Learning Efficient and Robust Multi-Agent Communication via Graph Information Bottleneck

Shifei Ding[1,2], Wei Du[1,*], Ling Ding[3,*], Lili Guo[1,2], Jian Zhang[1,2]

[1] School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China
[2] Mine Digitization Engineering Research Center of Ministry of Education of the People's Republic of China, Xuzhou 221116, China
[3]College of Intelligence and Computing, Tianjin University, Tianjin, 300350, China
dingsf@cumt.edu.cn, 1394471165@qq.com, dltjdx2022@tju.edu.cn, liliguo@cumt.edu.cn, zhangjian10231209@cumt.edu.cn

Abstract

Efficient communication learning among agents has been shown crucial for cooperative multi-agent reinforcement learning (MARL), as it can promote the action coordination of agents and ultimately improve performance. Graph neural network (GNN) provide a general paradigm for communication learning, which consider agents and communication channels as nodes and edges in a graph, with the action selection corresponding to node labeling. Under such a paradigm, an agent aggregates information from neighbor agents, which can reduce uncertainty in local decision-making and induce implicit action coordination. However, this communication paradigm is vulnerable to adversarial attacks and noise, and how to learn robust and efficient communication under perturbations has largely not been studied. To this end, this paper introduces a novel Multi-Agent communication mechanism via Graph Information bottleneck (MAGI), which can optimally balance the robustness and expressiveness of the message representation learned by agents. This communication mechanism aims at learning the minimal sufficient message representation for an agent by maximizing the mutual information (MI) between the message representation and the selected action, and simultaneously constraining the MI between the message representation and the agent feature. Empirical results demonstrate that MAGI is more robust and efficient than state-of-the-art GNN-based MARL methods.

Introduction

Cooperative multi-agent reinforcement learning (MARL) has attracted prevalent interest and achieved surprising success in various challenging real-world tasks, such as auction trading (Qiu et al. 2021), autonomous driving (Du and Ding 2021), and traffic signal control (Yang et al. 2021). To solve the issues of scalability and non-stationarity in MARL, the paradigm of centralized training with decentralized execution (CTDE) is widely adopted, in which decentralized policies can be derived in a centralized manner so that experiences, parameters, etc., are shared during the training phase. The value function decomposition methods (Hostallero et al. 2019; Sunehag et al. 2020; Rashid et al. 2018; Wang et al. 2021) further extend this paradigm by learning a decentralized local Q function for each agent and utilizing a mixing network to integrate these local Q values into global value.

Despite the function representation advantages, value function decomposition methods still perform unsatisfactorily in multi-agent scenarios requiring action coordination, probably mainly due to partial observability and stochasticity during the decentralized execution period. Partial observability and stochasticity can heighten the uncertainty of an agent about the state and actions of other agents, which can lead to action miscoordination. To tackle these issues, various multi-agent communicative reinforcement learning (MACRL) methods have been presented, which allow agents to exchange information such as local individual observation or the corresponding feature embedding during the execution period. By exchanging messages between agents, MACRL greatly enhances the coordination ability of multiple agents in a variety of tasks (Sukhbaatar, Fergus et al. 2016; Jiang and Lu 2018; Wang et al. 2020, 2019).

Graph Neural Network (GNN) is an effective representation method that processes the attribute information and topological information of the graph-structured data into feature representation for downstream tasks such as node classification and link prediction (Gilmer et al. 2017; Park and Neville 2019). MACRL has utilized GNN to build a communication learning paradigm, which considers agents and communication channels as nodes and edges in a graph, with the action selection corresponding to node labeling. In fact, many state-of-the-art MACRL methods, such as TarMAC (Das et al. 2019) and MAGIC (Niu, Paleja, and Gombolay 2021), fall into this paradigm. Message representation learning in GNN-based MACRL is a challenging and crucial task because both agent feature embeddings and graph structure carry important information for decision-making and action selection. However, recent GNN-based MACRL works still encounter some issues. On the one side, the features of neighbor agents may include useless information that can have a negative impact on the selection of optimal action. Besides, GNN-based MACRL relies on the edges of the graph to pass messages among agents, which also makes it vulnerable to adversarial attacks and noise on the agent features and graph structure.

As shown in Figure 1, $\mathcal{D} = (A, H)$ carries information from both the graph topological structure A and agent feature embeddings H. If communication message represen-

*Corresponding author

tation carries irrelevant information from A and H, it is susceptible to hyperparameter change of model, adversarial attacks, and noise perturbations on $\mathcal{D}$. The performance of MACRL methods tends to degrade under adversarial attacks, which can leave many practical applications based on MACRL models at high risk. For example, researchers have shown that in multi-agent autonomous driving systems, adversarial attacks on the communication process between multi-agent vehicles can trick autonomous vehicles into making abnormal judgments, such as driving into the opposite lane (Du and Ding 2021).

Therefore, we tackle these issues and rethink what constitutes a "good" message representation that promotes action coordination and decision-making in GNN-based MACRL. Particularly, the Information Bottleneck (IB) (Tishby and Zaslavsky 2015) provides a crucial principle for general representation learning: the optimal representation should include the minimal sufficient information that is useful for downstream tasks. Inspired by this principle, we define the optimal message representation as the representation that contains sufficient and minimal information for the action selection task as shown in Figure 1. Nevertheless, applying the IB principle to message representation learning on agent features encounters two challenges. On the one hand, previous representation learning methods that utilize the IB principle generally assume that the input data should be independent and identically distributed (i.i.d.). For agent features, the condition is no longer supported, making the IB principle difficult to implement. On the other hand, the relational information contained in the graph structure is crucial to representing the message, however, this information is discrete and therefore difficult to optimize. In GNN-based MACRL models, how to properly obtain efficient and succinct message representation from agent features is a challenge that has not been studied yet.

To tackle these challenges, we propose a Multi-Agent communication mechanism with Graph Information bottleneck optimization (MAGI) for communication message representation learning. To address the first challenge, we propose two information-theoretic regularizers to derive the minimal sufficient communication message: one to constrain the information from the graph topological structure and agent feature embeddings, and the other to maximize the information for the action selection and coordination in the message representation. With these two regularizers, MAGI ensures communication learning to be both efficient (i.e., efficiently reducing the uncertainty of action selection of agents), succinct (i.e., message representations only contain necessary information), and robust (i.e., the communication protocol is not vulnerable to adversarial attacks and noise). Besides, to address the issue caused by non-i.i.d. agent features, we utilize the local-dependence assumption of agent features to extract information hierarchically from agent features H and graph structure A. The main contributions of our work are summarized as follows:

1) To the best knowledge, our work is the first attempt to extend the graph information bottleneck principle (Jiang et al. 2018) to GNN-based MACRL methods, which achieve efficient and robust multi-agent communication learning.

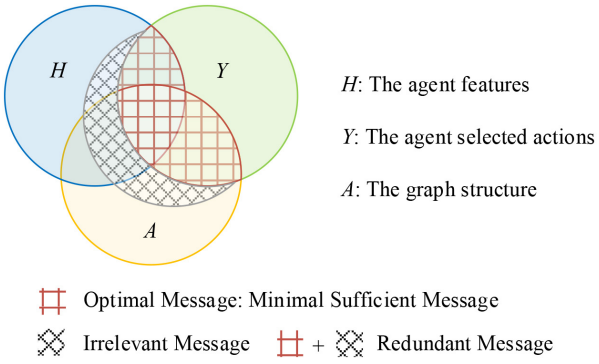

Figure 1: Multi-Agent communication via Graph Information bottleneck aims to optimize the message representation to capture the minimal sufficient information within the graph-structured agent features $\mathcal{D} = (A, H)$ to obtain optimal actions Y.

2) We propose two information-theoretic regularizers to obtain the optimal message representation that contains sufficient and minimal information for action selection and action coordination downstream tasks.

3) We propose a general MARL framework that can flexibly integrate the proposed communication learning mechanism with any value function factorization methods.

4) We evaluate the proposed method on several MARL environments, including SMAC (Wang et al. 2018) and MAgent (Sheng et al. 2020). Experimental results demonstrate that MAGI is more robust and efficient than the state-of-the-art MACRL methods.

Related Work

GNN-based MACRL

Recently, various MACRL methods utilize GNN to promote communication learning and provide a diverse GNN-based MACRL paradigm. DGN (Jiang et al. 2018) first introduces GNN to MACRL for multi-agent communication learning to learn cooperation. NerveNet (Wang et al. 2018) utilizes GNN to tackle complicated tasks with various types of agents, but it can merely handle graphs with fixed size. HAMA (Ryu, Shin, and Park 2020) presents a hierarchical attentional communication protocol based on GNNs, which effectively models the relations between agents. LSC (Sheng et al. 2020) introduces the hierarchical GNN to realize effective communication learning by exchanging messages among groups and agents. GA2NET (Liu et al. 2020) introduces a two-stage attention mechanism to model the complete graph for multi-agent communication learning. TarMAC (Das et al. 2019) also falls within this paradigm, which utilizes GAT to model the complete graph to learn what messages to pass and whom to receive messages. MAGIC (Niu, Paleja, and Gombolay 2021) presents an attentional GNN to tackle the issue of how to address messages and when to communicate. The existing GNN-based MACRL methods have successfully promoted communication by aggregating information of agent features. However, these methods are

prone to adversarial attacks and suffer from performance degradation under perturbation. How to learn efficient and robust communication under attacks has been largely unstudied, our work provides a way to solve this issue.

Adversarial Attack

Zhang et. al. (Zhang et al. 2022) introduce a general definition of adversarial attacks on graph data: (General Adversarial Attack on Graph Data) Given a graph data $\mathcal{D} = (A, H)$, after slightly modifying $\mathcal{D}$ (denoted as $\widehat{\mathcal{D}}$), the adversarial samples $\widehat{\mathcal{D}}$ and $\mathcal{D}$ should be similar under the imperceptibility metrics, but the performance of graph downstream task (such as action selection task in our work) becomes much worse than before. In general, the adversarial perturbations can be categorized as follows: 1) Modifying node features: Adversarial attack can slightly modify the node features while maintaining the structure of the graph.

Ma et al. (Ma, Ding, and Mei 2020) add adversarial perturbation on node features and set a novel local constraint on node access. Wu et al. (Wu et al. 2019) present an integrated gradients based attack method that adds perturbations on both the node features and edges. 2) Modifying Edges: Adversarial attack can add or delete edges to existing nodes with a given total action budget. Xu et al.(Xu et al. 2019) present a novel gradient-based attack method that only changes a small number of edges, including addition and deletion. Zang et al. (Zang et al. 2021) find a set of anchor nodes to mislead the classification of all nodes in the graph.

Methodology

Problem Formulation

The multi-agent reinforcement learning issues can be formulated as Decentralized Partially Observable Markov Decision Process (Dec-POMDPs). Dec-POMDPs can be represented by a tuple $< S, U, P, R, O, N >$. At each timestep, the agent selects its action a_i based upon its local observation $o_i \in O$. The joint action of agents is represented as $a = (a_1, \ldots, a_n) \in U$. The state changes based upon the transition function $P : S \times U \to S$. The objective of the agent i is to maximize its total discounted return $R_i = \sum_{t=0}^{T} \gamma^t r_i^t$, where $\gamma \in [0, 1]$ represents a discount factor. The goal is to learn a joint policy $\pi(\tau, a)$ that can maximize the global value $Q_{tot}^{\pi}(\tau, a) = \mathbb{E}_{s,a}\left[\sum_{t=0}^{\infty} \gamma^t R(s, a) \mid s_0 = s, a_0 = a\right]$, where τ represents the observation history.

In this work, we consider a graph $G = (V, E, H)$ with n nodes to model the multi-agent system with n agents, where $V = \{1, 2, \ldots n\}$ represents the node/agent set, $E \subseteq V \times V$ denotes the edge set, $H \in \mathbb{R}^{n \times f}$ represents the node attributes/agent features. We use $A \in \mathbb{R}^{n \times n}$ to represent the adjacency matrix of G, if $(i, j) \in E$, then $A_{ij} = 1$, otherwise $A_{ij} = 0$. We utilize $d(i, j)$ to represent the shortest path distance between two agents $i, j (\in V)$ over A. Therefore, our input information data for GNN-based communication module can be overall denoted as $\mathcal{D} = (A, H)$. We focus on extracting agent-level message representations $M_H \in \mathbb{R}^{n \times f'}$ from $\mathcal{D}$ so that M_H can be used to assist the agent in selecting actions Y. The subscript with an agent $i \in V$ is utilized to represent the affiliation with agent i. For example, the message representation of i is represented by $M_{H,i} = m_i$ and its corresponding action selection is represented by $Y_i = a_i$.

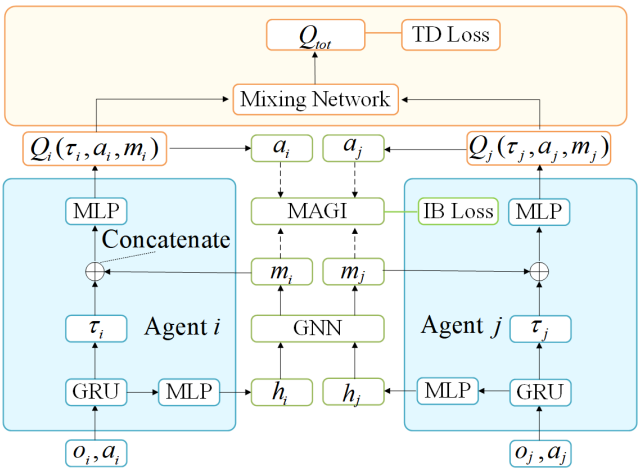

Figure 2: The framework of Multi-Agent communication via Graph Information bottleneck (MAGI).

Overall Framework

The overall framework of MAGI is shown in Figure 2, for agent i, Gated Recurrent Unit (GRU) and Multi-Layer Perceptron (MLP) are utilized to process the local observation o_i and generate the feature information h_i. During the communication learning phase, GNN is utilized to produce the message representation m_i, which fuses the feature information of neighbor agents and graph structure information. Besides, MAGI uses the graph information bottleneck principle to obtain minimal sufficient message representations. The message m_i is concatenated with the current local history τ_i to serve as an input to the local action-value function $Q_i(\tau_i, a_i, m_i)$. As shown in Figure 2, local action-values of all agents are fed into the mixing network which outputs the estimation of global action-value Q_{tot}. MAGI adopts the mixing network proposed by QMIX (Rashid et al. 2018) and it can be flexibly replaced by any mixing network of other value decomposition methods.

Multi-Agent Communication via Graph Information Bottleneck

Inspired by the principle of graph information bottleneck (GIB), Multi-Agent communication via Graph Information bottleneck (MAGI) necessitates the message representation M_H to maximize the information to selected actions Y and minimize the information from the feature $\mathcal{D} = (A, H)$. The minimization target of MAGI is represented in Eq.(1), where $I(\cdot; \cdot)$ represents the mutual information,

$$\min \mathrm{MAGI}_{\beta}(\mathcal{D}, Y; M) \triangleq [-I(Y; M) + \beta I(\mathcal{D}; M)]. \quad (1)$$

We utilize a comprehensively adopted local-dependence assumption for the graph-structured agent features: Given the

neighbor-related agent for a given agent i within a certain number of hops, the other agent features are independent of the agent feature of agent i. We leverage this assumption to constrain the optimal message representations space Ω, which makes the MAGI principle more tractable. In other words, we assume optimal message representation follows the Markovian dependence.

Concretely, we use $\mathbb{P}(M_H \mid \mathcal{D})$ to iterate message representations to model the correlation of features hierarchically, where $\mathbb{P}(\cdot)$ represents joint probabilistic distribution function. At each time step, a graph neural network layer corresponds to one round of message exchange among the agent and the neighbor agents. In each message exchange l, we leverage the local-dependence assumption: The message representation of each agent will be refined by aggregating features of its neighbor agents, w.r.t a graph structure M_A^l. Therefore, $(M_A^l)_{1\leq l\leq L}$ is acquired by modifying the original graph structure A locally, which fundamentally controls the message flow from graph structure A. In the final, we leverage high-level message representation M_H^L to coordinate the action. According to this formulation, the objective of MAGI minimization can be reduced to the following optimization:

$$\begin{aligned}&\min_{\mathbb{P}(M_H^L|\mathcal{D})\in\Omega} \mathrm{MAGI}_\beta(\mathcal{D}, Y; M_H^L)\\ &\triangleq \left[-I(Y; M_H^L) + \beta I(\mathcal{D}; M_H^L)\right],\end{aligned} \tag{2}$$

where Ω represents the optimal message representation space of the conditional distribution of M_H^L given the agent feature $\mathcal{D}$. In this formula, we only need to optimize two series of distributions $\mathbb{P}(M_H^l \mid M_H^{l-1}, M_A^l)$ and $\mathbb{P}(M_A^l \mid M_H^{l-1}, A)$, which have local dependence between agents and therefore are easier to be optimized.

We leverage the simplistic MAGI principle and some proper parameterization of $\mathbb{P}(M_H^l \mid M_H^{l-1}, M_A^l)$ and $\mathbb{P}(M_A^l \mid M_H^{l-1}, A)$, the calculation of $I(Y; M_H^L)$ and $I(\mathcal{D}; M_H^L)$ of Eq.(2) is still intractable. Therefore, we should introduce variational bounds on the two terms of Eq.(2), which leads to the optimization of the final objective. As proved in (Wu et al. 2020), we can obtain the lower bound of $I(Y; M_H^L)$ and the upper bound of $I(\mathcal{D}; M_H^L)$, which are represented in Eq.(3) and Eq.(4), respectively. For any distributions $\mathbb{V}_1(Y_i \mid M_{H,i}^L)$ for $i \in V$ and $\mathbb{V}_2(Y)$, the lower bound of $I(Y; M_H^L)$ is shown in Eq.(3),

$$\begin{aligned}I(Y; M_H^L) \geq 1 + \mathbb{E}\left[\log \frac{\prod_{i\in V} \mathbb{V}_1(Y_i \mid M_{H,i}^L)}{\mathbb{V}_2(Y)}\right]\\ + \mathbb{E}_{\mathbb{P}(Y)\mathbb{P}(M_H^L)}\left[\frac{\prod_{i\in V} \mathbb{V}_1(Y_i \mid M_{H,i}^L)}{\mathbb{V}_2(Y)}\right].\end{aligned} \tag{3}$$

We select two sets of indices $S_H, S_A \subset [L]$ such that $\mathcal{D} \perp M_H^L \mid (M_H^l)_{l\in S_H} \cup (M_A^l)_{l\in S_A}$ according to the Markovian dependence, where $M_1 \perp M_2 \mid M_3$ to represent that M_1 and M_2 are conditionally independent given M_3. Then, for any distributions $\mathbb{V}(M_H^l), l \in S_H$ and $\mathbb{V}(M_A^l), l \in S_A$,

$$\begin{aligned}I(\mathcal{D}; M_H^L) &\leq I\left(\mathcal{D}; (M_H^l)_{l\in S_H} \cup (M_A^l)_{l\in S_A}\right)\\ &\leq \sum_{l\in S_A} \mathrm{AIB}^l + \sum_{l\in S_H} \mathrm{HIB}^l,\end{aligned} \tag{4}$$

$$\mathrm{AIB}^l = \mathbb{E}\left[\log \frac{\mathbb{P}(M_A^l \mid A, M_H^{l-1})}{\mathbb{V}(M_A^l)}\right], \tag{5}$$

$$\mathrm{HIB}^l = \mathbb{E}\left[\log \frac{\mathbb{P}(Z_X^l \mid M_H^{l-1}, M_A^l)}{\mathbb{V}\left(M_H^l\right)}\right]. \tag{6}$$

Eq.(4) indicates that we should choose a set of random variables with index S_H and S_A to ensure the conditional independence between $\mathcal{D}$ and M_H^L. S_H and S_A have the following properties: (1) $S_H \neq \emptyset$. (2) if the greatest index in S_H is l, the S_A contains all the integers of $[l+1, L]$.

In order to utilize MAGI principle, we should model $\mathbb{P}(M_A^l \mid M_H^{l-1}, A)$ and $\mathbb{P}(M_H^l \mid M_H^{l-1}, M_A^l)$. Then, we select variational distributions $\mathbb{V}(M_H^l)$ and $\mathbb{V}(M_A^l)$ to estimate the corresponding AIB^l and HIB^l for regularization, and some $\mathbb{V}_1(Y_i \mid M_{H,i}^L$. and $\mathbb{V}_2(Y)$ to give the lower bound in Eq.(3). Therefore, we can obtain the upper bound to optimize the objective by applying Eq.(3) and Eq.(4) to the Eq.(2).

The MAGI principle can be applied to various GNN-based MACRL methods. In this work, we utilize graph attention network (GAT) (Veličković et al. 2018) as the GNN architecture in the communication learning module. In each GAT layer, MAGI should first refine the graph structure of multi agent leveraging the attention weights to obtain M_A^l and then refine feature representations M_H^l by propagating M_H^{l-1} over M_A^l.

For neighbor agents sampling, we utilize categorical distribution and consider attention weights to be the parameters of the categorical distributions, which can sample the structure of the refined graph to capture structural information. We then extract k neighbor agents with alternatives from the built set of agents V_{ic} for agent i, in which V_{ic} contains the agents whose shortest-path-distance to agent i is c. We utilize $\mathcal{C}$ to be the upper limitation of c to ensure the assumption of local dependence. Then we sum-pool the neighbor agents and use the output to calculate the parameters of the Gaussian distribution, in which the refined agent features are sampled.

In order to optimize parameters of MAGI module, the bounds of term $I(Y; M_H^L)$ in Eq.(3) and $I(\mathcal{D}; M_H^L)$ in Eq.(4) should be specified to further calculate the bound of MAGI in Eq.(2). In order to characterize AIB^l in Eq.(4), we can assume $\mathbb{V}(M_A^l)$ is a non-informative distribution. Concretely, we utilize the uniform categorical distribution: $M_A \sim \mathbb{V}(M_A)$, $M_{A,i} = \cup_{d=1}^{\mathcal{T}} \left\{j \in V_{ic} \mid j \overset{\text{iid}}{\sim} \mathrm{Cat}\left(\frac{1}{|V_{ic}|}\right)\right\}$ and $M_{A,i} \perp M_{A,j}$ if $i \neq j$. We utilize $\mathrm{Cat}(\phi)$ to represent the categorical distribution with parameter ϕ, which corresponds to different categories of probabilities and therefore $\|\phi\|_1 = 1$. After the module calculate ϕ_{ic}^l, we can obtain an

empirical estimation of AIB^l,

$$\widehat{\mathrm{AIB}}^l = \mathbb{E}_{\mathbb{P}(M_A^l|A,M_H^{l-1})}[\log \frac{\mathbb{P}(M_A^l \mid A, M_X^{l-1})}{\mathbb{V}(M_A^l)}], \quad (7)$$

which is instantiated as follows,

$$\widehat{\mathrm{AIB}}^l = \sum_{i\in V, d\in[\mathcal{T}]} \mathrm{KL}(\mathrm{Cat}(\phi_{id}^l)||\,\mathrm{Cat}(\frac{1}{|V_{id}|}). \quad (8)$$

To estimate HIB^l, we set $\mathbb{V}(M_H^l)$ as a mixture of Gaussian distributions. Concretely, $M_H \sim \mathbb{V}(M_H)$, we set $M_{H,i} \sim \sum_{u=1}^m w_u \,\mathrm{Gaussian}\left(\mu_{0,u}, \sigma_{0,u}^2\right)$, where $w_u, \mu_{0,u}, \sigma_{0,u}$ denote learnable parameters shared by all agents and $M_{H,i} \perp M_{H,j}$, if $i \neq j$. We can estimate HIB^l by utilizing the sampled M_H^l:

$$\begin{aligned}\widehat{\mathrm{HIB}}^l = \sum_{i\in V}[&\log \Phi\left(M_{H,i}^l; \mu_i, \sigma_i^2\right) \\ &- \log(\sum_{u=1}^n w_u \Phi\left(M_{H,i}^l; \mu_{0,u}, \sigma_{0,u}^2\right))].\end{aligned} \quad (9)$$

Hence, we can choose appropriate index set S_H, S_A that satisfy the assumption in Eq.(4) and utilize substitution:

$$I(\mathcal{D}; M_H^L) \to \sum_{l\in S_A} \widehat{\mathrm{AIB}}^l + \sum_{l\in S_H} \widehat{\mathrm{HIB}}^l. \quad (10)$$

To characterize Eq.(3), we can straightly set $\mathbb{V}_2(Y) = \mathbb{P}(Y)$ and $\mathbb{V}_1(Y_i \mid Z_{H,i}^L) = \mathrm{Cat}(M_{H,i}^L)W_{\text{out}}$. Therefore, the Eq.(3) can reduce to the cross-entropy loss without constants as follows,

$$I(Y; M_H^L) \to -\sum_{i\in V} \text{Cross-Entropy}\left(M_{H,i}^L W_{\text{out}}; Y_i\right). \quad (11)$$

Applying Eq.(10) and Eq.(11) to Eq.(2), MAGI objective can be obtained to train communication learning module.

Apart from the MAGI constraints on the message representations learning in the communication component, all the parameters in other components (feature process component and value decomposition component) are updated by minimizing the TD loss L_{TD}. In the end, TD loss and the overall optimization objective of MAGI are presented in Eq.(12) and Eq.(13), respectively.

$$L_{TD} = \left[r + \gamma \max_{a'} Q_{tot}\left(\tau', a'; \theta^-\right) - Q_{tot}(\tau, a; \theta)\right], \quad (12)$$

where θ represents all parameters in the MAGI and θ^- denotes the parameters of target network.

$$L = L_{TD} + \lambda L_{IB}, \quad (13)$$

where λ is a hyper-parameter that can be adjusted to achieve a trade-off between the IB loss $L_{IB} = \mathrm{MAGI}_\beta(\mathcal{D}, Y; M_H^l)$ and the TD loss L_{TD}. We set $\lambda = 0.1$ based on experimental results, which can be found in Ablations. The detail of MAGI is shown in Algorithm 1.

Algorithm 1: MAGI

Input: $o_i \in O$ and $a_i^{t-1} \in A$ of agent i
Initialize: The weights of networks W, the maximum size of replay buffer b, the number of neighbor agents to be sampled k, the integral limitation to impose local dependence $\mathcal{C}$.
Output: Global action-value Q_{tot}

for each timestep t **do**
  for each agent i **do**
    % During decentralized execution period
    Generate agent feature h_i by GRU and MLP
    Construct matrix data $\mathcal{D} = (A, H)$ based on h_i
    Input $\mathcal{D}$ to $L-$ layers GNN
    for Layers = $1, \ldots, L$ **do**
      $\tilde{M}_{H,i}^{l-1} \leftarrow \sigma\left(M_{H,i}^{l-1}\right) W^l$
      construct sets $V_{ic} \leftarrow \{j \in V \mid d(i,j) = c\}$
      for $c \in [\mathcal{C}]$ **do**
        $\phi_{ic}^l \leftarrow \mathrm{softmax}\left\{\left(\tilde{M}_{H,i}^{l-1} \oplus \tilde{M}_{H,j}^{l-1}\right) a^T\right\}$
        $M_{H,i}^l \leftarrow \cup_{c=1}^{\mathcal{C}}\left\{j \in V_{ic} \mid j \overset{\text{iid}}{\sim} \mathrm{Cat}\left(\phi_{ic}^{l-1}\right)\right\}$
      end for
      $\bar{M}_{H,i}^l \leftarrow \sum_{j\in M_{H,i}^l} \tilde{M}_{H,i}^{l-1}$
      $\mu_i^l \leftarrow \bar{M}_{H,i}^l[0 : f']$
      $\sigma_i^{2l} \leftarrow \mathrm{softplus}\left(\bar{M}_{H,i}^l[f' : 2f']\right)$
      $M_{H,i}^l \sim \mathrm{Gaussian}\left(\mu_i^l, \sigma_i^{2l}\right)$
    end for
    Obtain final message representation $m_i = M_{H,i}^L$
    Calculate action-value Q_i based on m_i and τ_i
    $a_i^t \leftarrow \pi\left(Q_i\right)(\epsilon-$ greed $)$
    Store episode history τ_i and a_i^t in replay buffer
    % During centralized training period
    Input Q_i to mixing network and output Q_{tot}
    Minimize loss function based on Eq.(13)
    Update weights of all networks
  end for
end for

Experiments

In this section, we conduct various experiments on two complicated environments including StarCraft II Multi-Agent Challenge (SMAC)(Samvelyan et al. 2019) and MAgent(Zheng et al. 2018) to answer: **Q1**: Are MAGI and other GNN-based MACRL methods vulnerable to adversarial attacks and noise? **Q2**: Can MAGI module improve the robustness of communication learning under adversarial attacks and noise? **Q3**: Can MAGI scale to large-scale multi-agent settings? **Q4**: Which components contribute to the performance of MAGI? **Q5**: How does λ and η influence the performance of MAGI. **Q6**: Can MAGI perform well under more complicated adversarial attack methods? We select QMIX (Rashid et al. 2018), TarMAC (Das et al. 2019), and MAGIC (Niu, Paleja, and Gombolay 2021) as baseline methods. The details of the environment description, baselines introduction, and hyper-parameters settings are given in Appendix.

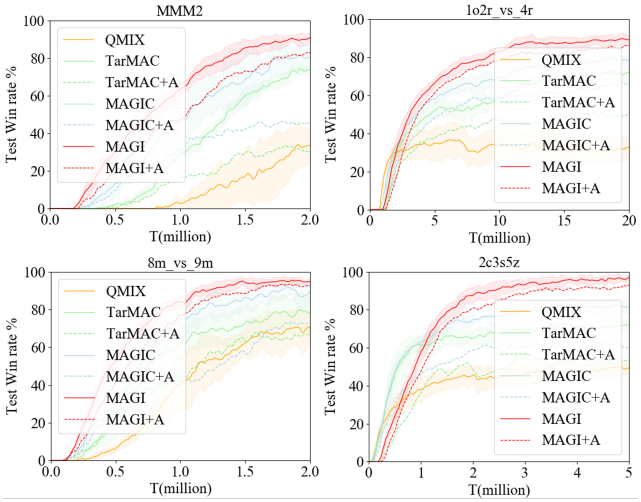

Figure 3: Learning curves of different methods under adversarial attacks and noise ($\eta = 1.5$).

Robustness (Q1,Q2). To demonstrate the robustness and effectiveness of communication learning of MAGI under adversarial attacks and noise, we generate random perturbations and inject them into the agent features H and adjacency matrix A. Concretely, for agent features, independent Gaussian noise is added to agent features H with increasing amplitude. The average of the maximum feature value of each agent is utilized to be the reference amplitude ϕ, and we inject Gaussian noise $\eta \cdot \phi \cdot \epsilon$ for each agent feature, where $\epsilon \sim N(0, 1)$, and η denote the feature noise ratio. We evaluate the robustness of GNN-based MACRL methods with different parameters $\eta \in \{0.5, 1, 1.5\}$.

To generate perturbations on the adjacency matrix A, we first randomly drop the edges of the graph structure. However, this simple strategy does not significantly reduce the communication effectiveness of the MACRL methods. Therefore, projected gradient descent (Xu et al. 2019) is adopted to generate perturbations on the graph structure, instead of randomly dropping edges. The learning curves of MAGI and baselines on several scenarios of SMAC are shown in Figure 3. The mean win rate of all the methods without adversarial attack (QMIX, TarMAC, MAGIC, and MAGI) is represented by the solid line in the middle, and the corresponding shaded area shows a 95% confidence interval. The dashed line shows the win rate of the GNN-based MACRL methods with adversarial attacks and noise (TarMAC+A, MAGIC+A, and MAGI+A).

As shown in Figure 3, we can draw several conclusions as follows: 1) MAGI performs significantly better than other baseline methods in all scenarios. 2) Besides, without adversarial attack, by comparing MAGI with other GNN-based MACRL methods (TarMAC and MAGIC), MAGI performs better than other baselines, which demonstrates the effectiveness of the proposed framework that fusing the communication learning mechanism with value factorization. 3) Furthermore, the performance of TarMAC+A and MAGIC+A degrades significantly compared with TarMAC and MAGIC respectively, demonstrate the other existing GNN-based MACRL methods are susceptible to adversarial attacks. 4) Comparing with MAGI, MAGI+A shows only a slight performance degradation, which demonstrates that MAGI is robust under adversarial attacks.

Scalability (Q3). To demonstrate that MAGI can be applied to large-scale multi-agent scenarios, we compared MAGI and baselines under adversarial attack on Battle scenario of MAgent with the different number of agents ($K \in \{30, 40, 50\}$). As shown in Table 1, MAGI can always perform best compared with baselines as the number of agents increases, which demonstrates the scalability ability of the MAGI. The proposed framework integrates communication learning and value decomposition, which can be a paradigm for tackling large-scale multi-agent communication learning tasks.

Methods	$K = 30$	$K = 40$	$K = 50$
TarMAC	0.92±0.21	1.05±0.13	1.09±0.08
MAGIC	0.96±0.15	1.12±0.21	1.14±0.16
MAGI	**1.17±0.12**	**1.26±0.09**	**1.32±0.06**

Table 1: Mean reward of different methods with different number of agents in Battle of MAgent

Contribution (Q4). We further evaluate the contribution of each component in MAGI. We design two variants of MAGI: 1) MAGI-VD is MAGI without the value decomposition component, such that we input individual value into the fully connected layer and output the global value. 2) MAGI-IB is MAGI without the information bottleneck optimization component. As shown in Table 2, by comparing MAGI and MAGI-IB, we can see that the removal of the information bottleneck optimization module causes a drop in performance under adversarial attacks and noise, which demonstrates graph information bottleneck optimization is able to enhance the robustness and effectiveness of communication learning.

Moreover, when comparing MAGI and MAGI-VD, we can see that the removal of the value decomposition module leads to only a slight performance decline. It is worth noting that the MAGI module can be flexibly integrated with various value decomposition methods. These experimental results demonstrate that information bottleneck optimization can enhance the robustness and effectiveness of communication learning under adversarial attacks and that the value decomposition module can further promote action collaboration and policy learning among agents.

Parameters (Q5). To explore the effect of different hyperparameters on performance. We first conduct ablations on feature noise ratios in the 2c3s5z and MMM3 scenarios of SMAC. As shown in Figure 4, with different feature noise ratios ($\eta \in \{0.5, 1.0, 1.5\}$), MAGI consistently performs better than other GNN-based MACRL methods. Especially, as the η is large ($\eta = 1.5$), the performance of other methods is significantly affected, while MAGI remains stable, which demonstrates that IB optimization of MAGI makes the com-

Scenarios	MAGI-IB	MAGI-VD	MAGI
MMM2	60.07±5.75	79.93±3.64	**82.87±2.93**
MMM3	42.77±4.62	52.28±3.25	**57.84±3.96**
8m vs 9m	68.12±3.83	87.29±2.40	**92.55±1.78**
1o2r vs 4r	71.24±5.49	84.64±3.56	**86.01±3.04**
2c3s5z	69.35±4.18	89.47±1.90	**93.02±1.45**

Table 2: Win rate with different variants on several scenarios

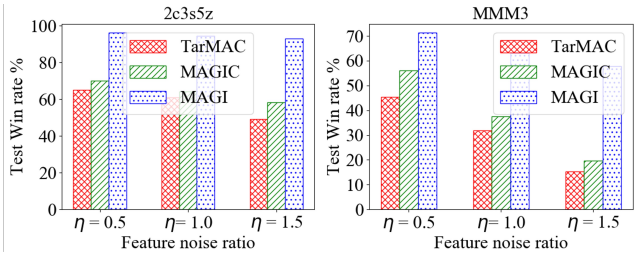

Figure 4: Win rate of different methods with increasing feature noise ratio η in 2c3s5z and MMM3 of SMAC.

munication learning more robust under adversarial attacks and noise perturbations.

To explore the effect of different λ on the performance, we conduct ablations on different scenarios of SMAC under adversarial attacks and noise perturbations. As shown in Table 3, with different λ ($\lambda \in \{0.05, 0.10, 0.15\}$), MAGI achieve the best performance with $\lambda = 0.15$ in MMM3 scenario and with $\lambda = 0.10$ in other scenarios. Thus, for the sake of consistency, we set the $\lambda = 0.10$ for all scenarios.

Scenarios	$\lambda = 0.05$	$\lambda = 0.10$	$\lambda = 0.15$
MMM2	80.17±3.02	**82.87±2.93**	81.94±2.85
MMM3	53.06±3.77	57.84±3.96	**59.27±3.71**
8m vs 9m	88.20±2.61	**92.55±1.78**	90.94±2.15
1o2r vs 4r	83.06±4.34	**86.01±3.04**	85.39±3.16
2c3s5z	92.17±1.51	**93.02±1.45**	91.06±1.83

Table 3: Win rate with different λ on several scenarios

Besides, we evaluate the robustness and effectiveness of different variants with increasing feature noise ratio ($\eta \in \{1.5, 2.0, 2.5\}$). As shown in Figure 5, from the comparison results of the three variants, it can be seen that the performance of the MAGI-IB drops sharply with the increase of the feature noise ratio η. In contrast, the MAGI is excellent for stability. The effect of MAGI is the best, the variance of MAGI-IB is the largest, and meanwhile, the performance of MAGI-IB is the worst.

Generality (Q6). Adversarial attacks can be generally categorized as modifying features and modifying edges. Therefore, in the previous experiment, we selected two adversarial attacks (GN+PGD) to generate perturbations at the same time for experiments: Adding Gaussian noise (modifying features) and PGD (Xu et al. 2019) (modifying edges). Furthermore, we utilize two other complicated adversarial attacks (IG-JSMA (Wu et al. 2019) and GUA (Zang et al. 2021)) for experiments to verify the generality of the proposed method and the results are shown in Table 4. We utilize IG-JSMA to add adversarial perturbations on both the agent features and edges. We leverage GUA to change edges by flipping the connections between the anchor agents (refer to (Zang et al. 2021) for more details) and the target agent. As shown in Table 4, MAGI always achieves the best performance compared with other baseline methods under various adversarial attack methods, which demonstrates the generality of the proposed method.

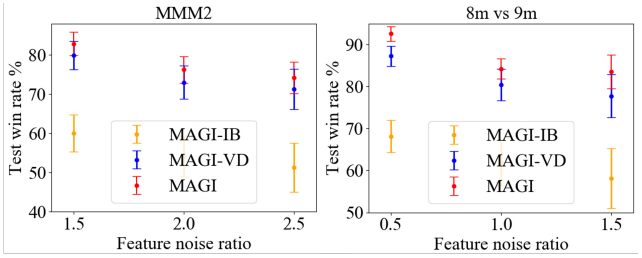

Figure 5: Win rate of different variants with increasing feature noise ratio η.

Methods	GN+PGD	IG-JSMA	GUA
TarMAC	45.42±8.03	39.24±6.58	42.38±7.25
MAGIC	56.06±6.32	49.05±5.40	51.76±4.92
MAGI	**71.34±4.26**	**67.26±4.08**	**65.13±3.52**

Table 4: Win rate with different adversarial attacks in MMM3 of SMAC

Conclusions

In this paper, we present a GNN-based MACRL method that incorporates graph information bottleneck optimization. Our approach aims to achieve robust and efficient communication learning by leveraging two information-theoretic regularizers. These regularizers minimize the mutual information between the communication message and the agent features, while simultaneously maximizing the MI between the message representation and the action selection. Empirical results from diverse multi-agent scenarios demonstrate that our proposed method outperforms other baseline approaches significantly. To the best of our knowledge, this work represents the initial exploration of learning robust communication using graph information bottleneck optimization in the MACRL domain. We believe that the proposed method holds promise in establishing efficient communication in large-scale multi-agent systems, even in the presence of adversarial attacks and noise perturbations. In the future, it would be worthwhile to apply our proposed method to real-world large-scale multi-agent cooperative tasks.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant no.62276265, no.61976216.

References

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

Du, W.; and Ding, S. 2021. A survey on multi-agent deep reinforcement learning: from the perspective of challenges and applications. *Artificial Intelligence Review*, 54(5): 3215–3238.

Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In *ICLR*, 1263–1272.

Hostallero, W. J. K. D. E.; Son, K.; Kim, D.; and Qtran, Y. Y. 2019. Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *ICML*, 5887–5896.

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

Jiang, J.; and Lu, Z. 2018. Learning attentional communication for multi-agent cooperation. In *NeurIPS*, 7254–7264.

Liu, Y.; Wang, W.; Hu, Y.; Hao, J.; Chen, X.; and Gao, Y. 2020. Multi-agent game abstraction via graph attention neural network. In *AAAI*, 7211–7218.

Ma, J.; Ding, S.; and Mei, Q. 2020. Towards more practical adversarial attacks on graph neural networks. 4756–4766.

Niu, Y.; Paleja, R. R.; and Gombolay, M. C. 2021. Multi-Agent Graph-Attention Communication and Teaming. In *AAMAS*, 964–973.

Park, H.; and Neville, J. 2019. Exploiting Interaction Links for Node Classification with Deep Graph Neural Networks. In *IJCAI*, 3223–3230.

Qiu, D.; Wang, J.; Wang, J.; and Strbac, G. 2021. Multi-Agent Reinforcement Learning for Automated Peer-to-Peer Energy Trading in Double-Side Auction Market. In *IJCAI*, 2913–2920.

Rashid, T.; Samvelyan, M.; Schroeder, C.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2018. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *ICML*, 4295–4304.

Ryu, H.; Shin, H.; and Park, J. 2020. Multi-agent actor-critic with hierarchical graph attention network. In *AAAI*, 7236–7243.

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. In *AAMAS*, 2186–2188.

Sheng, J.; Wang, X.; Jin, B.; Yan, J.; Li, W.; Chang, T.-H.; Wang, J.; and Zha, H. 2020. Learning structured communication for multi-agent reinforcement learning. *arXiv preprint arXiv:2002.04235*.

Sukhbaatar, S.; Fergus, R.; et al. 2016. Learning multiagent communication with backpropagation. In *NeurIPS*, 2244–2252.

Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W. M.; Zambaldi, V.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo, J. Z.; Tuyls, K.; et al. 2020. Value-decomposition networks for cooperative multiagent learning. In *AAMAS*, 2085–2087.

Tishby, N.; and Zaslavsky, N. 2015. Deep learning and the information bottleneck principle. In *ITW*, 1–5.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *ICLR*.

Wang, J.; Ren, Z.; Liu, T.; Yu, Y.; and Zhang, C. 2021. QPLEX: Duplex dueling multi-agent Q-learning. In *ICLR*, 1–27.

Wang, R.; He, X.; Yu, R.; Qiu, W.; An, B.; and Rabinovich, Z. 2020. Learning efficient multi-agent communication: An information bottleneck approach. In *ICML*, 9908–9918.

Wang, T.; Liao, R.; Ba, J.; and Fidler, S. 2018. Nervenet: Learning structured policy with graph neural networks. In *ICLR*, 1–26.

Wang, T.; Wang, J.; Zheng, C.; and Zhang, C. 2019. Learning nearly decomposable value functions via communication minimization. In *ICLR*, 1–15.

Wu, H.; Wang, C.; Tyshetskiy, Y.; Docherty, A.; Lu, K.; and Zhu, L. 2019. Adversarial examples on graph data: Deep insights into attack and defense. In *IJCAI*, 4816–4823.

Wu, T.; Ren, H.; Li, P.; and Leskovec, J. 2020. Graph information bottleneck. In *NeurIPS*, 20437–20448.

Xu, K.; Chen, H.; Liu, S.; Chen, P.-Y.; Weng, T.-W.; Hong, M.; and Lin, X. 2019. Topology attack and defense for graph neural networks: An optimization perspective. In *IJCAI*, 3961–3967.

Yang, S.; Yang, B.; Kang, Z.; and Deng, L. 2021. IHG-MA: Inductive heterogeneous graph multi-agent reinforcement learning for multi-intersection traffic signal control. *Neural networks*, 139(2): 265–277.

Zang, X.; Xie, Y.; Chen, J.; and Yuan, B. 2021. Graph universal adversarial attacks: A few bad actors ruin graph learning models. In *IJCAI*.

Zhang, M.; Wang, X.; Zhu, M.; Shi, C.; Zhang, Z.; and Zhou, J. 2022. Robust heterogeneous graph neural networks against adversarial attacks. In *AAAI*, 4363–4370.

Zheng, L.; Yang, J.; Cai, H.; Zhou, M.; Zhang, W.; Wang, J.; and Yu, Y. 2018. Magent: A many-agent reinforcement learning platform for artificial collective intelligence. In *AAAI*, 8222–8223.