Graph-GRPO: Stabilizing Multi-Agent Topology Learning via Group Relative Policy Optimization

Yueyang Cang[1], **Xiaoteng Zhang**[1], **Erlu Zhao**[1], **Zehua Ji**[1],
Yuhang Liu[1], Yuchen He[1], Zhiyuan Ning[1], Yijun Chen[1],
Wenge Que[2,⋆], Li Shi[1,⋆]

[1]Tsinghua University
[2]Donghua University
⋆Corresponding author

Abstract

Optimizing communication topology is fundamental to the efficiency and effectiveness of Large Language Model (LLM)-based Multi-Agent Systems (MAS). While recent approaches utilize reinforcement learning to dynamically construct task-specific graphs, they typically rely on single-sample policy gradients with absolute rewards (e.g., binary correctness). This paradigm suffers from severe gradient variance and the credit assignment problem: simple queries yield non-informative positive rewards for suboptimal structures, while difficult queries often result in failures that provide no learning signal. To address these challenges, we propose Graph-GRPO, a novel topology optimization framework that integrates Group Relative Policy Optimization. Instead of evaluating a single topology in isolation, Graph-GRPO samples a group of diverse communication graphs for each query and computes the advantage of specific edges based on their relative performance within the group. By normalizing rewards across the sampled group, our method effectively mitigates the noise derived from task difficulty variance and enables fine-grained credit assignment. Extensive experiments on reasoning and code generation benchmarks demonstrate that Graph-GRPO significantly outperforms state-of-the-art baselines, achieving superior training stability and identifying critical communication pathways previously obscured by reward noise.

1 Introduction

The rapid evolution of Large Language Models (LLMs) has catalyzed the development of Multi-Agent Systems (MAS), where collaborative agents demonstrate emergent capabilities in complex reasoning, coding, and decision-making tasks (Li et al., 2023; Xi et al., 2025; Hong et al., 2024; Qian et al., 2024). A growing number of studies suggest that the communication topology—the structural framework governing information exchange among agents—is a key determinant of system performance (Zhuge et al., 2024; Qian et al., 2025; Liu et al., 2024). While early approaches relied on static, predefined structures such as chains, trees, or fully connected graphs (Wei et al., 2022; Wu et al., 2023; Yao et al., 2024), recent state-of-the-art methods like EIB-LEARNER (Shen et al., 2025) have shifted towards dynamically generating task-specific topologies. EIB-LEARNER, for instance, provides a causal framework to balance "error suppression" and "insight propagation", demonstrating that adaptive connectivity is the key to robust collaboration (Zhang et al., 2025b; Wang et al., 2024).

Although topology modeling has advanced, the optimization paradigms for these discrete structures remain suboptimal. Most leading methods currently rely primarily on standard Reinforcement Learning (RL) techniques, such as the REINFORCE algorithm (Williams, 1992), with single-sample estimation and absolute rewards (e.g., binary correctness) (Ouyang et al., 2022). This optimization strategy suffers from two fundamental limitations:

1. **High Gradient Variance:** The difficulty of queries in datasets is often uneven (Wang et al., 2023). For simple queries, a wide range of suboptimal topologies may fortuitously yield correct answers (reward=1), introducing significant noise into the policy update. As illustrated in Figure 1, standard methods indiscriminately reinforce these redundant edges. Conversely, for difficult queries, the system often fails regardless of the topology (reward=0), leading to vanishing gradients.
2. **The Credit Assignment Problem:** When a topology succeeds, standard methods attribute the reward equally to all edges in the graph (Sutton and Barto, 2018). This coarse-grained feedback fails to distinguish which specific connections were causally responsible for the

Findings of the Association for Computational Linguistics: ACL 2026, pages 20222–20231
July 2-7, 2026

success and which were redundant, hindering the model's ability to learn precise structural patterns.

To address these challenges, we propose Graph-GRPO (Graph-based Group Relative Policy Optimization), a novel framework that fundamentally stabilizes topology learning. Inspired by recent advances in LLM reasoning optimization (Shao et al., 2024; Schulman et al., 2017), we shift the objective from maximizing absolute rewards to maximizing relative advantage within a sampled group. Specifically, for each query, Graph-GRPO samples a group of diverse communication topologies. Instead of evaluating each graph in isolation, we compute a baseline from the group's average performance and derive the advantage of each specific edge.

This group-based approach offers a dual benefit. First, it acts as a dynamic normalization mechanism: for simple tasks where the average performance is high, only topologies that perform better than average (e.g., more efficient) are reinforced, effectively filtering out "easy-win" noise. Second, it enables fine-grained credit assignment: edges that consistently appear in the higher-performing topologies within a group are assigned positive advantages, while those associated with failure are suppressed. By integrating this mechanism, Graph-GRPO allows the model to identify critical communication pathways that were previously obscured by reward noise.

In summary, our contributions are as follows:

- We identify the limitations of absolute-reward optimization in MAS topology learning and propose Graph-GRPO, the first framework to apply Group Relative Policy Optimization to discrete structure search.
- We introduce a fine-grained edge scoring mechanism that solves the credit assignment problem by leveraging relative advantages across a group of sampled topologies.
- Extensive experiments on six benchmarks, including MMLU and HumanEval, demonstrate that Graph-GRPO significantly outperforms EIB-LEARNER, achieving superior stability and convergence efficiency.

2 Related Work

2.1 LLM-based Multi-Agent Systems

The paradigm of utilizing multiple Large Language Models (LLMs) to tackle complex tasks has garnered significant attention (Xi et al., 2025; Wang et al., 2024). Early frameworks such as CAMEL (Li et al., 2023) and AutoGen (Wu et al., 2023) demonstrated that role-playing agents can collaboratively solve problems through dialogue. However, these initial systems typically operated on predefined, static communication structures, such as chain-of-thought sequences (Wei et al., 2022), star topologies (centralized manager), or fully connected graphs (Hong et al., 2024; Qian et al., 2024). While effective for specific scenarios, static topologies lack the flexibility to adapt to the varying complexity of user queries, often leading to either redundant communication costs or insufficient information exchange (Liu et al., 2024; Zhuge et al., 2024).

2.2 Communication Topology Optimization

To overcome the rigidity of static structures, recent research has focused on learning adaptive communication topologies. Approaches like AgentPrune (Zhang et al., 2025a) and AgentDropout (Wang et al., 2025) employ pruning techniques to remove redundant connections from a full graph. More advanced generative methods, such as G-Designer (Zhang et al., 2025b) and EIB-LEARNER (Shen et al., 2025), utilize Graph Neural Networks (GNNs) to construct task-specific topologies from scratch. EIB-LEARNER, in particular, introduced a causal perspective to balance error suppression and insight propagation.

Despite these advances in topology modeling, the optimization strategy remains largely unchanged: these methods predominantly rely on standard policy gradient algorithms (e.g., REINFORCE) with absolute, binary rewards (Williams, 1992). As noted in our analysis, this single-sample optimization paradigm suffers from high variance and poor credit assignment, especially when dealing with the diverse difficulty levels inherent in reasoning datasets. Our work builds upon the architectural strengths of EIB-LEARNER but fundamentally redesigns the optimization process to ensure stability and robustness.

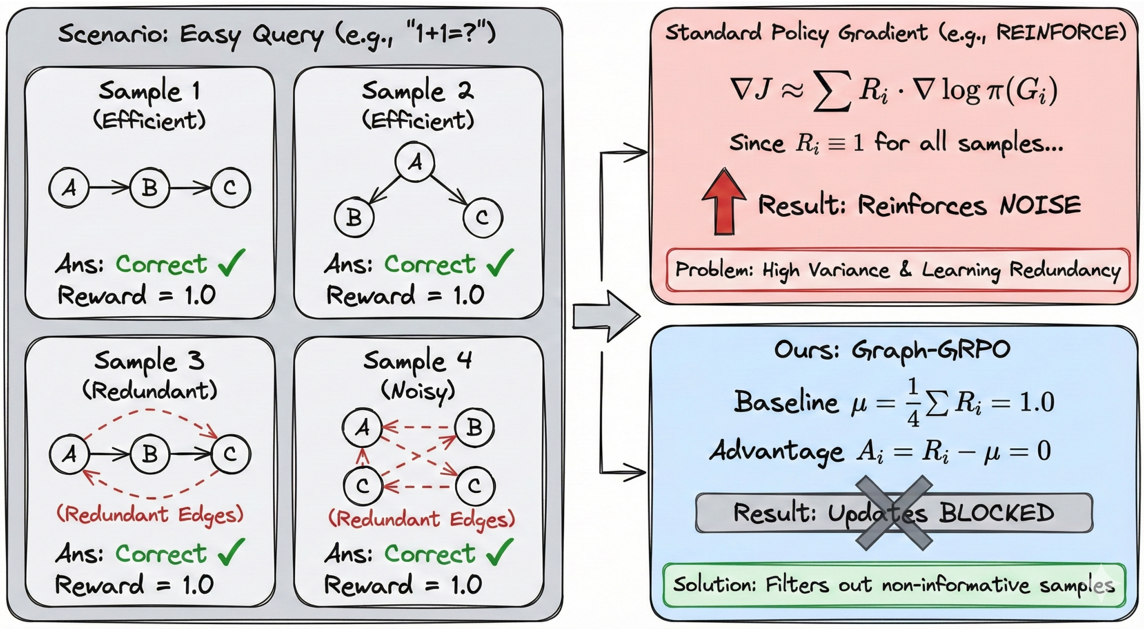

Figure 1: **Motivation Analysis: The Trap of Non-Informative Batches in Easy Queries.** The figure illustrates a scenario where a task is simple enough that diverse sampled topologies (Samples 1–4, ranging from efficient chains to dense structures with redundant edges) all yield correct answers and identical rewards ($R_k = 1$). (Top Right) Standard policy gradient methods like REINFORCE use raw rewards. Since $R_k \equiv 1$ across the entire group, the gradient estimation indiscriminately reinforces *all* sampled edges, including noise and redundancies (e.g., extra edges in S3 & S4), leading to suboptimal convergence. (Bottom Right) Our proposed Graph-GRPO addresses this by incorporating a group baseline μ. In such uniform-reward scenarios, μ equals individual rewards, resulting in near-zero advantage ($A_{ij} \approx 0$). This mechanism effectively blocks parameter updates from non-informative batches, preventing the model from learning redundant structures from noise.

2.3 Reinforcement Learning for Reasoning

Reinforcement Learning (RL) has become a foundational approach for aligning LLMs with human preferences and logical constraints (Ouyang et al., 2022). While Proximal Policy Optimization (PPO) (Schulman et al., 2017) is widely used, its dependence on a value network (Critic) introduces significant memory overhead and training instability. Recently, Group Relative Policy Optimization (GRPO), introduced in DeepSeekMath (Shao et al., 2024), has emerged as a powerful alternative. By eliminating the Critic and normalizing rewards within a sampled group, GRPO effectively reduces gradient variance for mathematical reasoning tasks.

However, existing applications of GRPO are largely confined to continuous text generation domains. To the best of our knowledge, our work is the first to adapt the group-relative mechanism to the domain of discrete structure search in multi-agent systems, addressing the unique challenges of edge-level credit assignment in graph topology learning.

3 Methodology

In this section, we present the proposed Graph-GRPO framework. The overall architecture is depicted in Figure 2. We first outline the policy network architecture used to generate communication topologies, incorporating strict structural constraints to ensure logical progression. Then, we detail our core contribution: a group relative optimization mechanism that performs fine-grained credit assignment by estimating the marginal success rate of each edge, effectively eliminating the need for a value network (Critic).

3.1 Policy Network Architecture

We strictly followed the architectural design proposed in G-Designer (Zhang et al., 2025b) as our policy backbone. The framework utilizes a Graph Neural Network (GNN) to parameterize the communication topology and consists of two primary modules: a Node Encoder and a Structure Generator.

Node Representation. Given a task query Q and a set of agents $\mathcal{V} = \{v_1, \ldots, v_N\}$, we first initial-

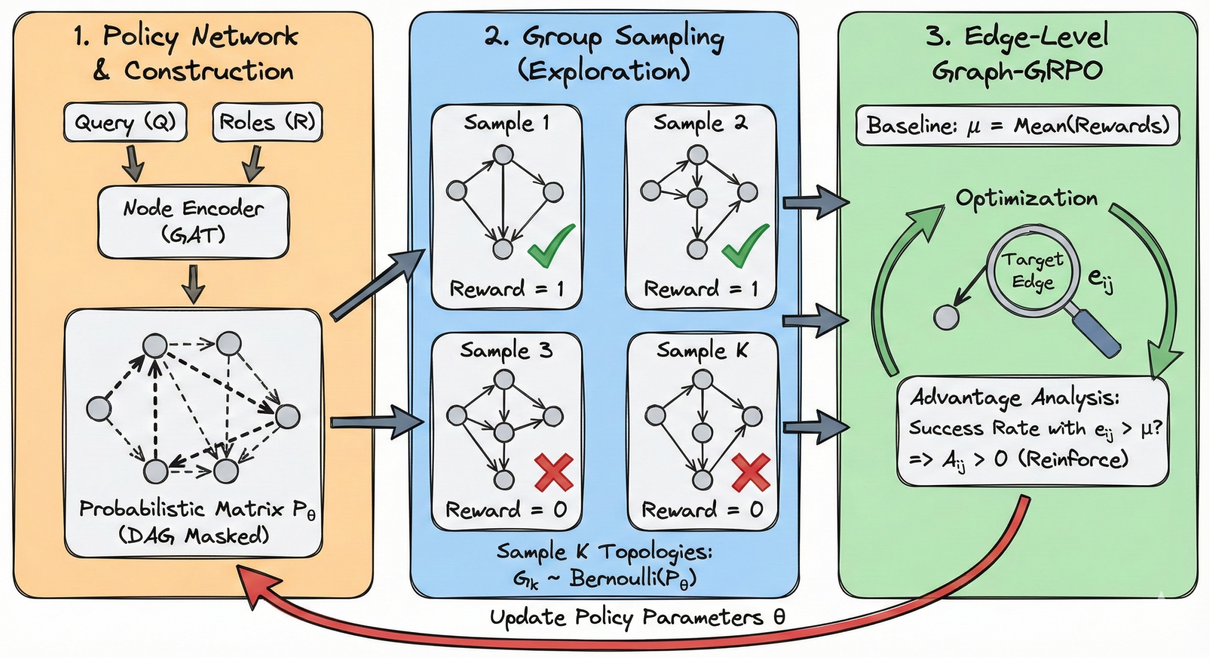

Figure 2: **The overall framework ofGraph-GRPO**. **(1) Policy Network & Construction:** The module encodes agent roles and the task query using a GAT-based encoder to generate a probabilistic connectivity matrix P_θ, constrained by a DAG mask to ensure acyclic flow. **(2) Group Sampling (Exploration):** Instead of a single estimation, we generate a group of K diverse topologies via independent Bernoulli sampling. This exploration captures various structural patterns, where successful topologies receive positive rewards (Reward=1) and failures (e.g., disconnected graphs) receive zero. **(3) Edge-Level Graph-GRPO:** The core optimization phase. We calculate a group baseline μ and estimate the specific advantage of each target edge e_{ij}. Edges that result in a success rate higher than the baseline ($A_{ij} > 0$) are reinforced, iteratively updating the policy parameters θ.

ized the feature vector x_i for each agent. Consistent with G-Designer, this was achieved by concatenating the agent's role description with the query content, followed by the pre-trained MiniLM encoder (Wang et al., 2020):

$$x_i = \text{Encoder}(\text{Role}_i \oplus \mathcal{Q}) \tag{1}$$

where the encoder is fixed to the `all-MiniLM-L6-v2` checkpoint. This shared encoder ensures that agents with similar functional roles (e.g., two different "Coder" agents) perform similar topological behaviors, facilitating generalization.

Topology Generation with DAG Constraint. To capture the potential high-order dependencies between agents, we employed a multi-layer Graph Attention Network (GAT) (Veličković et al., 2018). We used a fully connected graph as the computational substrate for message passing. The GAT module updated agent embeddings by aggregating information from all other nodes, resulting in context-aware embeddings $H \in \mathbb{R}^{N \times D}$.

The probability of a directed connection from agent v_j to v_i was modeled via a bilinear inner product. Crucially, to ensure the reasoning process is acyclic and progressive, we applied a Directed Acyclic Graph (DAG) mask prior to activation. This inductive bias enforced $(P_\theta)_{ij} = 0$ for all $j \leq i$, constraining information to flow strictly from earlier agents to later ones (typically converging towards the final agent v_N). The valid connection probabilities are computed as:

$$(P_\theta)_{ij} = \begin{cases} \sigma(h_i W h_j^T) & \text{if } j < i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $W \in \mathbb{R}^{D \times D}$ is a learnable weight matrix modeling the affinity between roles, and $\sigma(\cdot)$ is the sigmoid function. This continuous probability matrix (P_θ) serves as the basis for both stochastic sampling during training and deterministic thresholding during inference.

3.2 Graph-GRPO Optimization

Standard policy gradient methods, such as REINFORCE (Williams, 1992), assign a uniform reward to all edges in a graph. This creates a coarse-grained feedback loop where redundant edges in a successful graph are falsely reinforced, while

Algorithm 1 Graph-GRPO Training Procedure

```
Require: Training dataset $\mathcal{D}$, Group size $K$, Epochs $T$
Initialize policy network parameters $\theta$
for epoch $= 1$ to $T$ do
    for each batch $(\mathcal{Q}, \text{Roles})$ in $\mathcal{D}$ do
        Compute probability matrix $P_\theta$ via Eq. (2)
        Sample Group: Generate $K$ topologies $\{\mathcal{G}_1, \dots, \mathcal{G}_K\}$ via Bernoulli sampling (Eq. 3)
        Evaluation: Execute each $\mathcal{G}_k$ with LLM agents to get rewards $\{r_1, \dots, r_K\}$
        for each unique edge $(i, j)$ in the group do
            Calculate Conditional Success Rate $S_{ij}$ (Eq. 4)
        end for
        Compute group stats $\mu_S, \sigma_S$ from all $\{S_{ij}\}$
        Compute Advantage $A_{ij}$ (Eq. 5)
        Update $\theta$ by minimizing $\mathcal{L}(\theta)$ (Eq. 6)
    end for
end for
```

critical edges in a failed graph are unfairly penalized. Inspired by Group Relative Policy Optimization (GRPO) (Shao et al., 2024), we propose an Edge-Level Graph-GRPO strategy. Unlike PPO (Schulman et al., 2017), our method does not require a separate Critic network, reducing memory overhead and training instability.

3.2.1 Group Sampling via Monte Carlo Approximation

For each query $\mathcal{Q}$, we approximated the gradient expectation by sampling a group of K distinct topologies $\{\mathcal{G}_1, \dots, \mathcal{G}_K\}$ from the current policy π_θ. To ensure structural diversity and enable the exploration of various reasoning paths, we employed a stochastic sampling strategy. Specifically, the binary existence of an edge in the k-th sampled topology is determined by independent Bernoulli sampling parameterized by the predicted probabilities:

$$\mathbb{I}((i, j) \in \mathcal{G}_k) \sim \text{Bernoulli}((P_\theta)_{ij}) \tag{3}$$

This probabilistic process transforms the continuous probability matrix into discrete graph structures. Crucially, this stochasticity allows the model to explore different connectivity patterns (e.g., sparse chains vs. dense trees) within the same group, constructing a robust local baseline from the group's own statistics for the subsequent advantage estimation.

3.2.2 Marginal Success Rate Estimation

To quantify the contribution of specific connections, we define an edge-specific score S_{ij}. The core intuition is *counterfactual reasoning*: if an edge e_{ij} is truly beneficial, its presence should be positively correlated with task success within the group. We calculate S_{ij} as the conditional success rate:

$$S_{ij} = \frac{\sum_{k=1}^{K} \left(\mathbb{I}((i, j) \in \mathcal{G}_k) \cdot r_k \right)}{\sum_{k=1}^{K} \mathbb{I}((i, j) \in \mathcal{G}_k) + \epsilon} \tag{4}$$

where $r_k \in \{0, 1\}$ is the binary reward of the k-th topology, and ϵ is a small constant for numerical stability. The numerator represents the number of correct trials where edge e_{ij} was active, while the denominator represents the total number of trials containing e_{ij}. Consequently, $S_{ij} \in [0, 1]$ estimates the empirical probability $P(\text{Success}|e_{ij} \in \mathcal{G})$. This mechanism effectively distinguishes critical pathways (high S_{ij}) from noise edges (random $S_{ij} \approx$ Group Average).

3.2.3 Relative Advantage and Objective

To mitigate the variance caused by varying task difficulties (e.g., simple tasks yield high success rates for all edges), we applied the GRPO principle to normalize these scores. The advantage A_{ij} is computed as:

$$A_{ij} = \frac{S_{ij} - \mu_S}{\sigma_S + \epsilon} \tag{5}$$

where μ_S and σ_S are the mean and standard deviation of the scores $\{S_{ij}\}$ computed across *all active edges* in the current group. This normalization ensures that only edges contributing *more than average* to the success rate receive positive reinforcement ($A_{ij} > 0$), while less effective edges are suppressed ($A_{ij} < 0$).

Following the standard formulation of GRPO (Shao et al., 2024), we incorporated a KL-divergence term to constrain the policy update, preventing the model from deviating excessively from the initial distribution. The final loss function is defined as:

$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{E}_{batch}|} \sum_{(i,j)\in\mathcal{E}_{batch}} \Bigg(- A_{ij} \log \pi_\theta(e_{ij}|\mathcal{Q}) + \beta D_{KL}(\pi_\theta || \pi_{ref}) \Bigg) \tag{6}$$

where π_{ref} represents the reference policy (initialized with the supervised fine-tuned parameters and frozen during RL training), and β is the coefficient controlling the KL penalty strength. D_{KL} denotes the Kullback-Leibler divergence between the current policy π_θ and the reference policy π_{ref} for the specific edge distribution. This regularization ensures training stability and prevents reward hacking.

The complete training procedure is summarized in Algorithm 1.

3.3 Inference Mechanism

During the inference phase, we adopt a deterministic strategy to ensure reproducibility and stability. Given a test query $\mathcal{Q}$, we first computed the probability matrix P_θ using the trained policy network. To derive the final discrete topology $\mathcal{G}^*$, we applied a hard thresholding operation:

$$\mathbb{I}((i,j) \in \mathcal{G}^*) = \begin{cases} 1 & \text{if } (P_\theta)_{ij} > \tau \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where τ is a hyperparameter set to 0.5. This mechanism effectively filters out low-confidence connections, resulting in a sparse, task-specific communication structure that minimizes redundancy while preserving critical reasoning pathways.

4 Experiments

4.1 Experimental Setup

Datasets. Following the standard protocol in EIB-LEARNER (Shen et al., 2025), we evaluated our method on six benchmarks across three domains. For general reasoning, we used MMLU (Hendrycks et al., 2021) to assess multi-task knowledge. In the mathematical domain, we employed four widely-used datasets: GSM8K (Cobbe et al., 2021), MultiArith (Roy and Roth, 2015), SVAMP (Patel et al., 2021), and AQUA (Ling et al., 2017). Additionally, we used HumanEval (Chen et al., 2021) to evaluate code generation capabilities.

Baselines. We compared Graph-GRPO against three categories of baselines: (1) Single-Agent Methods, including Chain-of-Thought (CoT) (Wei et al., 2022) and Self-Consistency (SC) (Wang et al., 2023); (2) Fixed Topologies, covering standard structures such as Chain, Tree, Complete Graph, and LLM-Debate (Du et al., 2023); and (3) Topology Optimization Methods, which serve as our primary competitors, including AgentPrune (Zhang et al., 2025a), AgentDropout (Wang et al., 2025), G-Designer (Zhang et al., 2025b), and EIB-LEARNER (Shen et al., 2025).

Implementation Details. We employed GPT-3.5-Turbo as the backbone LLM. The policy network utilized the `all-MiniLM-L6-v2` encoder and a 3-layer GAT, strictly aligned with G-Designer. The agent number N was set to 6 for MMLU, 5 for HumanEval, and 4 for mathematical tasks. During training, we set the group sampling size $K = 16$ and maximize communication rounds to 3. Optimization was performed via Adam with a learning rate of $1e-4$ on NVIDIA A100 GPUs.

4.2 Main Results

Graph-GRPO achieves state-of-the-art performance on all six benchmarks, demonstrating superior adaptability across diverse domains. As presented in Table 1, Graph-GRPO attains the highest average accuracy of **92.45%**, establishing a new benchmark for topology learning.

Comparison with Fixed Structures. Traditional static topologies (Chain, Tree, Complete) struggle to adapt to varying query complexities, capping their average performance at roughly 84%. Notably, while the Complete Graph allows for maximum information flow, it suffers from a lower accuracy (82.16%) compared to simpler structures. This counter-intuitive result highlights the detrimental effect of "information overload" and noise propagation in uncontrolled communication, validating the necessity of topology pruning.

Comparison with SOTA Optimization Methods. Compared to previous dynamic topology methods, Graph-GRPO shows distinct advantages. While EIB-LEARNER represents a strong baseline (91.38%), its reliance on standard policy gradients limits its potential on harder tasks. Graph-GRPO outperforms EIB-LEARNER by a significant margin on complex reasoning benchmarks, such as **+0.9% on GSM8K** and **+2.1% on HumanEval**.

Method	MMLU	GSM8K	AQuA	MultiArith	SVAMP	HumanEval	Avg.
Vanilla	80.39	82.30	71.06	93.09	86.55	71.39	80.80
CoT	81.69	86.50	73.58	93.25	87.36	74.67	82.84
SC (CoT)	83.66	81.60	75.63	94.12	88.59	79.83	83.91
Chain	83.01	88.30	74.05	93.27	87.17	81.37	84.53
Tree	81.04	85.20	71.23	93.68	88.91	80.53	83.43
Complete	82.35	80.10	72.95	94.53	84.01	79.03	82.16
Random	84.31	86.90	76.48	94.08	87.54	82.66	85.33
LLM-Debate	84.96	91.40	77.65	96.36	90.11	84.70	87.53
AgentPrune	85.07	91.10	80.51	94.65	90.58	86.75	88.09
AgentDropout	85.62	91.70	80.94	95.60	91.04	85.98	88.48
G-designer	86.92	93.80	81.60	96.50	93.10	88.33	90.04
EIB-LEARNER	88.90	95.20	83.49	96.83	94.70	89.15	91.38
Graph-GRPO	**90.12**	**96.10**	**84.21**	**97.07**	**96.01**	**91.25**	**92.45**

Table 1: Performance comparison (%) on six benchmarks. The best results are highlighted in **bold**, and the second best are underlined. Baseline results are retrieved from (Shen et al., 2025).

Method	MMLU	GSM8K	HumanEval	Avg.
Graph-GRPO	**90.12**	**96.10**	**91.25**	**92.49**
Graph-Level GRPO	88.54	94.40	89.07	90.67
Δ	-1.58	-1.70	-2.18	-1.82

Table 2: Ablation study on optimization granularity: Edge-Level vs. Graph-Level.

This indicates that as task difficulty increases, the stability provided by our group-relative objective becomes increasingly critical. The overall improvement of **1.07%** over the previous state-of-the-art confirms that our fine-grained credit assignment strategy successfully uncovers more effective reasoning pathways that were previously obscured by optimization noise.

4.3 Ablation Study

To investigate the source of our performance gains, we conducted a rigorous ablation study comparing our Edge-Level Graph-GRPO with a coarse-grained Graph-Level variant.

Graph-Level GRPO. In this variant, we assign the same advantage score to all edges within a sampled topology based on the graph's final result. This simulates a scenario where the "credit assignment problem" is not addressed.

Analysis of Degradation. Table 2 reveals a consistent performance degradation across all tasks when switching to Graph-Level optimization, with an average drop of **1.82%**. The decline is particularly pronounced in HumanEval (-2.18%), a task requiring precise logic chains. This degradation substantiates our hypothesis: Graph-level rewards introduce severe structural noise. In a successful topology, not all edges are beneficial; some may be redundant or irrelevant. By rewarding the entire graph uniformly, the Graph-Level baseline reinforces these "freeloader" edges. Over time, this leads to denser, noisier graphs that hinder reasoning. In contrast, Graph-GRPO's edge-level estimation acts as a soft filter. By aggregating statistics over K samples, it isolates the marginal contribution of each edge, ensuring that only connections causally linked to success are reinforced. This fine-grained granularity is the cornerstone of our framework's robustness.

4.4 Token Efficiency

Beyond accuracy, economic efficiency is paramount for scalable MAS. We analyze the token consumption of Graph-GRPO relative to its performance in Figure 3.

Pareto Superiority. As illustrated in Figure 3, Graph-GRPO occupies the Pareto-optimal frontier (bottom-right corner), offering the best trade-off between cost and accuracy. Traditional methods like LLM-Debate or Complete Graphs incur prohibitive costs (high vertical position) due to quadratic message passing complexity ($O(N^2)$).

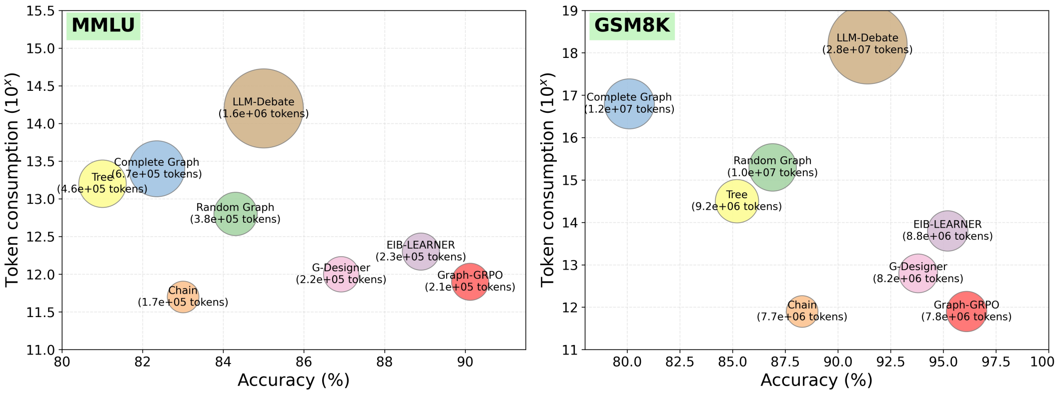

Figure 3: Token efficiency analysis on MMLU and GSM8K benchmarks. The bubble size represents the relative token consumption. **Graph-GRPO (Red)** achieves the highest accuracy (positioned furthest to the right) while maintaining a low token cost comparable to **EIB-LEARNER (Purple)** and **G-Designer (Pink)**. Our method effectively suppresses redundant edges without explicit pruning constraints, achieving a superior performance-efficiency trade-off compared to complete graphs (Blue) and debate-based baselines (Brown).

Crucially, Graph-GRPO achieves a token usage level comparable to explicit pruning methods like AgentPrune, yet delivers significantly higher accuracy. This implies that our method naturally converges to sparse but semantic topologies. By accurately identifying and penalizing non-informative edges during training, Graph-GRPO reduces the "cognitive load" on the system. It demonstrates that the key to efficiency is not merely cutting edges randomly, but preserving the high-value information pathways while eliminating noise, thereby maximizing the "Signal-to-Token Ratio".

5 Conclusion

In this work, we introduce **Graph-GRPO**, a novel framework that stabilizes multi-agent topology learning by fundamentally shifting the optimization paradigm from absolute rewards to group-relative advantage. By implementing a fine-grained edge-level score estimation strategy, our method successfully decouples structural optimization from the noise of task difficulty, effectively resolving the long-standing credit assignment problem in discrete topology search. Extensive evaluations across six reasoning and coding benchmarks demonstrate that Graph-GRPO not only establishes a new state-of-the-art but also naturally converges to sparse, semantic-rich structures, achieving a Pareto-optimal trade-off between decision accuracy and token efficiency. We believe this critic-free, variance-reduced paradigm paves the way for scalable, self-organizing agent swarms, with future work poised to extend this mechanism to larger-scale heterogeneous systems and open-ended, dynamic environments.

6 Limitations

While Graph-GRPO demonstrates strong performance, we acknowledge two main limitations. First, regarding scalability, our policy network relies on a GAT backbone with $\mathcal{O}(N^2)$ complexity. While efficient for typical reasoning groups ($N \leq 6$), applying it to massive swarms (e.g., $N > 100$) may encounter computational bottlenecks, necessitating hierarchical or sparse generation strategies. Second, regarding dynamic adaptability, our framework generates a single static topology for each query. For complex, multi-turn dialogues where optimal communication structures might shift across turns, a finer-grained, turn-level topology adjustment mechanism would be more ideal.

References

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. In *International Conference on Machine Learning (ICML)*, pages 8155–8168.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*.

Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Ceyao Zhang, Ziyang Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, and 1 others. 2024. Metagpt: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations (ICLR)*.

Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. Camel: Communicative agents for "mind" exploration of large language model society. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 51991–52008.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 158–167.

Zeyu Liu, Huimo Yao, Chaowei Zhang, Zihuai Yang, Jiakai Tang, Ye Yuan, Xu Chen, Yankai Lin, and Maosong Sun. 2024. Dynamic llm-agent network: An llm-agent collaboration framework with agent-team optimization. In *International Conference on Learning Representations (ICLR)*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, and 1 others. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 27730–27744.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 2080–2094.

Chen Qian, Wei Liu, Hong Liu, Nuo Chen, Yufan Dang, Guohao Li, Cheng Yang, Weize Chen, Yusheng Su, Zhiyuan Liu, and 1 others. 2024. Chatdev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Chen Qian, Zihao Xie, Yifei Wang, Wei Liu, Yufan Dang, Zhuoyun Du, Weize Chen, Cheng Yang, Zhiyuan Liu, and Maosong Sun. 2025. Scaling large-language-model-based multi-agent collaboration. In *International Conference on Learning Representations (ICLR)*.

Subhro Roy and Dan Roth. 2015. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1743–1752.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Xiao, Y Yang, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300. Origin of Group Relative Policy Optimization (GRPO).

Xu Shen, Yixin Liu, Yiwei Dai, Yili Wang, Rui Miao, Yue Tan, Shirui Pan, and Xin Wang. 2025. Understanding the information propagation effects of communication topologies in llm-based multi-agent systems. *arXiv preprint arXiv:2505.23352*. The EIB-LEARNER paper.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations (ICLR)*.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, and 1 others. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 5776–5788.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*.

Zhexuan Wang, Yutong Wang, Xuebo Liu, Liang Ding, Miao Zhang, Jie Liu, and Min Zhang. 2025. Agentdropout: Dynamic agent elimination for token-efficient and high-performance llm-based multi-agent collaboration. *arXiv preprint arXiv:2503.18891*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 24824–24837.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Peng, Xiubo Wang, and Shaokun Zhang. 2023. Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, and 1 others. 2025. The rise and potential of large language model based agents: A survey. *Science China Information Sciences*, 68(2):121101.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Guibin Zhang, Yanwei Yue, Zhixun Li, Sukwon Yun, Guancheng Wan, Kun Wang, Dawei Cheng, Jeffrey Xu Yu, and Tianlong Chen. 2025a. Cut the crap: An economical communication pipeline for llm-based multi-agent systems. In *International Conference on Learning Representations (ICLR)*. Reference for AgentPrune.

Guibin Zhang, Yanwei Yue, Xiangguo Sun, Guancheng Wan, Miao Yu, Junfeng Fang, Kun Wang, Tianlong Chen, and Dawei Cheng. 2025b. G-designer: Architecting multi-agent communication topologies via graph neural networks. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*.

Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jürgen Schmidhuber. 2024. Gptswarm: Language agents as optimizable graphs. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.