

The challenge of team information pooling is the dynamic nature of teams, where the number of members can change as the environment state changes. This property requires an adaptive approach to handle varying team sizes and compositions effectively. TieComm ([11]) selects a representative agent from each team based on their highest tie strength within that team. It breaks gradient backpropagation during training, requires high complexity in argmax calculation, and cannot ensure lossless information transfer. This aligns with the findings of [47, 48], which

suggest that common aggregators (such as Mean, Maximal, Minimum, Standard deviation, and Normalized moments) may not effectively capture team information. To address the challenge of varying team sizes, we utilize the attention mechanism [49] to transform a variable-sized set into a fixed-size representation while maintaining permutation-invariant and permutation-equivalent [50].

Specifically, we assume the team set c^j has k agents, which has a raw observation o^k with size $1 \times d$ dimension: $c^j = \left\{o^1, o^2, \cdots, o^k\right\}, o^k \in \mathbb{R}^{1\times d}$. First, we leverage a shared Multi-Layer Perceptron (MLP) model to compress each raw observationo^k into 1×1 value and then apply softmax activation to compute a normalized attention score $c^w = \{w^1, w^2, \cdots, w^k\}$ with $\sum_1^k w^k = 1$. Next, these attention scores are then element-wise multiplied with their corresponding original features in team c, resulting in a new set of weighted features $c^j = \left\{h^1, h^2, \cdots, h^k\right\}$ with $h^k = o^k * w^k$. To obtain a fixed-size representation, we sum the weighted features across all elements, resulting in the final feature representation c^j. Here we abuse the notion c^i, it not only represents the set index j, it also represents the pooled representation of set j. In the above process, both the denominator and numerator are summations of permutation-equivariant terms, and the team observation value $o(c^j)$ remains invariant to different permutations of the team set c^j. Employing this approach allows us to effectively transform a variable-sized team feature set into a fixed-size representation while retaining important information through attention weights. This technique also enables the backpropagation of gradients during the optimization process.

4.3.2 Differentiable Message Passing

For message passing in different levels, we utilize attention directly to discern the desired information for communication. With a smaller number of agents in each group, there is no need to consider the structural properties of the graph as some previous studies required (e.g. [8, 9, 11]).

$$m_{\text{level}}^i = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in c_{\text{level}}}\alpha_{ij}^k \mathbf{W}^k o^{\text{level}}\right) \tag{15}$$

where k denotes the attention head, σ represents the activation function, α indicates the attention score, $\mathbf{W}$ is the weight matrix.c_{level} refers to corresponding level sets.

The intra-team message aims to pass the other teammates' observation within the same team, therefore $c_{\text{level}} = c_{\text{intra}}$. The inter-team message aims to transmit the pooled team representation information, therefore $c_{\text{level}} = c_{\text{inter}}$. Please note that while attention is utilized for message passing at both levels, the content being transmitted is different, thus resulting in heterogeneous channels.

4.4 Effective Communication with Information Bottleneck

In this section, our objective is to address the challenge of what to effective communication in MARL based on Information Bottleneck (IB). Then we introduce the optimization of conditional action policy for MARL.

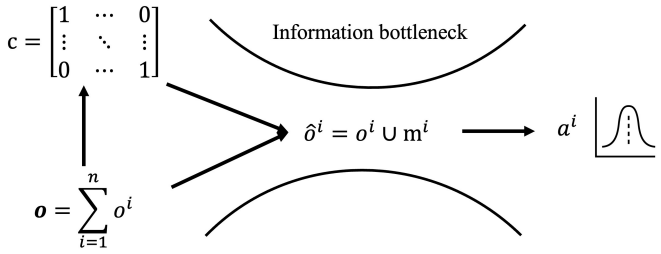


Fig. 5 This figure illustrates the information bottleneck in team-wise multi-agent information transmission, where the direction of information transmission can be regarded as a probabilistic model.

4.4.1 Information Bottleneck for Team-wise Communication

Our goal is to learn a message m^i that is maximally informative to choose actions a^i measured by $I(a^i; \hat{o}^i)$. Additionally, we aim to generate a message that extracts relevant and concrete information from the global observation o^{tot} and team-wise communication structure c, measured by $I(\hat{o}^i; o^{tot}, c)$.

To incorporate the IB principle into our team-wise multi-agent communication reinforcement learning, we first formulate the process of our team-wise communication as a probabilistic model, and the sequence is shown in Fig. 5. Hence, we can make the following assumption that:

$$\begin{aligned} p\left(a^i, \hat{o}^i, c, o^{tot}\right) &= p\left(\hat{o}^i \mid a^i, o^{tot}, c\right) p\left(a^i \mid o^{tot}, c\right) p(c \mid o^{tot}) p(o^{tot}) \\ &= p\left(\hat{o}^i \mid o^{tot}, c\right) p\left(a^i \mid o^{tot}, c\right) p(c \mid o^{tot}) p(o^{tot}) \end{aligned} \tag{16}$$

Based on above the assumption, we propose the following optimization:

$$p^\star(o^{\hat{tot}} \mid o^{tot}, c) = \arg \max_{p(o^{\hat{tot}} \mid (o^{tot}, c)} \left[I\left(o^{\hat{tot}}; a^{tot}\right) - \beta I(o^{\hat{tot}}; o^{tot}, c) \right] \tag{17}$$

where $\beta \geq 0$ is a Lagrange multiplier.

Considering that our problem involves MARL, we can utilize the joint policy function defined in Eq.(7) to factorize Eq.(17) for each agent i. Additionally, by applying the communication function described in Eq.(5), we can further factorize Eq.(17) into the message m^i, which serves as our optimization target. Consequently, Eq.(17) can be replaced with the following expression:

$$\begin{aligned} p^\star(\hat{o}^{tot} \mid o^{tot}, c) &\propto p^\star(m^{tot} \mid o^{tot}, c) \\ &= \sum_{i=1}^{n} p^{i,\star}(m^i \mid o^{tot}, c) \\ &\propto \sum_{i=1}^{n} \arg \max_{p^i(m^i \mid o^{tot}, c)} \left[I^i(\hat{o}^i; a^i) - \beta I^i(m^i; o^{tot}, c) \right] \end{aligned} \tag{18}$$

The first term $I^i\left(\hat{o}^i; a^i\right)$ encourages observation after communication $\hat{o}^i$ to be predictive of the corresponding action a^i, while the second term $I^i(m^i; (o^{tot}, c))$ encourages

the message m^i to "forget" irrelevant information from the global observation o^{tot} and communication structure c.

To estimate the $I(\hat{o}^i; a^i)$, we draw inspiration from [51]. Specifically, based on the fact that the Kullback Leibler divergence is always positive and the definition of mutual information, we have:

$$\begin{aligned} I^i(\hat{o}^i; a^i) &= \int_{\hat{O}^i} \int_{A^i} p(a^i, \hat{o}^i) \log \frac{p(a^i \mid \hat{o}^i)}{p(a^i)} \, da^i \, d\hat{o}^i \\ &\geq \int_{\hat{O}^i} \int_{A^i} p(a^i, \hat{o}^i) \log q(a^i \mid \hat{o}^i) \, da^i \, d\hat{o}^i + H(a^i) \end{aligned} \tag{19}$$

where $q(a^i \mid \hat{o}^i)$ be a variational approximation to $p(a^i \mid \hat{o}^i)$. The entropy of our true action $H(a^i)$ is independent of the optimization procedure and can be ignored. Leveraging Eq.(16), we can rewrite $p(a^i, \hat{o}^i)$ as

$$p\left(\hat{o}^i, a^i\right) = \int_C \int_{o^{tot}} p\left(\hat{o}^i \mid o^{tot}, c\right) \, p\left(a^i \mid o^{tot}, c\right) p(c \mid o^{tot}) p(o^{tot}) \, do^{tot} \, dc \tag{20}$$

By substituting Eq.(20) into Eq.(19), we obtain a new lower bound for the first term in our objective function.

$$\begin{aligned} I^i\left(\hat{o}^i; a^i\right) \geq \int_C \int_{O^{tot}} \int_{\hat{O}^i} \int_{A^i} & p\left(\hat{o}^i \mid o^{tot}, c\right) p\left(a^i \mid o^{tot}, c\right) p(c \mid o^{tot}) p(o^{tot}) \\ & \log q\left(a^i \mid \hat{o}^i\right) da^i \, d\hat{o}^i \, do^{tot} \, dc \end{aligned} \tag{21}$$

Similarly, for the second term, we have:

$$\begin{aligned} I^i\left(m^i; o^{tot}, c\right) &= \int_C \int_{O^{tot}} \int_{M^i} p(m^i, o^{tot}, c) \log \frac{p(m^i \mid o^{tot}, c)}{p(m^i)} \, dm^i \, do^{tot} \, dc \\ &= \int_C \int_{O^{tot}} \int_{M^i} p(m^i, o^{tot}, c) \log p(m^i \mid o^{tot}, c) \, dm^i \, do^{tot} \, dc \\ &\quad - \int_{M^i} p(m^i) \log p(m^i) \, dm^i \end{aligned} \tag{22}$$

where M^i is the space of message m^i.

Let $z(m^i)$ be the variational approximation to the marginal distribution of m^i. Since $\mathrm{KL}[p(m^i), z(m^i)] \geq 0$, we can get $\int_{M^i} p(m^i) \log p(m^i) \, dm^i \geq \int_{M^i} p(m^i) \log z(m^i) \, dm^i$. Hence, the following inequality can be derived:

$$\begin{aligned} & I^i\left(m^i; o^{tot}, c\right) \leq \\ & \int_C \int_{O^{tot}} \int_{M^i} p(o^{tot}) p(c \mid o^{tot}) p\left(m^i \mid o^{tot}, c\right) \log \frac{p\left(m^i \mid o^{tot}, c\right)}{z\left(m^i\right)} \, dm^i \, do^{tot} \, dc \end{aligned} \tag{23}$$

By combining Eq.(21) and Eq.(23), we can estimate IB for our team-wise communication based on Eq.(18).

In the following section, we will discuss the practical implementation of applying the information bottleneck to optimize the conditional action policy in reinforcement learning.

4.4.2 Conditional Action Policy Optimization

The loss function for the conditional action policy π^i can be defined as the sum of the information bottleneck loss L_{IB} and the RL loss L_c related to the communication structure c. Mathematically, it can be expressed as follows:

$$L_\pi = \alpha L_{\mathrm{IB}} + L_c \tag{24}$$

where α is a hyper-parameter to balance these two kinds of loss.

To estimate the information bottleneck loss L_{IB} in practice, we employ the reparameterization trick [52] for estimating $p(m^i \mid o^{tot}, c)$. In this approach, we assume that the message m^i has d dimensions and is sampled from a normal distribution $\mathcal{N}\left(m^i \mid f_e^\mu(o^{tot}, c), f_e^\Sigma(o^{tot}, c)\right)$. Here, f_e refers to a multi-layer perceptron (MLP) that outputs both the d-dimensional mean μ of m^i and the $d \times d$ covariance matrix Σ. By sampling from this distribution, we can relate $p(m^i \mid o^{tot}, c)dcdo^{tot}$ to $p(\epsilon)d\epsilon$, where $m^i = f(o^{tot}, c, \epsilon)$ is a deterministic function of o^{tot}, c and the Gaussian random variable ϵ. This equivalence allows us to estimate L_{IB} as follows:

$$L_{\mathrm{IB}} = \sum_{i=1}^{n}\{\frac{1}{d}\sum_{1}^{d}\mathbb{E}_{\epsilon\sim p(\epsilon)}\left[-\log q\left(a^i \mid f\left(o^{tot}, c, \epsilon\right)\right)\right] + \beta \mathrm{KL}\left[p\left(m^i \mid o^{tot}, c\right), z(m^i)\right]\} \tag{25}$$

Now, we consider the L_c. With a fixed teaming reason policy $\rho(\varrho)$, a fully decentralized multi-agent Actor-Critic framework [53] is adopted, where the loss L_c is expressed as:

$$\mathcal{L}_c = -\mathbb{E}_{\pi\theta}\left[\log \pi_\theta\left(a_t \mid \hat{o}^i\right) \cdot Q^{\text{critic}}\left(\hat{o}_t^i, a_t^i\right)\right] \tag{26}$$

The critic network ϕ takes aggregating message $\hat{o}$ after communication as input instead of the global observation o^{tot}. The loss $\mathcal{L}_Q(\phi)$ for the critic network is defined as:

$$\mathcal{L}_Q^{\text{critic}}(\vartheta) = \mathbb{E}_{\left(o_t^i, a_t^i\right)\sim\mathcal{D}}\left[\left(y_t^i - Q^i\left(\hat{o}_t^i, a_t^i; \vartheta\right)\right)^2\right] \tag{27}$$

with $y_t^i = r^i + Q^i\left(\hat{o}_{t+1}^i, a_{t+1}^i; \bar{\vartheta}\right)$, where $\bar{\phi}$ is the frozen critic network used for stable training, and $\mathcal{D}$ is the replay buffer used for sampling. Please note that if the environment has no individual reward r^i, the local state-action value Q^i can be replaced by the global state-action value Q^{tot}.

4.5 Discussion

Previous approaches (e.g., [8, 9]) with dynamic communication directly employ neural networks to predict the existence of edges, forming communication topology. However,

the search space for topology exploration grows exponentially with the number of agents. For example, if there are n agents in the environment, the number of possible topologies can be as high as $2^{n(n-1)}$. Such a vast space may limit the quality of the learned topology. To address this, TieComm [11] learns a threshold to remove edges with lower tie strengths based on a given prior topology. It offers the advantages of a smaller search space and lower complexity in generating hierarchical topology. Our method TeamComm, on top of TieComm, eliminates the need for a prior relay topology by directly leveraging the original agent's local observations. Furthermore, TeamComm directly utilizes the agent team assignment matrix, eliminating the need for more complex team detection methods such as greedy methods (e.g., Louvain method [46]). This approach provides greater flexibility compared to relying solely on a fixed threshold while avoiding exponential space exploration.

4.6 Practical Implementations

In our experiments, we utilized parameter sharing [54] to enhance training efficiency. This approach involves sharing the parameters of the policy network among homogeneous agents. The core of our method relies on a multi-threaded synchronous multi-agent policy gradient, while the value function is estimated using an additional value head in the policy network [55]. For a detailed implementation of our TeamComm method, please refer to Algorithm 1.

5 Numerical Experiments

In this section, we aim to study the following research questions in our study: **RQ1:** Can TeamComm achieve competitive performance when compared with a diverse set of state-of-the-art communication methods? **RQ2:** Are heterogeneous communication channels necessary? **RQ3:** What is the impact of the team-based communication structure on the performance of our method? Does the team reasoning policy effectively form useful teams? **RQ4:** Does the information bottleneck make communication more effective?

5.1 Algorithm setup

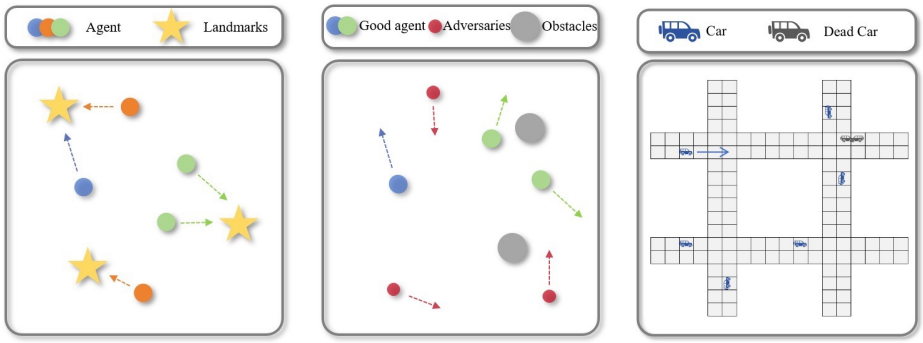
In our experiment, parameter sharing is used across the homogeneous agents for training efficiency in all models. Besides, all models employ a multi-threaded synchronous multi-agent actor-critic with RMSProp optimizer of leaning rate $1e-3$. All types of messages are set up to 64 hidden units. All methods are evaluated on them with 5 different random seeds. We increase the smoothing effect slightly to account for the variability in results. The source code, data, and appendix with additional details can be accessed through this public link: https://github.com/MinGink/TeamComm.

5.2 Evaluation Environment

As shown in Fig.6, we evaluate the methods on three popular multi-agent scenarios, namely Predator Prey (PP), Cooperative Navigation (CN) in MPE environments [56], and Traffic Junction (TJ)[6]. To meet the requirement of partial observability for

Algorithm 1 TeamComm

Initialize the weights θ of conditional action policy π^i for all TeamComm agents $i \in N$ and ϱ for team reasoning policy ρ.
for episode = 1 to MAX_EPISODE **do**
  for $t = 1$ to MAX_STEP **do**
    for each TeamComm agent $i \in N$ **do**
      observe around and generate its own original local observation o_t^i.
    end for
    generate team assignment matrix c_t following $\rho(o_t^{tot}; \varrho)$
    for each TeamComm agent $i \in N$, given c, **do**
      generate intra-team message m_{intra}^i following Eq.(15).
      do team representation pooling for each team.
      generate inter-team message m_{inter}^i following Eq.(15).
      form the final local observation $\hat{o}_t^i$ following Eq.(5).
      select an action $a_t^i \sim \pi^i\left(\cdot \mid \hat{o}_t^i; \theta\right)$.
    end for
    execute joint action $a_t^{tot} = \{a_t^i\}_{i=1}^n$.
    observe each agent reward r_i^t and the next state observation $o_{t+1}^{tot} = \{o_t^i\}_{i=1}^n$.
    store the tuple $(o_t^{tot}, c_t, a_t^{tot}, r_t, o_{t+1}^{tot})$ in the replay buffer.
  end for
  Sample batches of tuples from buffer for gradient calculation.
  Fix the ϱ in ρ, update θ in π following Eq.(27).
  Fix the θ in π,update ϱ in ρ following Eq.(9) or Eq.(14).
end for



(a) Cooperative Navigation (b) Predator Prey (c) Traffic Junction

Fig. 6 Environments used in this manuscript, more details can be seen in Appendix.

communication learning, we made certain modifications. In these scenarios, agents are tasked with communicating with specific agents and exchanging relevant information to prevent collisions and successfully complete their objectives. Table 1 and Table 2 provide a brief overview of the settings for each scenario.

Table 1 Brief description of scenario settings on MPE environment

Environment	Agents	Types	Distribution	Targets
CN(1)	9	3	[3, 3, 3]	3
CN(2)	12	3	[3, 4, 5]	3
CN(3)	15	3	[5, 5, 5]	3
PP(1)	6	2	[2, 4]	2
PP(2)	8	2	[3, 5]	2

Table 2 Brief description of scenario settings on TJ environment

Environment	Agents	Vision	Steps	Map	Level
TJ(1)	20	1	80	18×18	hard
TJ(2)	30	1	80	21×21	hard

Table 3 Summary of the performance of MARL policies. Best values in bold.

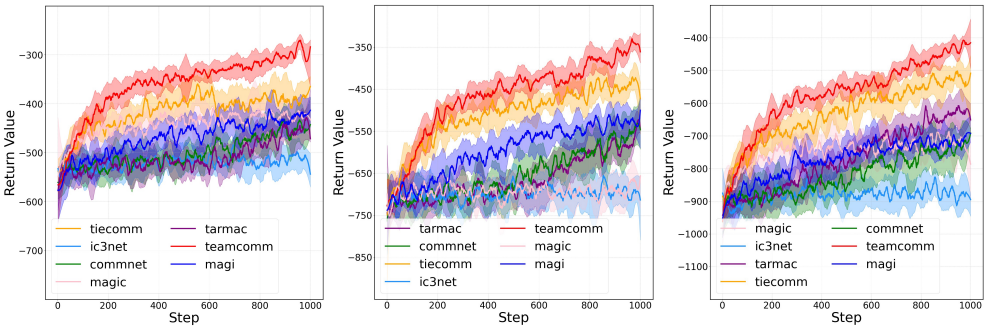
	CommNet	IC3Net	TarMAC	MAGIC	MAGI	TieComm	TeamComm
CN(1)	-365.1±18.5	-427.3±11.3	-375.9±13.1	-330.3±18.1	-337.9±14.1	-301.3±9.1	**-277.8±9.1**
CN(2)	-460.8±19.6	-575.5±11.1	-475.9±9.1	-571.7±16.3	-431.9±16.3	-360.5±18.2	**-298.4±15.1**
CN(3)	-673.3±31.5	-711.2±22.8	-543.8±11.5	-576.3±22.5	-560.1±17.3	-429.8±13.2	**-365.8±16.5**
PP(1)	306.8±25.1	176.4±12.1	350.1±11.7	248.7±16.8	267.7±19.2	370.4±13.1	**426.7±25.3**
PP(2)	530.4±16.9	242.6±12.3	560.7±28.8	603.1±17.1	582.4±17.5	732.9±36.8	**788.1±37.6**
TJ(1)	-73.1±41.2	-62.5±22.8	-118.2±32.6	-55.7±23.1	-51.2±25.2	-41.8±15.1	**-40.3±23.9**
TJ(2)	-316.6±36.3	-376.8±17.6	-415.5±40.1	-457.9±113.2	-317.2±24.2	**-266.1±29.4**	-295.9±49.2

5.3 Results and Discussion

5.3.1 RQ1: Performance

To evaluate the effectiveness of our approach, we prepared the following methods as baselines: (1) CommNet[4], (2) IC3Net[6], (3) TarMAC[7], (4) MAGIC[9], (5) TieComm[11], and (6) MAGI[38].

Fig. (7) illustrates the convergence speed of all models on CN. Table (3) presents the performance in all environments. More experiment results can be found in public repository. Across nearly all settings, our TeamComm outperforms the baseline models, while TieComm ranks second in terms of performance and learning efficiency. This verifies that the team-wise communication structure is a suitable choice for certain multi-agent scenarios. Our method outperforms TieComm in several aspects. Firstly, we adopt a more flexible team reasoning policy, providing greater adaptability during the teaming process. Secondly, by ensuring gradient backpropagation during team representation pooling, we avoid the risk of information loss. Furthermore, TeamComm attempts to communicate a valuable and precise message. These combined advantages significantly enhance our approach compared to TieComm, even though both methods utilize a team-wise communication structure. Furthermore, it is worth noting that the advantage of our method becomes more pronounced as the number of agents increases, as illustrated in Fig. 7(a) to Fig. 7(c) (Please note that the ordinate scales

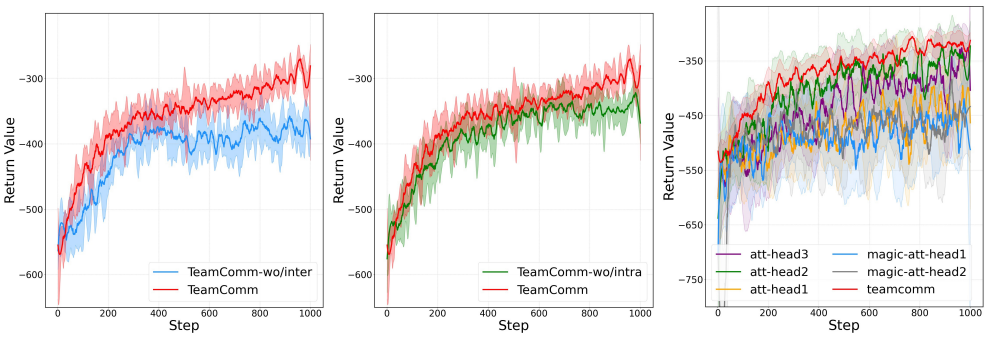


(a) CN(1) with 9 agents (b) CN(2) with 12 agents (c) CN(3) with 15 agents

Fig. 7 The learning curves on three different settings in Cooperative Navigation. Our method (red) demonstrates the highest performance, while TieComm achieves the second-best results.

are different between the figures). This observation suggests that teaming could be a viable approach to mitigate the scalability of the multi-agent problem.

5.3.2 RQ2: Effect of Heterogeneous Communication Channels



(a) TeamComm-wo/inter (b) TeamComm-wo/intra (c) Vary channels

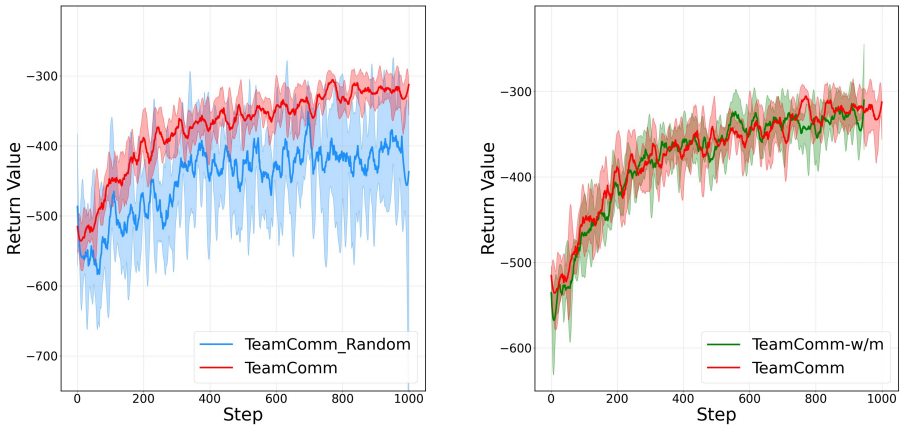
Fig. 8 Results for RQ2 in CN task.

To investigate the impact of heterogeneous communication channels, we consider two aspects. (1) We examine the impact of each communication channel in our method. (2) We explore whether simply increasing the number of channels is sufficient. To address (1), we design two variant versions, TeamComm-wo/inter and TeamComm-wo/intra, where inter-team and intra-team message passing are blocked from the beginning of training, respectively. Note that essential information can still be transmitted through the remaining channels due to gradient optimization. For (2), we design a basic attention method with varying attention heads and MAGIC [9] with different attention heads. We focus on MAGIC [9] as a baseline because its method shares similarities with ours, as both involve graph generation. The key difference lies in

our approach, which incorporates heterogeneous message channels, whereas MAGIC [9] only increases the number of message channels without considering heterogeneity, thereby lacking a crucial limitation.

As shown in Fig.(8), restricting any communication channel in our method negatively impacts performance. Additionally, increasing the number of attention heads yields improvements only up to a certain point, beyond which additional heads do not necessarily lead to further gains. In other words, directly increasing the number of channels is not necessarily effective; instead, we should also limit different channels to release different roles, thereby maintaining heterogeneity. Therefore, our method consistently outperforms other baseline methods, underscoring the essentiality of employing diverse communication channels.

5.3.3 RQ3: Impact of Team-based Communication Structure



(a) meaningful team formation has a positive impact on performance

(b) TeamComm-w/m exhibits performance similar to TeamCom.

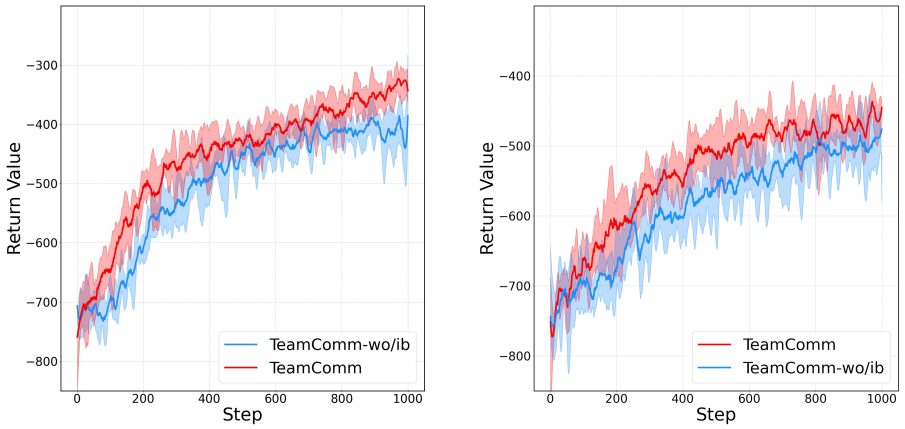
Fig. 9 Results for RQ3 in CN task.

To evaluate the impact of our team-based communication structure, we design a variant called TeamComm-Random, where agents are randomly assigned to groups. We compare its performance with two other versions: TeamComm, which uses the team reasoning policy to optimize the global Q-value following Proposition (1), and TeamComm-w/m, which optimizes the team reasoning policy based on Eq. (14). To ensure a fair comparison, we conduct all experiments in the same environment setting.

As shown in Fig. 9(a), our TeamComm approach outperforms the TeamComm-Random variant, demonstrating the effectiveness of our method in forming useful dynamic team-based communication structures. Fig. 9(b) illustrates that our proposed modularity loss, used to optimize the team reasoning policy, achieves similar performance to the global action-state value Q^{tot}. This suggests that, unlike most previous MARL studies (e.g., [8, 9]) that directly optimize the global Q value maximum, communication optimization can be a separate and independent objective. Notably, our

modularity loss may provide a viable alternative for tasks where no global reward is available. We have revised Section 5.3.3 to emphasize that our method offers a new option.

5.3.4 RQ4 Impact of Information Bottleneck



(a) TeamComm-wo/ib (CN 1) (b) TeamComm-wo/ib (CN 2)

Fig. 10 Results for RQ4 in CN task: IB can improve the performance.

We examined the impact of the Information Bottleneck in TeamComm by evaluating a variant version called TeamComm-wo/ib, where the information bottleneck loss is removed. As shown in Fig.10, TeamComm outperforms TeamComm-wo/ib, which aligns with the observation mentioned in Section 4.1. This result indicates the importance of valuable and concrete communication content for effective communication.

6 Final Remarks

In this work, we introduce a novel approach called TeamComm, which aims to learn effective team-wise communication with heterogeneous channels for both cooperative and competitive reinforcement learning tasks. Our approach addresses the key questions of "whom", "how", and "what" to communicate by employing an end-to-end differentiable framework. We decompose the learning process into two components: the joint policy of conditional action policies of agents and the team reasoning policy. These components are jointly optimized through an iterative process, and we incorporate the information bottleneck to enhance effective communication. Through extensive experiments on three representative multi-agent reinforcement learning environments, namely Traffic Junction, Predator Prey, and Cooperative Navigation, we demonstrate the effectiveness of our method compared to various state-of-the-art

approaches. Going forward, we plan to further explore the agent's ability to identify and coordinate with its teammates from alternative perspectives, such as reward sharing among teammates, thereby enhancing our understanding of multi-agent collaboration.

References

[1] Zivan, R., Rachmut, B., Perry, O., *et al.*: Effect of asynchronous execution and imperfect communication on max-sum belief propagation. Autonomous Agents and Multi-Agent Systems **37**, 40 (2023) https://doi.org/10.1007/s10458-023-09621-w

[2] Yang, M., Wang, Y., Yu, Y., Zhou, M., U, L.H.: Mixlight: Mixed-agent cooperative reinforcement learning for traffic light control. IEEE Transactions on Industrial Informatics, 1–9 (2023) https://doi.org/10.1109/TII.2023.3296910

[3] Foerster, J.N., Assael, Y.M., Freitas, N., Whiteson, S.: Learning to communicate with deep multi-agent reinforcement learning. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 2145–2153 (2016)

[4] Sukhbaatar, S., Fergus, R., *et al.*: Learning multiagent communication with backpropagation. Advances in Neural Information Processing Systems **29**, 2244–2252 (2016)

[5] Jiang, J., Lu, Z.: Learning attentional communication for multi-agent cooperation. Advances in Neural Information Processing Systems **31**, 7254–7264 (2018)

[6] Singh, A., Jain, T., Sukhbaatar, S.: Learning when to communicate at scale in multiagent cooperative and competitive tasks. In: International Conference on Learning Representations (2018)

[7] Das, A., Gervet, T., Romoff, J., Batra, D., Parikh, D., Rabbat, M., Pineau, J.: Tarmac: Targeted multi-agent communication. In: International Conference on Machine Learning, pp. 1538–1546 (2019). PMLR

[8] Du, Y., Liu, B., Moens, V., Liu, Z., Ren, Z., Wang, J., Chen, X., Zhang, H.: Learning correlated communication topology in multi-agent reinforcement learning. In: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, pp. 456–464 (2021)

[9] Niu, Y., Paleja, R., Gombolay, M.: Multi-agent graph-attention communication and teaming. In: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, pp. 964–973 (2021)

[10] Jiang, J., Dun, C., Huang, T., Lu, Z.: Graph convolutional reinforcement learning.

In: International Conference on Learning Representations (2020)

[11] Yang, M., Dong, R., Wang, Y., Liu, F., Du, Y., Zhou, M., Hou U, L.: Tiecomm: Learning a hierarchical communication topology based on tie theory. In: International Conference on Database Systems for Advanced Applications, pp. 604–613 (2023). Springer

[12] Arney, C.: Linked: How everything is connected to everything else and what it means for business, science, and everyday life. Mathematics and Computer Education **43**(3), 271 (2009)

[13] Granovetter, M.S.: The strength of weak ties. American journal of sociology **78**(6), 1360–1380 (1973)

[14] Montgomery, J.D.: Weak ties, employment, and inequality: An equilibrium analysis. American Journal of Sociology **99**(5), 1212–1236 (1994)

[15] Han, S., Dastani, M., Wang, S.: Model-based sparse communication in multi-agent reinforcement learning. In: Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, pp. 439–447 (2023)

[16] Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., Whiteson, S.: Counterfactual multi-agent policy gradients. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)

[17] Lowe, R., WU, Y., Tamar, A., Harb, J., Pieter Abbeel, O., Mordatch, I.: Multi-agent actor-critic for mixed cooperative-competitive environments. Advances in Neural Information Processing Systems **30**, 6379–6390 (2017)

[18] Du, Y., Han, L., Fang, M., Liu, J., Dai, T., Tao, D.: Liir: Learning individual intrinsic reward in multi-agent reinforcement learning. Advances in Neural Information Processing Systems **32** (2019)

[19] Han, L., Sun, P., Du, Y., Xiong, J., Wang, Q., Sun, X., Liu, H., Zhang, T.: Grid-wise control for multi-agent reinforcement learning in video game ai. In: International Conference on Machine Learning, pp. 2576–2585 (2019). PMLR

[20] Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W.M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J.Z., Tuyls, K., *et al.*: Value-decomposition networks for cooperative multi-agent learning based on team reward. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, pp. 2085–2087 (2018)

[21] Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., Whiteson, S.: Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In: International Conference on Machine Learning, pp. 4295–4304 (2018). PMLR

[22] Son, K., Kim, D., Kang, W.J., Hostallero, D.E., Yi, Y.: Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In: International Conference on Machine Learning, pp. 5887–5896 (2019). PMLR

[23] Mahajan, A., Rashid, T., Samvelyan, M., Whiteson, S.: Maven: Multi-agent variational exploration. Advances in Neural Information Processing Systems **32** (2019)

[24] Freed, B., James, R., Sartoretti, G., Choset, H.: Sparse discrete communication learning for multi-agent cooperation through backpropagation. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7993–7998 (2020). IEEE

[25] Li, X., Zhang, J.: Context-aware communication for multi-agent reinforcement learning. 23nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024) (2024)

[26] Wang, Y., Wang, Y., Sartoretti, G.: Full communication memory networks for team-level cooperation learning. Autonomous Agents and Multi-Agent Systems **37**(2), 33 (2023)

[27] Yuan, L., Wang, J., Zhang, F., Wang, C., Zhang, Z., Yu, Y., Zhang, C.: Multi-agent incentive communication via decentralized teammate modeling. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 9466–9474 (2022)

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

[29] Chu, T., Chinchali, S., Katti, S.: Multi-agent reinforcement learning for networked system control. In: International Conference on Learning Representations (2019)

[30] Gupta, S., Hazra, R., Dukkipati, A.: Networked multi-agent reinforcement learning with emergent communication. In: Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS, vol. 2020, pp. 1858–1860 (2020)

[31] Zhang, K., Yang, Z., Liu, H., Zhang, T., Basar, T.: Fully decentralized multi-agent reinforcement learning with networked agents. In: International Conference on Machine Learning, pp. 5872–5881 (2018). PMLR

[32] Kim, D., Moon, S., Hostallero, D., Kang, W.J., Lee, T., Son, K., Yi, Y.: Learning to schedule communication in multi-agent reinforcement learning. In: ICLR 2019: International Conference on Representation Learning (2019). International Conference on Representation Learning

[33] Liu, Y., Wang, W., Hu, Y., Hao, J., Chen, X., Gao, Y.: Multi-agent game abstraction via graph attention neural network. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 7211–7218 (2020)

[34] Sheng, J., Wang, X., Jin, B., Yan, J., Li, W., Chang, T.-H., Wang, J., Zha, H.: Learning structured communication for multi-agent reinforcement learning. Autonomous Agents and Multi-Agent Systems **36**(2), 50 (2022)

[35] Ruan, J., Du, Y., Xiong, X., Xing, D., Li, X., Meng, L., Zhang, H., Wang, J., Xu, B.: Gcs: Graph-based coordination strategy for multi-agentreinforcement learning. In: International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS) (2022)

[36] Seraj, E., Wang, Z., Paleja, R., Patel, A., Gombolay, M.: Learning efficient diverse communication for cooperative heterogeneous teaming. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States) (2022)

[37] Wang, R., He, X., Yu, R., Qiu, W., An, B., Rabinovich, Z.: Learning efficient multi-agent communication: An information bottleneck approach. In: International Conference on Machine Learning, pp. 9908–9918 (2020). PMLR

[38] Ding, S., Du, W., Ding, L., Zhang, J., Guo, L., An, B.: Robust multi-agent communication with graph information bottleneck optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)

[39] Oliehoek, F.A.: Decentralized pomdps. Reinforcement Learning: State-of-the-Art, 471–503 (2012)

[40] Myerson, R.: Game theory: Analysis of conflict harvard univ. Press, Cambridge **3** (1991)

[41] Voita, E., Talbot, D., Moiseev, F., Sennrich, R., Titov, I.: Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5797–5808. Association for Computational Linguistics, Florence, Italy (2019)

[42] McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: Homophily in social networks. Annual review of sociology **27**(1), 415–444 (2001)

[43] Bianchi, F.M., Grattarola, D., Alippi, C.: Spectral clustering with graph neural networks for graph pooling. In: International Conference on Machine Learning, pp. 874–883 (2020). PMLR

[44] Sutton, R.S., McAllester, D., Singh, S., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. Advances in neural information processing systems **12** (1999)

[45] Wei, E., Wicke, D., Freelan, D., Luke, S.: Multiagent soft q-learning. arXiv preprint arXiv:1804.09817 (2018)

[46] Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment **2008**(10), 10008 (2008)

[47] Corso, G., Cavalleri, L., Beaini, D., Liò, P., Veličković, P.: Principal neighbourhood aggregation for graph nets. Advances in Neural Information Processing Systems **33**, 13260–13271 (2020)

[48] Roy, K.K., Roy, A., Rahman, A.M., Amin, M.A., Ali, A.A.: Structure-aware hierarchical graph pooling using information bottleneck. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2021). IEEE

[49] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in Neural Information Processing Systems **30** (2017)

[50] Yang, B., Wang, S., Markham, A., Trigoni, N.: Robust attentional aggregation of deep feature sets for multi-view 3d reconstruction. International Journal of Computer Vision **128**(1), 53–73 (2020)

[51] Alemi, A.A., Fischer, I., Dillon, J.V., Murphy, K.: Deep variational information bottleneck. arXiv preprint arXiv:1612.00410 (2016)

[52] Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. In: 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings (2014)

[53] Iqbal, S., Sha, F.: Actor-attention-critic for multi-agent reinforcement learning. In: International Conference on Machine Learning, pp. 2961–2970 (2019). PMLR

[54] Christianos, F., Papoudakis, G., Rahman, M.A., Albrecht, S.V.: Scaling multi-agent reinforcement learning with selective parameter sharing. In: International Conference on Machine Learning, pp. 1989–1998 (2021). PMLR

[55] Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., Kavukcuoglu, K.: Asynchronous methods for deep reinforcement learning. In: International Conference on Machine Learning, pp. 1928–1937 (2016). PMLR

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