Robust Multi-Agent Communication With Graph Information Bottleneck Optimization

Shifei Ding, *Member, IEEE*, Wei Du, Ling Ding, Jian Zhang, Lili Guo, *Member, IEEE*, and Bo An, *Member, IEEE*

***Abstract*—Recent research on multi-agent reinforcement learning (MARL) has shown that action coordination of multi-agents can be significantly enhanced by introducing communication learning mechanisms. Meanwhile, graph neural network (GNN) provides a promising paradigm for communication learning of MARL. Under this paradigm, agents and communication channels can be regarded as nodes and edges in the graph, and agents can aggregate information from neighboring agents through GNN. However, this GNN-based communication paradigm is susceptible to adversarial attacks and noise perturbations, and how to achieve robust communication learning under perturbations has been largely neglected. To this end, this paper explores this problem and introduces a robust communication learning mechanism with graph information bottleneck optimization, which can optimally realize the robustness and effectiveness of communication learning. We introduce two information-theoretic regularizers to learn the minimal sufficient message representation for multi-agent communication. The regularizers aim at maximizing the mutual information (MI) between the message representation and action selection while minimizing the MI between the agent feature and message representation. Besides, we present a MARL framework that can integrate the proposed communication mechanism with existing value decomposition methods. Experimental results demonstrate that the proposed method is more robust and efficient than state-of-the-art GNN-based MARL methods.**

***Index Terms*—Graph neural network, multi-agent reinforcement learning, graph information bottleneck optimization, communication learning.**

I. Introduction

REINFORCEMENT learning has achieved notable progress in tackling single-agent complicated tasks, such as robot navigation [1] and vehicle control [2]. Meanwhile, many real-world scenarios such as intelligent traffic systems [3], contain not only one agent but usually multiple agents involved in learning cooperative tasks. Such scenarios naturally lead to the prevalence of multi-agent reinforcement learning (MARL), where crucial challenges include scalability and non-stationarity (caused by partial observability limitation). Recently, the paradigm of centralized training with decentralized execution (CTDE) has been comprehensively adopted to tackle these challenges, in which agents execute only based on decentralized local information but their policies can be trained with additional global information [4], [5]. Value function decomposition methods [6], [7], [8], [9] further explore this paradigm, which decomposes the global value function into a set of individual value functions. However, these methods still fail to perform well in scenarios where action coordination is required, because the partial observability during the execution period will increase the uncertainty of one agent to actions of other agents, resulting in difficulty in action coordination.

Allowing agents to learn to communicate efficiently can strengthen action coordination and eventually enhance the quality of learned policies. To this end, numerous multi-agent communicative reinforcement learning (MACRL) methods [10], [11], [12], [13], [14], [15] have been proposed, which allow agents to exchange messages during the decentralized execution phase, such as local individual observations or agent feature embeddings. Recently, graph neural network (GNN) has been developed as an efficient representation learning method, which can process the topological information and attribute information of the graph-structured data to feature representation learning for the final tasks. GNN has been utilized to build communication learning mechanisms of MACRL, which generally regards agents and communication channels as nodes and edges in the graph, with the action selection corresponding to node labeling. Many state-of-the-art MACRL methods fall into this GNN-based communication paradigm, such as TarMAC [16].

The crucial task of GNN-based MACRL is to learn efficient message representation that carries both agent feature information and topological relationship information for action selection and coordination. However, the message representation learning of recent GNN-based MACRL methods still encounter some

Manuscript received 14 April 2023; revised 7 November 2023; accepted 27 November 2023. Date of publication 29 November 2023; date of current version 3 April 2024. This work was supported by the National Natural Science Foundation of China under Grants 62276265, 61976216, and 62206297. Recommended for acceptance by W. Zhang. *(Corresponding authors: Wei Du; Ling Ding.)*

Shifei Ding is with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China, and also with the Mine Digitization Engineering Research Center of Ministry of Education of the People's Republic of China, Xuzhou 221116, China (e-mail: dingsf@cumt.edu.cn).

Wei Du is with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China (e-mail: 1394471165@qq.com).

Ling Ding is with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China (e-mail: 414211048@qq.com).

Jian Zhang and Lili Guo are with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China (e-mail: zhangjian10231209@cumt.edu.cn; liliguo@cumt.edu.cn).

Bo An is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: boan@ntu.edu.sg).

Digital Object Identifier 10.1109/TPAMI.2023.3337534

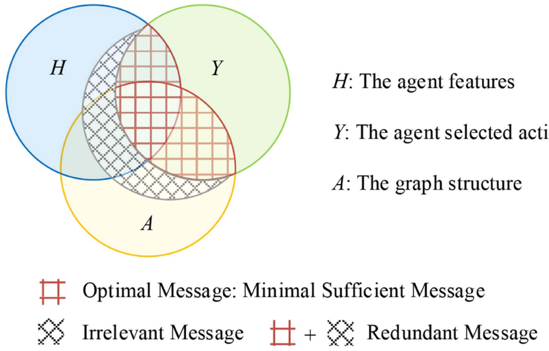

Fig. 1. MAGI aims at optimizing the communication message representation to extract sufficient and minimal information within the $\mathcal{D} = (A, H)$ to select optimal actions Y.

challenges. On the one side, the message representation aggregates the information of features of neighbor agents, which may contain useless information that negatively affects action selection. On the other side, GNN-based MACRL relies on the edges of the graph to implement the messages exchange among agents, which makes it susceptible to adversarial attacks and noise perturbations on the agent features and topological structure.

As shown in Fig. 1, $\mathcal{D} = (A, H)$ carries information from both the graph topological structure A and agent feature embeddings H. If communication message representation 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 [17].

Therefore, in this paper, we rethink what is and how to obtain an "optimal" communication message representation that contains sufficient information to facilitate action coordination while avoiding the impact of adversarial attacks. For general representation learning, the Information Bottleneck (IB) [18] introduces an essential principle: the optimal representation should contain sufficient and minimal information useful for the final 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 Fig. 1. However, extending the IB principle to GNN-based MACRL methods to obtain the optimal message encounters two main challenges as follows. 1) The topological structure information contained in the graph structure is essential for communication message representation, but this information is discrete and therefore it is difficult to optimize. 2) Previous IB-based representation learning methods generally restricts input data to satisfy independent and identically distributed condition (i.i.d.). Therefore, the IB principle is difficult to implement in MACRL, because conditions are not supported for the agent feature.

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, we can ensure that learned communication messages is both sufficient (efficiently facilitating the action selection of agents) and minimal (message representations do not contain unnecessary information). In addition, to address the second challenge caused by the non-i.i.d. property of agent features, we leverage the local-dependence assumption of agent features to capture information from graph topological structure A and agent feature embeddings H hierarchically. 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 [19] to GNN-based MACRL methods, which achieve efficient and robust multi-agent communication learning.
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 [20] and MAgent [21]. Experimental results demonstrate that MAGI is more robust and efficient than the state-of-the-art MACRL methods.

II. Related Work

A. Multi-Agent Communicative Reinforcement Learning

Multi-agent communicative reinforcement learning methods aim at achieving consensus and cooperation of multiple agents through communication learning. Agents need to enhance action coordination by learning to communicate with other agents and process the message representations they receive. Previous MACRL methods can be divided into two main categories. The first category focuses on generating meaningful messages for the message senders. One straightforward approach is to treat raw local observations or the local information history as messages. For example, NDQ [11] aims to generate minimal messages for different teammates, allowing them to learn decomposable value functions. NDQ optimizes the message generator utilizing two information-theoretic regularizers to achieve expressive communication.

The second category of work aims to efficiently extract the most useful messages at the receiver's end. For example, MASIA [12] explicitly addresses the optimization of multiple received messages and introduces two self-supervised representation objectives. These objectives aim to make the received information representation both abstract of the true states and predictive of the multi-step future information. TarMAC [15] achieves targeted communication through a soft-attention mechanism, in which the sender broadcasts a key encoding the agents' properties, and the receiver processes all received messages for a weighted sum of messages to make decisions.

The most relevant works to the idea of this paper are NDQ [11] and MASIA [12], both of which aim to achieve efficient communication. NDQ aims to ensure communication is both expressive (effectively reducing the uncertainty of action-value functions of agents) and succinct (only sending necessary and useful information). MASIA focuses on ensuring communication is both compact (high information density) and sufficient (containing a rich amount of information). Our method is designed to ensure communication is sufficient (efficiently facilitating the action selection of agents), minimal (not containing unnecessary information), and robust (not vulnerable to adversarial attacks and noise). Different from NDQ and MASIA, we aim at the redundancy and efficiency issue of communication information in the GNN-based MACRL methods and consider the robustness of these methods, which NDQ and MASIA do not pay attention to.

B. GNN-Based MACRL

In contrast to the above methods, GNN-based MACRL methods generally utilize neighboring communication architecture. In this architecture, agents communicate with neighbors simultaneously, which can reduce communication costs. Meanwhile, the powerful information aggregation ability and representation learning capability provided by GNN can make GNN-based MACRL methods suitable for large-scale agent scenarios. DGN [22] first extends GNN to MACRL methods and adopts neighboring architecture, where multiple rounds of communications are utilized to widen the receptive field. NerveNet [23] utilizes GNN to represent the policy of the agent and propagate information over the graph structure of agents and then predicts actions for the agents.

HAMA [24] adopts a hierarchical GNN based on a pre-defined hierarchical graph and attention mechanism to facilitate agents to capture interrelations. However, the predefined fixed grouping scheme adopted by HAMA restricts its adaptability in dynamic settings. GA2NET [25] adopts a two-stage attentional mechanism to model the multi-agent setting, which can infer whether there is interaction between two agents and then evaluate the importance of interaction. TarMAC [16] utilizes GNN with a soft attention mechanism to learn whom to receive messages and what messages to pass. MAGIC [15] presents an attentional GNN to operate on the constructed graph for multiple rounds of communication, which can tackle the issue of when to communicate.

Recent GNN-based MACRL methods have successfully promoted action coordination by efficient communication learning, which aggregates information on agent features and models interactions among agents by diverse graph neural networks. Despite significant progress in GNN-based MACRL, if these methods encounter adversarial attacks and noise perturbations, multi-agent action coordination will rapidly disintegrate. The issue that how to achieve robust communication under adversarial attacks and noise perturbations has been largely unstudied. Our work intends to provide a promising way to address this, which extends the graph information bottleneck optimization to GNN-based MACRL methods.

C. Value Function Decomposition

For effectiveness and scalability in multi-agent environments, CTDE has become a prevalent MARL paradigm. Value function factorization methods further explore the CTDE paradigm based on the Individual-Global-Max (IGM) principle [9], which is an essential assumption that ensures consistency between local greedy action selections and joint action selections. VDN [7] adopts linear value factorization to ensure the sufficient condition for the IGM principle. Due to its excellent scalability, this simple linear architecture has become very prevalent in MARL and has inspired many subsequent approaches. QMIX [8] presents a monotonic mixing network to enhance the expressiveness of the decomposed function class. QTRAN [6] aims at realizing the entire IGM function class, however, its proposed goal is computationally intractable and necessitates two additional soft regularizers to approximate IGM, where the strict IGM guarantee is not guaranteed. QPLEX [9] extends the IGM principle into the dueling network architecture, however, it has potential limitations in terms of scalability. Our work utilizes different value decomposition methods in the proposed MARL framework to demonstrate the flexibility of the proposed method.

D. Adversarial Attack

Sun et al. [26] 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 modifying node features, modifying (adding/deleting) edges.

1) Modifying node features: Adversarial attacks can slightly modify the node features while maintaining the structure of the graph. Ma et al. [27] add adversarial perturbation on node features and set a novel local constraint on node access, prohibiting perturbation on the top node, and only a few nodes can be disturbed, which is more realistic in practice. Wu et al. [28] present an integrated gradients-based attack method that adds perturbations on both the node features and edges. 2) Modifying Edges: Adversarial attacks can add or delete edges to existing nodes with a given total action budget. Xu et al. [29] present a novel gradient-based attack method that only changes a small number of edges, including addition and deletion. However, this method can lead to a noticeable decrease in the performance of downstream tasks. Zang et al. [30] find a set of anchor nodes to mislead the classification of all nodes in the graph by flipping the connections between the anchors and the target node.

E. Adversarial Attacks and Defenses in MACRL

Robust single-agent RL has been investigated from different perspectives [31], and recently, the issue of adversarial attacks and defenses in MACRL has gained significant attention. Researchers have been exploring various approaches to achieve robust communication in the face of adversarial attacks. Tu et al. [32] focus on exploring adversarial attacks specifically in the multi-agent setting where perturbations are introduced to learned intermediate representations. Mitchell et al. [33] propose a method that utilizes a Gaussian Process-based probabilistic module to calculate posterior probabilities, determining the truthfulness of each partner. Xue et al. [34] propose a two-stage message filter that involves learning an anomaly detector and a message reconstructor to recover the true messages. They employ two populations of defenders and attackers for training, aiming to improve the generalizability of defense mechanisms. Sun et al. [35] introduce the first certifiable defense in MARL against communication attacks, which considers a particularly strong threat model in which half of communication messages can be arbitrarily compromised. Nevertheless, these previous methods generally focus on the malicious perturbations on the received message itself, but the adversarial attacks can also be the perturbations on the communication structure and agent feature, which has not been paid attention to in the previous work. This paper tries to solve this problem by using the GNN-based MACRL methods as a starting point.

III. Methodology

A. Problem Fomulation

The MARL problems can be generally modeled as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP), which can be characterized by a tuple $< I, S, U, P, R, O >$. I represents the finite set of agents indexed from 1 to n. S represents the finite set of states. O denotes the joint observations, in which $o_i \in O$ is local observation of agent i. U represents the finite space of joint actions. At each timestep, the agent i takes its action a_i depending on its local observation o_i. $a = (a_1, \dots, a_n) \in U$ denotes a joint action. The state changes depending on the Markovian transition $P : S \times U \to S$. $R : S \times U \to R$ represents the reward function. The overall task of the agent i is maximizing its total discounted reward $R_i = \sum_{t=0}^{T} \gamma^t r_i^t$, where $\gamma \in [0, 1]$ means a discount factor. Agents aim to learn an optimal joint policy $\pi(\tau, a)$ to maximize the global value $Q_{tot}^{\pi}(\tau, a) = \mathbb{E}_{s,a}[\sum_{t=0}^{\infty} \gamma^t R(s, a)]$, where τ means the joint observation history.

In our work, we model the multi-agent system utilizing a graph $G = (V, E, H)$ with n nodes, where $V = \{1, 2, \dots n\}$ means the agent/node set, $E \subseteq V \times V$ represents the edge set, $H \in \mathbb{R}^{n \times f}$ means the agent features/node attributes. We utilize $A \in \mathbb{R}^{\ n \times n}$ to denote the adjacency matrix of G, if $(i, j) \in E$, then $A_{ij} = 1$, otherwise $A_{ij} = 0$. We leverage $d(i, j)$ to denote the shortest path distance of two agents $i, j (\in V)$. Thus, the input feature information data for the GNN-based communication learning mechanism 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 utilized to facilitate the agent to select actions Y. The subscript with an agent $i \in V$ is leveraged to denote the affiliation with the agent i. For example, the communication message representation of agent i is denoted by $M_{H,i} = m_i$ and its corresponding optimal action selection is represented by $Y_i = a_i$.

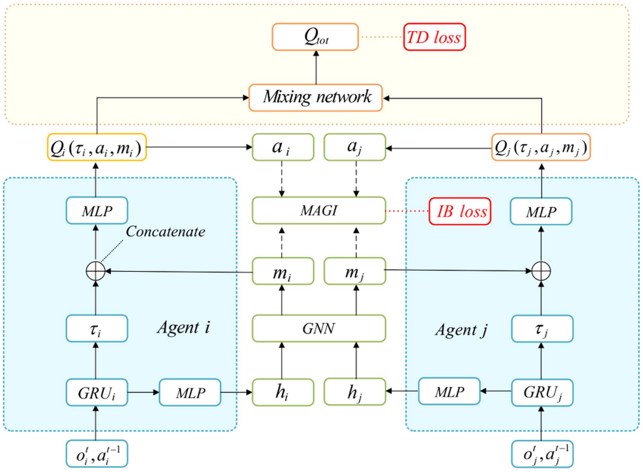

Fig. 2. Overall framework of MAGI.

B. Overall Framework

The overall framework of the proposed method is illustrated in Fig. 2. For agent i, it obtains local observation o_i and utilizes gated recurrent unit (GRU) and multi-layer perceptron (MLP) to generate the agent feature h_i. Then we fed feature h_i into the communication learning module, where graph neural network is leveraged to produce the communication message representation m_i. In our work, we utilize graph attention network (GAT) [36] as the GNN structure in the communication learning module. We fuse the feature information of neighbor agents and topological structure information by stacking multiple GNN layers. High-level communication message representation of each agent can be extracted by multiple rounds of communication. Besides, we utilize graph information bottleneck optimization to further obtain the optimal message representations, which are minimal and sufficient for decision making.

Then, for agent i, 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 Fig. 2, we fed local action-values of all agents into the mixing network and finally obtained the estimation of global action-value Q_{tot}. In this work, we select the mixing networks proposed by VDN [7], QMIX [8], and QPLEX [9], and it can be flexibly replaced by other mixing networks of existing value factorization methods. For instance, in the case of QMIX, the mixing network consists of MLPs. However, the weights and biases of the mixing network are obtained from a hypernetwork to ensure adherence to the monotonicity constraints. In the next section, we will introduce the multi-agent communication via graph information bottleneck optimization (MAGI) module of this framework in detail.

C. Multi-Agent Communication via Graph Information Bottleneck

The objective of the Multi-Agent communication via Graph Information bottleneck (MAGI) module is to obtain the optimal communication message representation, which is minimal and sufficient. To achieve this objective, MAGI necessitates the communication message M_H to maximize the mutual information for selecting optimal actions Y and minimize the information from the feature $\mathcal{D} = (A, H)$. The objective of MAGI is represented in (1), where $I(\cdot;\cdot)$ means the mutual information,

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

In the MAGI, we introduce a local-dependence assumption [19] for the graph-structured agent features: for each agent i, given the neighbor-related agents within certain hops, the other agent features are independent of the agent feature of agent i. We utilize this assumption to restrict the optimal communication message representations space Ω, which makes the optimization of the MAGI objective more tractable. Concretely, we utilize $\mathbb{P}(M_H \mid \mathcal{D})$ to hierarchically iterate communication message representations to model the correlation of agent features, where $\mathbb{P}(\cdot)$ means joint probabilistic distribution function. Each GNN layer corresponds to a round of communication message exchange among the agent with its neighboring agents.

In each message exchange round l, we utilize the local-dependence assumption. The communication message of each agent can be refined by aggregating the agent features of its neighboring agents, w.r.t a graph structure M_A^l. Thus, $(M_A^l)_{1 \le l \le L}$ is obtained by adjusting the original graph topological structure A locally, which can essentially control the message flow from graph topological structure A. In the end, we utilize high-level communication message representation M_H^l to achieve action selection and action coordination. Therefore, the objective of MAGI optimization can be reduced to the following representation:

$$\min_{\mathbb{P}(M_H^l \mid \mathcal{D}) \in \Omega} \mathrm{MAGI}_\beta(\mathcal{D}, Y; M_H^l)$$
$$\triangleq \left[-I(Y; M_H^L) + \beta I(\mathcal{D}; M_H^L)\right], \quad (2)$$

In this equation, we just need to optimize the distributions of these two series: $\mathbb{P}(M_H^l \mid M_H^{l-1}, M_A^l)$ and $\mathbb{P}(M_A^l \mid M_H^{l-1}, A)$. These distributions are easier to optimize because they have local dependence between agents. We utilize the simplistic MAGI principle and some appropriate parameterization of $\mathbb{P}(M_H^l \mid M_H^{l-1}, M_A^l)$ and $\mathbb{P}(M_A^l \mid M_H^{l-1}, A)$, however, the computation of $I(Y; M_H^L)$ and $I(\mathcal{D}; M_H^L)$ of (2) is still intractable. Thus, we should introduce variational bound regularizers on the two terms of (2), which facilitates the optimization of the final objective. As demonstrated in [18], we can derive the lower bound of term $I(Y; M_H^L)$ and the upper bound of term $I(\mathcal{D}; M_H^L)$, which are shown in (3) and (4), respectively. For any distributions $\mathbb{V}_1(Y_i \mid M_{H,i}^L)$ for agent $i \in V$ and $\mathbb{V}_2(Y)$, the lower bound of term $I(Y; M_H^L)$ is shown in (3),

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

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

$$I(\mathcal{D}; M_H^L) \le I\left(\mathcal{D}; (M_H^l)_{l \in S_H} \cup (M_A^l)_{l \in S_A}\right)$$
$$\le \sum_{l \in S_A} \mathrm{AIB}^l + \sum_{l \in S_H} \mathrm{HIB}^l, \quad (4)$$

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

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

Equation (4) illustrates that we should select a set of random variables with index S_H and S_A to satisfy the conditional independence between M_H^L and $\mathcal{D}$. S_H and S_A have the following attributes: 1) If the maximum index in S_H is l, the S_A includes all the integers of $[l+1, L]$. 2) $S_H \neq \emptyset$. To utilize MAGI principle, we should model $\mathbb{P}(M_A^l \mid M_H^{l-1}, A)$ and $\mathbb{P}(M_H^l \mid M_H^{l-1}, M_A^l)$. Next, we select certain variational distributions $\mathbb{V}(M_H^l)$ and $\mathbb{V}(M_A^l)$ to estimate the corresponding AIB^l and HIB^l for the regularization, and some $\mathbb{V}_1(Y_i \mid M_{H,i}^L$. and $\mathbb{V}_2(Y)$ to obtain the lower bound in (3). Thus, we can acquire the upper bound to optimize the objective by plugging (3) and (4) into the (2).

In this work, we leverage graph attention network (GAT) [36] as the GNN structure in the communication learning module. In each layer of the graph attention network, MAGI first needs to refine the graph structure of agents utilizing the attention weights to acquire M_A^l and next refine communication message representations M_H^l by propagating M_H^{l-1} over M_A^l. For the neighboring agents sampling, we leverage categorical distribution and regard attention weights as the parameters of the categorical distributions, which can sample the topological structure of the refined graph to capture topological structure information. Next, we extract k neighboring agents with alternatives from the built pool of agents V_{ic} for agent i, in which V_{ic} contains the agents whose shortest path distance to agent i is c. We leverage $\mathcal{C}$ to be the upper constraint of c to guarantee the assumption of local dependence. Next, we sum-pool the neighboring agents and utilize the output to calculate the parameters of the Gaussian distribution, in which the refined agent features will be sampled.

Then, we aim to optimize the parameters of the MAGI module. The bounds of term $I(Y; M_H^L)$ in (3) and term $I(\mathcal{D}; M_H^L)$ in (4) should be specified, further the bound of MAGI optimization objective in (2) should be calculated. In order to characterize AIB^l in (4), we take assumption that $\mathbb{V}(M_A^l)$ satisfies the non-informative distribution. Concretely, we leverage the uniform categorical distribution: $M_A \sim \mathbb{V}(M_A)$, $M_{A,i} = \cup_{d=1}^{\mathcal{T}} \{j \in V_{ic} \mid j \overset{\text{iid}}{\sim} \text{Cat}(\frac{1}{|V_{ic}|})\}$ and $M_{A,i} \perp M_{A,j}$ if $i \neq j$. We leverage $\text{Cat}(\phi)$ to denote the categorical distribution with parameter ϕ, which corresponds to different categories of probabilities and therefore $\|\phi\|_1 = 1$. After ϕ_{ic}^l is calculated, we can acquire an empirical estimation of AIB^l,

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

which is instantiated as follows,

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

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

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

Thus, we can select appropriate index set S_H, S_A that guarantee the assumption in (4) and utilize substitution:

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

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

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

Plugging (10) and (11) into (2), the MAGI optimization objective can be obtained to train our proposed communication learning module.

Except for the MAGI optimization constraints on the communication message representations learning in the communication module, all the parameters in other modules in the framework are updated by minimizing the TD loss L_{TD}. In the end, TD loss is shown in (12),

$$L_{TD} = \left[r + \gamma \max_{a'} Q_{tot} \left(\tau', a'; \theta^- \right) - Q_{tot}(\tau, a; \theta) \right]^2, \quad (12)$$

where θ means all the parameters in the proposed method and θ^- means the parameters of target network. The overall optimization objective of the proposed method is shown as follows,

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

where λ denotes a hyper-parameter that can be fine-tuned to achieve a trade-off between the graph information bottleneck optimization loss $L_{IB} = \text{MAGI}_\beta(\mathcal{D}, Y; M_H^l)$ and the TD loss L_{TD}. The hyper-parameter λ is set to be $\lambda = 0.1$ depending on experimental results.

Algorithm 1: Instantiation of MAGI.

```
Input: $o_i \in O$ and $a_i^{t-1} \in A$ of agent $i$.
Initialize: The weights of networks $W$, the number of
  neighboring agents to be sampled $k$, the integral
  limitation to impose local dependence $\mathcal{C}$.
Output: $Q_{tot}$.
1: for each timestep $t \in T$ do
2:   for each agent $i \in N$ do
3:     % During the decentralized execution phase
4:     Process agent feature $h_i$ by GRU and MLP
5:     Build data matrix $\mathcal{D} = (A, H)$ based on $h_i$
6:     Fed $\mathcal{D}$ to $L-$ layers GAT
7:     for layers $= 1, \ldots, L$ do
8:       $\tilde{M}_{H,i}^{l-1} \leftarrow \sigma(M_{H,i}^{l-1}) W^l$
9:       build sets $V_{ic} \leftarrow \{j \in V \mid d(i,j) = c\}$
10:      for $c \in [\mathcal{C}]$ do
11:        $\phi_{ic}^l \leftarrow \text{softmax}\{(\tilde{M}_{H,i}^{l-1} \oplus \tilde{M}_{H,j}^{l-1}) a^T\}$
12:        $M_{A,i}^l \leftarrow \cup_{c=1}^{\mathcal{C}} \{j \in V_{ic} \mid j \overset{\text{iid}}{\sim} \text{Cat}(\phi_{ic}^{l-1})\}$
13:      end for
14:      $\bar{M}_{H,i}^l \leftarrow \sum_{j \in M_{H,i}^l} \tilde{M}_{A,i}^{l-1}$
15:      $\mu_i^l \leftarrow \bar{M}_{H,i}^l[0 : f']$
16:      $\sigma_i^{2\ l} \leftarrow \text{softplus}(\bar{M}_{H,i}^l[f' : 2f'])$
17:      $M_{H,i}^l \sim \text{Gaussian}(\mu_i^l, \sigma_i^{2\ l})$
18:    end for
19:    Acquire final message representation $m_i = M_{H,i}^L$
20:    Compute action-value $Q_i$ based on $m_i$ and $\tau_i$
21:    $a_i^t \leftarrow \pi(Q_i)(\epsilon-$ greed)
22:    Store episode history $\tau_i$ and $a_i^t$ in replay buffer
23:    % During centralized training phase
24:    Fed $Q_i$ to mixing network and acquire $Q_{tot}$
25:    Minimize loss function according to (13)
26:    Update weights $W$ of all networks
27:  end for
28: end for
```

The detail of the instantiation of the proposed method is shown in Algorithm 1. Steps 1-4 show the feature processing phase. For each timestep $t \in T$, each agent $i \in N$ obtains its observation o_i and uses GRU and MLP to generate its initial agent feature. Steps 5-6 describe the process of building a graph using agent features. The core process of message representation optimization is instantiated in Steps 7-18. In each layer, MAGI first refines the graph structure utilizing the attention weights to obtain $M_{A,i}^l$ (Steps 8-13) and then refines message representations $M_{H,i}^l$ by propagating $M_{H,i}^{l-1}$ over $M_{A,i}^l$ (Steps 14-18).

Step 14 is the sum-pooling operation of the neighbor agents, and the output will be leveraged to calculate the parameters for a Gaussian distribution in which the refined message representations will be sampled. MAGI relies loosely on the graph structure because A is only utilized to decide the potential neighbor agents for each agent, and performs message passing

Fig. 3. Illustration of the SMAC scenarios.

depending on M_A. This property makes our models very robust against structural attacks/perturbations. Our models also keep robustness to the feature perturbations, which is similar to other IB-based methods. In Steps 19-22, agent i calculates the local value based on the obtained message m_i and then selects action a_i^t. In Steps 23-27 describe the centralized training phase, which uses the loss function in (13).

IV. Experiments

In this section, we select StarCraft II Multi-Agent Challenge (SMAC) [20] and MAgent [21] as our benchmarks. We conduct various experiments on these complicated benchmarks to answer: *Q1*: Are the proposed method and other GNN-based MACRL methods vulnerable to adversarial attacks and noise perturbations? *Q2*: Can MAGI optimization enhance the robustness of communication learning under adversarial attacks and noise perturbations? *Q3*: Can MAGI achieve efficient communication that facilitates action coordination? *Q4*: Whether MAGI can be applied to large-scale multi-agent scenarios? *Q5*: Which component of the proposed framework contributes to the performance improvement? *Q6*: Can MAGI flexibly integrate with existing value function factorization methods? *Q7*: How do hyper-parameters influence the performance of MAGI? *Q8*: How does MAGI perform well under more complicated adversarial attack methods? We choose QMIX [8], TarMAC [16], and MAGIC [15] as baselines. QMIX proposes a mixing network where the joint action value is estimated through a nonlinear combination of individual action values. TarMAC employs GAT with a soft attention mechanism to determine the extent to which a message is processed. MAGIC constructs a communication graph and utilizes GAT for multiple iterations of communication. It is worth noting that TarMAC, MAGIC, and MAGI all leverage GAT as a GNN structure in the communication learning module. The experiments are all conducted on GPU Nvidia RTX 2080Ti with Pytorch.

A. Environments

SMAC: StarCraft Multi-Agent Challenge (SMAC) is constructed on StarCraft II, which is a prevalent strategy game. SMAC consists of various StarCraft II micro scenarios, the example is shown in Fig. 3. These scenarios are complicated, in which the ally agents are necessitated to learn no less than one micromanagement technology to defeat the enemy agents. In the scenarios of SMAC, all the ally agents are trained with the MARL methods, and all the enemy agents are controlled by the built-in AI.

The action space of agents contains 4 actions: move, attack, no-op, and stop. The range of attack is set to 6. At each time step, the ally agents can take attack action to fire enemy agents and acquire a global reward. Besides, the ally agents can acquire an extra reward for killing the enemy agent and winning the game. In particular, we fine-tune the default experimental setting to make the ally agents more difficult to coordinate actions. We reduce the scope of vision of the ally agent from 9 to 2. In addition, we select several complicated scenarios as shown in Table I. The details of the 5 scenarios are provided as follows.

MMM2 and MMM3 are two super-hard scenarios, which contain Marines, Marauders, and Medivac. Each type of agent in these scenarios has its own unique attributes and skills. The ally agents succeed only when each agent performs to the best of its ability and coordinates the actions of other agents. Medivac has the healing ability that can provide treatment to the injured ally agents. To win the battle, Ally agents have to learn to communicate with other agents, such as sending their health situation to the Medivac.

1o2r versus 4r scenario includes 1 Overseer and 2 Roaches of ally agents. The target of the Overseer is to find the 4 Reapers of enemy units, and the goal of the 2 Roaches is to reach the positions of Reapers and try to defeat the Reapers. The Overseers and the Reapers are randomly generated in the scenario, and the enemy Roaches are randomly generated at other points. Considering that only the Overseers can get information about the location of enemies, the Roaches must learn to aggregate the information of neighbor Overseers in order to win the battle effectively.

2c3s5z scenario contains Colossus, Stalkers, and Zealots both for ally agents and enemy agents. Ally agents must learn many tactics, such as utilizing Zealots to intercept enemy Zealots in order to protect Stalkers from serious damage. It is easier for agents to learn such strategies and coordinate actions under efficient information interaction. In 8 m versus 9 m scenario, both the ally and enemy agents are Marines.

MAgent: MAgent is a MARL platform that aims to support the tasks that necessitate hundreds of agents. We select the Battle scenario of MAgent to conduct experiments. The battle scenario contains K ally agents and Z enemy agents. The target of ally agents is to learn to defeat all enemy agents. Each agent can take two actions: move or attack and the range of action is 4. The individual enemy agent is more capable than an individual ally agent, therefore the ally agent must develop strategies to fight cooperatively with other ally agents.

Since the Battle scenario tends to lose balance after the death of agents, we stochastically add a new ally agent or enemy agent to the scenario to keep balance. MAGI and other baselines are first trained under the same setting of $K = 40$ and $Z = 24$. In the Battle scenario, an agent can acquire a positive reward of +5 while attacking the enemy. An agent can obtain a negative reward of -2 and -0.01, when it is killed by the enemy agent or hits a blank grid, respectively.

TABLE I
SCENARIOS OF SMAC

Scenarios	Ally agents	Enemy agents	Agents Type	Challenge
MMM2	1 Medivac, 2 Marauders, 7 Marines	1 Medivac, 2 Marauders, 8 Marines	Heterogeneous	Super Hard
MMM3	1 Medivac, 2 Marauders, 7 Marines	1 Medivac, 2 Marauders, 9 Marines	Heterogeneous	Super Hard
2c3s5z	2 Colossus, 3 Stalkers, 5 Zealots	2 Colossus, 3 Stalkers, 5 Zealots	Heterogeneous	Easy
1o2r vs 4r	1 Overseer, 2 Roaches	4 Reapers	Heterogeneous	Hard
8m vs 9m	8 Marines	9 Marines	Homogeneous	Hard

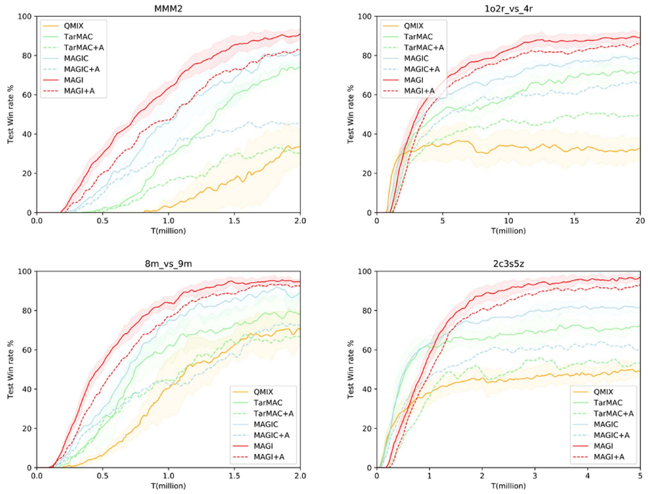

Fig. 4. Learning curves of different methods under adversarial attacks and noise ($\eta = 1.5$).

TABLE II
PERFORMANCE OF DIFFERENT METHODS ON BATTLE SCENARIO

Methods	Kills	Mean reward	K/D ratio
TarMAC	156	0.53±0.07	1.16±0.19
MAGIC	172	0.62±0.04	1.57±0.26
MAGI	**237**	**0.87±0.02**	**2.02±0.16**

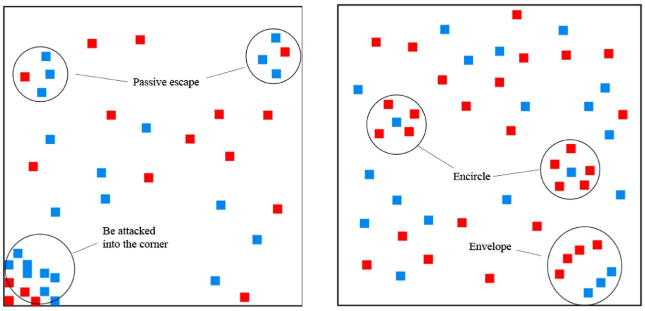

Fig. 5. Illustration of representative behaviors of MAGI (right) and other baselines (left) in Battle scenario.

B. Performance

Robustness (Q1, Q2): To verify whether MAGI and the existing GNN-based MACRL methods are vulnerable to adversarial attacks and noise perturbations, we produce random perturbations and inject them into adjacency matrix A and the agent features H. We add independent Gaussian noise to agent features H with increasing amplitude. Concretely, we utilize the average value of the maximum feature of each agent to be the reference amplitude ϕ. Then we add Gaussian noise $\eta \cdot \phi \cdot \epsilon$ into each agent feature, where $\epsilon \sim N(0, 1)$, and η means the feature noise ratio. We verify the robustness of MAGI and other GNN-based MACRL methods with $\eta = 1.5$. In the ablation section, we verify the robustness under different parameters with $\eta \in \{0.5, 1, 1.5\}$.

To produce adversarial attacks on the adjacency matrix A, we first try to randomly drop edges among agents with the graph structure. Nevertheless, experimental results show that this straightforward strategy can not significantly affect the communication efficiency of the GNN-based MACRL methods. Thus, we utilize projected gradient descent (PGD) [29] to produce attacks rather than stochastically drop edges. Fig. 4 shows the learning curves of MAGI and other GNN-based MACRL methods on four scenarios of SMAC. The solid line shows the mean win rate of the method without adversarial attack and noise perturbations, and the corresponding shaded area means a 95% confidence interval. The dashed line illustrates the mean win rate of the method with adversarial attacks and noise perturbations.

We can draw several conclusions from Fig. 4 as follows. 1) MAGI performs significantly better than other baseline methods in all scenarios. 2) By comparing QMIX and GNN-based MACRL methods, we can see that the GNN-based communication learning mechanism of these methods can indeed facilitate cooperation among the agents and improve performance. 3) 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 fuses the communication learning mechanism with the value factorization method. 4) Furthermore, the performance of TarMAC+A and MAGIC+A degrades significantly compared with TarMAC and MAGIC respectively, which demonstrate the other existing GNN-based MACRL methods are susceptible to adversarial attacks. 5) Compared with MAGI, MAGI+A shows only a slight performance degradation, which demonstrates that MAGI is robust under adversarial attacks because graph information bottleneck optimization can improve the robustness of communication learning.

Effectiveness (Q3): The performance of different methods under adversarial attacks in the Battle scenario is shown in Table II. For convenience, in later experiments, methods without the "+A" suffix (such as MAGI) also default to that method under adversarial attack. As shown in Table II, MAGI performs better than other baselines, in terms of kill number (kill number of enemy agents), kill-death rates (kill number of enemy agents divided by death number of ally agents), and mean reward.

Fig. 5 illustrates the representative actions of agents trained by different methods in the Battle scenario. As shown in Fig. 5,

TABLE III
MEAN REWARD OF METHODS WITH DIFFERENT NUMBERS OF AGENTS

Methods	$K = 20$	$K = 30$	$K = 40$	$K = 50$
TarMAC	0.47±0.13	0.51±0.14	0.53±0.07	0.58±0.09
MAGIC	0.58±0.09	0.60±0.08	0.62±0.04	0.66±0.06
MAGI	**0.71±0.04**	**0.80±0.03**	**0.87±0.02**	**0.90±0.01**

TABLE IV
WIN RATE WITH DIFFERENT VARIANTS ON SMAC

Scenarios	MAGI-IB	MAGI-VD	MAGI
MMM2	60.07±4.75	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 V
PERFORMANCE OF DIFFERENT VARIANTS ON BATTLE

Methods	Kills	Mean reward	K/D ratio
MAGI-IB	176	0.65±0.09	1.62±0.21
MAGI-VD	214	0.81±0.13	1.83±0.22
MAGI	**237**	**0.87±0.02**	**2.02±0.16**

with noise perturbations and adversarial attacks, MAGI-trained agents learn some techniques of action coordination, such as encircling. For the individual enemy agent, agents trained with MAGI can learn to coordinate their actions to surround it and defeat it. For the group of enemy agents, agents trained with MAGI can learn the ability to coordinate attacks on one side of the group.

In contrast, other GNN-based MACRL methods learn suboptimal strategies under adversarial attacks and noise perturbations, such as gathering in a corner to avoid being attacked by enemy agents. This behavior demonstrates the communication learning of these methods becomes ineffective under perturbations. This may be due to that MAGI adopts the graph information bottleneck to learn robust and minimal sufficient message representations, which realize the efficient communication that promotes the action coordination and policy learning of agents.

Scalability (Q4): To verify that MAGI can be extended to large-scale multi-agent scenarios, we compared MAGI and other baselines under perturbations in the Battle scenario with the different number of agents ($K \in \{20, 30, 40, 50\}$). As shown in Table III, the scalability ability of the MAGI is evidenced by the fact that MAGI always performs best compared to the baselines as the number of agents increases. We believe that this combination of communication learning and value factorization can be used as a general paradigm to solve large-scale multi-agent problems.

C. Ablations

Contribution (Q5): Furthermore, we evaluate the contributions of each component of MAGI. The MAGI contains two critical components: the value decomposition (VD) component and the IB optimization component. Thus, we design two variants of MAGI: 1) MAGI-VD is MAGI without the VD component, and 2) MAGI-IB is MAGI without the IB component. As shown in Tables IV and V, by comparing MAGI and MAGI-IB, we can see that the removal of the IB component causes a drop in performance under adversarial attacks and noise perturbations. Moreover, when comparing MAGI and MAGI-VD, we can see that the removal of the VD module also results in a slight decline in performance. These experimental results show that IB optimization can significantly improve the robustness and effectiveness of communication learning under adversarial attacks and that the VD module can further facilitate action coordination and policy learning.

TABLE VI
WIN RATE WITH DIFFERENT REGULARIZERS ON SMAC

Scenarios	MAGI w/ IB_{r_1}	MAGI w/ IB_{r_2}	MAGI-IB
MMM2	74.36±3.52	67.25±3.13	60.07±4.75
MMM3	50.47±4.28	51.13±4.62	42.77±4.62
8m vs 9m	83.06±2.28	76.38±2.71	68.12±3.83
1o2r vs 4r	84.52±3.91	73.17±4.36	71.24±5.49
2c3s5z	79.16±3.20	83.47±2.45	69.35±4.18

TABLE VII
WIN RATE WITH DIFFERENT REGULARIZERS ON BATTLE

Methods	Kills	Mean reward	K/D ratio
MAGI-IB	176	0.65±0.09	1.62±0.21
MAGI w/ IB_{r_1}	195	0.77±0.05	1.80±0.15
MAGI w/ IB_{r_2}	189	0.72±0.06	1.75±0.18
MAGI	237	0.87±0.02	2.02±0.16

TABLE VIII
WIN RATE WITH DIFFERENT VALUE DECOMPOSITION METHODS

Scenarios	VDN	MAGI(VDN)	QPLEX	MAGI(QPLEX)
MMM2	38.14±7.36	76.53±3.82	42.19±6.36	80.26±3.17
MMM3	29.38±5.15	61.58±4.13	36.38±4.39	64.85±3.74
8m vs 9m	61.50±4.16	85.31±2.64	68.34±3.30	84.15±2.19
1o2r vs 4r	67.17±4.27	82.05±3.27	71.93±3.38	83.28±2.05
2c3s5z	65.29±4.73	90.45±2.04	69.27±3.51	87.31±2.11

To further evaluate the contributions of the two regularizers, we use $IB_{r_1} = I(Y; M_H^L)$ to represent the IB loss of regularizer 1 and $IB_{r_2} = I(\mathcal{D}; M_H^L)$ to denote the IB loss of regularizer 2. Therefore, MAGI w/ IB_{r_1} represents MAGI only with regularizer 1, MAGI w/ IB_{r_2} represents MAGI only with regularizer 2, MAGI-IB denotes MAGI without regularizers. As shown in Tables VI and VII, both regularizers contribute to robust communication. In scenario MMM2, 8 m versus 9 m and 1o2r versus 4r, regularizer 1 is more effective than regularizer 2. In other scenarios, regularizer 2 is more effective.

Flexibility (Q6): To demonstrate that MAGI can be flexibly fused with any value factorization methods, we conduct experiments by integrating it with other popular value factorization methods VDN and QPLEX. The fused methods MAGI (VDN) and MAGI (QPLEX) are evaluated on various scenarios of SMAC. As shown in Table VIII, MAGI (VDN) and MAGI (QPLEX) perform better than the original value factorization methods VDN and QMIX, which indicates the flexibility of the proposed method. Besides, the proposed method can remove the value decomposition module for more general tasks, such as mixed cooperative and competitive tasks.

Parameters (Q7): 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.

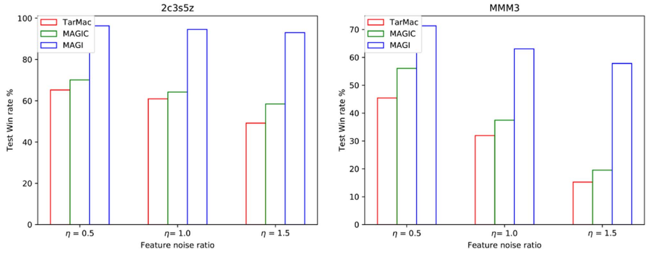

Fig. 6. Win rate with increasing feature noise ratio η.

TABLE IX
WIN RATE WITH DIFFERENT λ ON SMAC

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

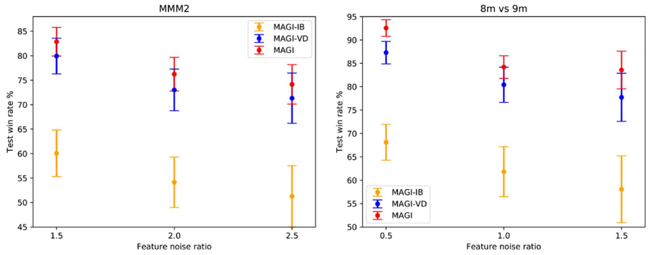

Fig. 7. Win rate of different variants with increasing feature noise ratio η.

As shown in Fig. 6, 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 communication learning more robust under adversarial attacks and noise perturbations.

To evaluate 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 IX, 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 in SMAC.

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 Fig. 7, 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 (Q8): 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 [29] (modifying edges). Furthermore, we utilize two other complicated adversarial attacks (IG-JSMA [28] and GUA [30]) for experiments to verify the generality of the proposed method, and the results are shown in Table X. We utilize IG-JSMA [28] to add adversarial perturbations on both the agent features and edges. We leverage GUA [30] to change edges by flipping the connections between the anchor agents. Please refer to [28], [29], [30] for more details. As shown in Table X, MAGI always achieves the best performance compared with other baseline methods under various adversarial attack methods, which demonstrates the generality of the proposed method.

TABLE X
WIN RATE WITH DIFFERENT ADVERSARIAL ATTACKS ON MMM3

Methods	GN+PGD [29]	IG-JSMA [28]	GUA [30]
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 XI
FIXED HYPER-PARAMETERS OF MAGI FOR ALL ENVIRONMENTS

Hyper-parameter	Value
GNN layers	3
GNN hidden dimension	64
MLP layers	2
RNN type	GRU
RNN hidden dimension	64
Mixing network	QMIX
Mixing embedding dimension	32
Hypernet layers (QMIX)	2
Hypernet embedding dimension (QMIX)	64
Learning rate	1e-5
Discount factor	0.99
Batch size	200
Optimizer	Adam

D. Details of Model Hyper-Parameters

In this section, we provide the details about our experiment settings for the convenience of reproducibility. Table XI describes the fixed hyper-parameters of all the experiment benchmarks.

V. CONCLUSION

In this paper, we introduce the graph information bottleneck optimization into the GNN-based MACRL method. The proposed method achieves robust and efficient communication learning by two information-theoretic regularizers, which minimizes the MI between the communication message and the agent features and simultaneously maximizes the MI between the message representation and the action selection. Experimental results in various multi-agent scenarios demonstrate that the proposed method significantly outperforms other baselines and can be flexibly fused with existing value factorization methods to promote action coordination.

To the best knowledge, this work is the first attempt at learning robust communication via graph information bottleneck optimization in the MACRL domain. We believe it is a promising way to establish efficient communication of large-scale multi-agent systems under adversarial attacks and noise perturbations. In the future, it is worthwhile to apply the proposed method to real-world large-scale multi-agent systems. It is important to acknowledge that our current method is specifically designed for discrete action and fully cooperative scenarios. It is not applicable to continuous action environments or mixed competitive cooperative scenarios. However, our future work aims to develop a more versatile and robust communication model that can be adapted to a broader range of scenarios.

References

[1] H. Li, Q. Zhang, and D. Zhao, "Deep reinforcement learning-based automatic exploration for navigation in unknown environment," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 6, pp. 2064–2076, Jun. 2020.

[2] N. Wang, Y. Gao, H. Zhao, and C. K. Ahn, "Reinforcement learning-based optimal tracking control of an unknown unmanned surface vehicle," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 7, pp. 3034–3045, Jul. 2021.

[3] C. Sun, W. Liu, and L. Dong, "Reinforcement learning with task decomposition for cooperative multiagent systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2054–2065, May 2021.

[4] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3826–3839, Sep. 2020.

[5] W. Du and S. Ding, "A survey on multi-agent deep reinforcement learning: From the perspective of challenges and applications," *Artif. Intell. Rev.*, vol. 54, no. 5, pp. 3215–3238, 2021.

[6] K. Son, D. Kim, W. J. Kang, D. Hostallero, and Y. Yi, "Learning to factorize with transformation for cooperative multi-agent reinforcement learning," in *Proc. 36th Int. Conf. Mach. Learn.*, Long Beach, CA, USA, 2019, pp. 5887–5896.

[7] P. Sunehag et al., "Value-decomposition networks for cooperative multi-agent learning," in *Proc. 19th Int. Conf. Auton. Agents Multiagent Syst.*, Auckland, New Zealand, 2020, pp. 2085–2087.

[8] T. Rashid, M. Samvelyan, C. Schroeder, G. Farquhar, J. Foerster, and S. Whiteson, "QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, 2018, pp. 4295–4304.

[9] J. Wang, Z. Ren, T. Liu, Y. Yu, and C. Zhang, "QPLEX: Duplex dueling multi-agent Q-learning," in *Proc. 7th Int. Conf. Learn. Representations*, 2019, pp. 1–27.

[10] S. Sukhbaatar and R. Fergus, "Learning multiagent communication with backpropagation," in *Proc. 30th Adv. Neural Inf. Process. Syst.*, Barcelona, Spain, 2016, pp. 2244–2252.

[11] T. Wang, J. Wang, C. Zheng, and C. Zhang, "Learning nearly decomposable value functions via communication minimization," 2019, *arXiv: 1910.05366*.

[12] C. Guan et al., "Efficient multi-agent communication via self-supervised information aggregation," in *Proc. 36th Adv. Neural Inf. Process. Syst.*, 2022, pp. 1020–1033.

[13] J. Jiang and Z. Lu, "Learning attentional communication for multi-agent cooperation," in *Proc. 32nd Adv. Neural Inf. Process. Syst.*, Montreal, Canada, 2018, pp. 7254–7264.

[14] R. Wang, X. He, R. Yu, W. Qiu, B. An, and Z. Rabinovich, "Learning efficient multi-agent communication: An information bottleneck approach," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 9908–9918.

[15] Y. Niu, R. Paleja, and M. C. Gombolay, "Multi-agent graph-attention communication and teaming," in *Proc. 20th Int. Conf. Auton. Agents Multiagent Syst.*, London, U.K., 2021, pp. 964–973.

[16] A. Das et al., "TarMAC: Targeted multi-agent communication," in *Proc. 36th Int. Conf. Mach. Learn.*, Long Beach, CA, USA, 2019, pp. 1538–1546.

[17] W. Zhou, D. Chen, J. Yan, Z. Li, H. Yin, and W. Ge, "Multi-agent reinforcement learning for cooperative lane changing of connected and autonomous vehicles in mixed traffic," *Auton. Intell. Syst.*, vol. 2, no. 5, pp. 1–11, 2022.

[18] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *Proc. IEEE Inf. Theory Workshop*, 2015, pp. 1–5.

[19] T. Wu, H. Ren, P. Li, and J. Leskovec, "Graph information bottleneck," in *Proc. 34th Adv. Neural Inf. Process. Syst.*, 2020, pp. 20437–20448.

[20] M. Samvelyan et al., "The StarCraft multi-agent challenge," in *Proc. 18th Int. Conf. Auton. Agents Multiagent Syst.*, Montreal, Canada, 2019, pp. 2186–2188.

[21] L. Zheng, J. Yang, H. Cai, M. Zhou, W. Zhang, and Y. Jun, "MAgent: A many-agent reinforcement learning platform for artificial collective intelligence," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 8222–8223.

[22] J. Jiang, C. Dun, and Z. Lu, "Graph convolutional reinforcement learning for multi-agent cooperation," in *Proc. 6th Int. Conf. Learn. Representations*, 2018, pp. 1–10.

[23] T. Wang, R. Liao, J. Ba, and S. Fidler, "NerveNet: Learning structured policy with graph neural networks," in *Proc. 6th Int. Conf. Learn. Representations*, Vancouver, Canada, 2018, pp. 1–26.

[24] H. Ryu, H. Shin, and J. Park, "Multi-agent actor-critic with hierarchical graph attention network," in *Proc. 34th AAAI Conf. Artif. Intell.*, New York, USA, 2020, pp. 7236–7243.

[25] Y. Liu, W. Wang, Y. Hu, J. Hao, X. Chen, and Y. Gao, "Multi-agent game abstraction via graph attention neural network," in *Proc. 34th AAAI Conf. Artif. Intell.*, New York, USA, 2020, pp. 7211–7218.

[26] M. Zhang, X. Wang, M. Zhu, X. Shi, Z. Zhang, and J. Zhou, "Robust heterogeneous graph neural networks against adversarial attacks," in *Proc. 36th AAAI Conf. Artif. Intell.*, 2022, pp. 4363–4370.

[27] J. Ma, S. Ding, and Q. Mei, "Towards more practical adversarial attacks on graph neural networks," 2020, *arXiv: 2006.05057*.

[28] H. Wu, C. Wang, Y. Tyshetskiy, A. Docherty, K. Lu, and L. Zhu, "Adversarial examples for graph data: Deep insights into attack and defense," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 4816–4823.

[29] K. Xu et al., "Topology attack and defense for graph neural networks: An optimization perspective," 2019, *arXiv: 1906.04214*.

[30] X. Zang, Y. Xie, J. Chen, and B. Yuan, "Graph universal adversarial attacks: A few bad actors ruin graph learning models," in *Proc. 30th Int. Joint Conf. Artif. Intell.*, 2021, pp. 1–7.

[31] H. Zhang et al., "Robust deep reinforcement learning against adversarial perturbations on state observations," in *Proc. 34th Adv. Neural Inf. Process. Syst.*, Vancouver, Canada, 2020, pp. 21024–21037.

[32] T. James, W. Tsunhsuan, M. Sivabalan, R. Mengye, and U. Raquel, "Adversarial attacks on multi-agent communication," in *Proc. IEEE/CVF 18th Int. Conf. Comput. Vis.*, Montreal, Canada, 2021, pp. 7768–7777.

[33] M. Rupert, B. Jan, and P. Amanda, "Gaussian process based message filtering for robust multi-agent cooperation in the presence of adversarial communication," 2020, *arXiv: 2012.00508*.

[34] W. Xue, W. Qiu, B. An, Z. Rabinovich, S. Obraztsova, and C. Yeo, "Misspoke or mis-lead: Achieving robustness in multi-agent communicative reinforcement learning," 2021, *arXiv:2108.03803*.

[35] Y. Sun et al., "Certifiably robust policy learning against adversarial multi-agent communication," in *Proc. 11th Int. Conf. Learn. Representations*, 2022, pp. 1–30.

[36] P. Veličković, G. Cucurull, and A. Casanova, "Graph attention networks," in *Proc. 6th Int. Conf. Learn. Representations*, Vancouver, Canada, 2018, pp. 1–12.

Shifei Ding (Member, IEEE) received the PhD degree from the Shandong University of Science and Technology, Qingdao, China, in 2004, and the postdoctoral degree from the Key Laboratory of Intelligent Information Processing (IIP), Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2006. He is a professor and PhD supervisor with the China University of Mining and Technology. His research interests include intelligent information processing, pattern recognition, machine learning, data mining, and granular computing.

Wei Du received the BS degree from Shandong University, Jinan, China, in 2016, and the MS degree from the China University of Mining and Technology, Xuzhou, China, in 2020. He is currently working toward the PhD degree with the China University of Mining and Technology, his supervisor is Prof. Shifei Ding. His research interests include machine learning and reinforcement learning.

Ling Ding received the BS and MS degrees from the Asia Pacific University of Technology and Innovation, Kuala Lumpur, Malaysia, in 2017 and 2019, respectively. She is currently working toward the PhD degree with Tianjin University, her supervisor is Dr. Di Jin. Her research interests include deep learning, graph machine learning, clustering, etc.

Jian Zhang received the PhD degree from the China University of Mining and Technology, Xuzhou, China, in 2019. He is a lecturer with the China University of Mining and Technology. His research interests include deep learning, structure design and optimization of neural networks, and image synthesis.

Lili Guo (Member, IEEE) received the PhD degree from Tianjin University, Tianjin, China, in 2020. She is a lecturer with the China University of Mining and Technology. Her research interests include deep learning, multimodal emotional computing, etc.

Bo An (Member, IEEE) received the PhD degree in computer science from the University of Massachusetts, Amherst, MA, USA, in 2010. He is an associate professor with Nanyang Technological University and chairman of the president's Committee. His research interests include artificial intelligence, multi-agent systems, reinforcement learning, game theory, and optimization.