Topology-GRPO: A Hierarchical Reinforcement Learning Optimization Algorithm for Intra-Graph Inter-Group Scenarios

Lanhaijian
Hongqiao University Academic Science and Technology Network

April 22, 2026

Declaration: This paper is a conceptual design based on the framework of *General Topological Agent*. It is for discussion only and has not been fully experimentally verified.

Abstract

The General Topological Agent (GTA) provides a unified paradigm for cross-domain structural intelligence based on topological decomposition and hierarchical reinforcement learning. However, the original framework assumes independent subgraphs with no inter-group coupling, which is inconsistent with real-world scenarios where nodes are grouped within a connected graph while dependencies still exist across groups. To address this core challenge, this paper proposes Topology-GRPO, a novel policy optimization algorithm designed for intra-graph inter-group scenarios. Topology-GRPO enhances state representation via inter-group message passing, introduces topological neighborhood relative advantage estimation, and imposes inter-group policy consistency constraints to ensure global coordination. We elaborate the design details in three typical scenarios: chip design, airline integrated scheduling, and molecular generation. Finally, we discuss open problems and future validation directions.

Keywords: General Topological Agent; Grouped Reinforcement Learning; Graph Neural Network; Structural Intelligence; Hierarchical RL

1 Introduction and Challenges

1.1 Original Decomposition Assumption

The GTA paradigm represents a complex system as a weighted undirected graph $G = (V, E, W)$ and decomposes it into disjoint subgraphs:

$$\bigcup_{i=1}^{k} G_i = G, \quad G_i \cap G_j = \emptyset.$$

Under this assumption, the high-level policy π_h coordinates groups, and each low-level policy $\pi_l^{(i)}$ optimizes locally. The original topological advantage function is:

$$A^g(s, a) = Q^g(s, a) - V^g(s).$$

1.2 Intra-Graph Inter-Group Dilemma

In real-world systems, **inter-group edges** E_{inter} are common and critical:

- **Chip design**: Inter-module nets determine timing closure.

- **Airline scheduling**: Flights connect fleet, crew, gates, and maintenance.
- **Molecular generation**: Scaffold and functional groups are chemically bonded.

Ignoring inter-group coupling causes:

1. Biased advantage estimation.
2. Conflicting local policies.
3. Ambiguous credit assignment.

1.3 Formalization

We upgrade hard decomposition to **soft decomposition**:

$$V = \bigcup_{g=1}^{k} V_g, \quad E = E_{\text{intra}} \cup E_{\text{inter}},$$

where $E_{\text{inter}} = \{e_{ij} \mid v_i \in G_g, v_j \in G_h, g \neq h\}$. The objective becomes:

$$\max_{\pi_h, \pi_l} \mathbb{E}\left[\sum_{g=1}^{k} \sum_{t=1}^{T} \gamma^t R^g(s_t, a_t)\right] \quad \text{s.t. inter-group consistency.}$$

2 Topology-GRPO Design

2.1 Overview

Topology-GRPO extends standard GRPO with three core improvements (Table 1).

Table 1: Standard GRPO vs Topology-GRPO

Component	Standard GRPO	Topology-GRPO
State	Raw state s	Enhanced state $\tilde{s} = [s; m]$
Advantage	Within-group relative	Global-local-topological neighborhood
Constraint	None	Inter-group consistency regularization
Message Passing	None	Sparse GNN-style passing
Structure	Single policy	Hierarchical $\pi_h + \pi_l^{(g)}$

2.2 Inter-Group Message Passing

2.2.1 Message Encoding

For inter-group edge $e_{ij} \in E_{\text{inter}}$:

$$m_{h \to g}^{(i)} = \text{MLP}\left([s_g^{(i)}; \bar{s}_h; W_{gh}]\right),$$

where $\bar{s}_h = \frac{1}{|V_h|} \sum_{j \in V_h} s_h^{(j)}$, W_{gh} is coupling strength.

2.2.2 Aggregation

- Chip design: MAXPOOL (timing criticality)
- Airline scheduling: SUM (resource accumulation)
- Molecule generation: Attention (feature fusion)

2.2.3 Gated Fusion

$$\tilde{s}_g = \text{CONCAT}\left(s_g, \sigma(\text{MLP}([s_g; m_g])) \odot m_g\right).$$

2.3 Topological Neighborhood Advantage

The advantage is decomposed into three parts:

$$\hat{A}_i^g = \alpha \cdot \frac{R_i^{\text{global}} - V^{\text{global}}}{\sigma^{\text{global}}} + (1-\alpha) \cdot \frac{r_i^g - V^g(s_g)}{\sigma^g} + \beta \cdot \frac{r_i^g - \mu^{\mathcal{N}(g)}}{\sigma^{\mathcal{N}(g)}}.$$

The topological neighborhood:

$$\mathcal{N}(g) = \{h \mid \exists e_{ij} \in E_{\text{inter}}, v_i \in G_g, v_j \in G_h\}.$$

2.4 Inter-Group Consistency Loss

$$\mathcal{L}_{\text{cons}} = \sum_{(g,h) \in E_{\text{inter}}^{\text{group}}} W_{gh} \cdot \left\| \mathbb{E}_{s_g} \pi_l^{(g)} - \mathbb{E}_{s_h} \pi_l^{(h)} \right\|^2.$$

2.5 Full Objective

$$\begin{aligned}\mathcal{L}^{\text{Topo-GRPO}} = &\sum_{g=1}^{k} \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G} \min\left(r_i^g(\theta)\hat{A}_i^g, \text{clip}(\cdot)\hat{A}_i^g\right)\right] \\ &- \lambda_{\text{cons}}\mathcal{L}_{\text{cons}} - \beta D_{\text{KL}}(\pi_\theta \parallel \pi_{\text{ref}}).\end{aligned}$$

2.6 Full Algorithm of Topology-GRPO

Algorithm 1 Topology-GRPO Training Procedure

Require: Topological graph $G = (V, E, W)$, group partition $\{V_g\}_{g=1}^k$, hyperparameters $\gamma, \lambda_{\text{cons}}, \beta, \epsilon$

Ensure: High-level policy π_h, low-level policies $\{\pi_l^{(g)}\}$

```
Initialize π_h, {π_l^(g)}, message encoder, and optimizers
while not converged do
    Construct intra-group edges E_intra and inter-group edges E_inter
    Collect trajectories in parallel across groups
    for each group g do
        Perform inter-group message passing to get enhanced state s̃_g
        Sample action a_g ~ π_l^(g)(·|s̃_g)
    end for
    Execute joint actions and obtain global & group-wise rewards
    Compute topological neighborhood advantage Â_i^g for all groups
    Calculate policy loss, consistency loss, and KL divergence
    Update π_h and {π_l^(g)} by minimizing L^Topo-GRPO
    Decay λ_cons gradually (curriculum learning)
end whilereturn π_h, {π_l^(g)}
```

3 Domain-Specific Designs

3.1 Chip Design

Nodes: standard cells, macros, I/O pins. Edges: intra-module and inter-module nets. Weights: 1/slack (timing criticality). Message: slack, arrival time, required time, critical flag.

3.2 Airline Integrated Scheduling

Bipartite graph: flights and resources (fleet, crew, gate, maintenance). Message: resource availability, fatigue risk, delay probability. Consistency: time-window alignment for the same flight.

3.3 Molecular Generation

Groups: scaffold, functional groups, pharmacophores, bonds. Message: valency, chemical validity, synthetic accessibility (SA). Consistency: bond-type matching and conformation coherence.

4 Implementation and Stability

Key techniques:

- Topology-aware gradient clipping
- Running mean/std normalization for value targets
- Entropy regularization to prevent policy collapse
- Delayed inter-group message synchronization
- Curriculum learning for consistency strength

5 Open Problems

1. Convergence guarantees under inter-group coupling
2. Optimal dynamic neighbor range scheduling
3. Distributed implementation for large-scale systems
4. Online adaptation to dynamically evolving topologies
5. Integration with large language model agents

6 Conclusion

Topology-GRPO unifies grouped reinforcement learning, graph message passing, and hierarchical policy optimization to solve intra-graph inter-group structural decision problems. It extends the General Topological Agent paradigm to real-world coupled systems while maintaining low computational complexity and high scalability.

References

[1] General Topological Agent: A Unified Paradigm for Cross-Domain Structural Intelligence, 2026.

[2] Shao Z, et al. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300, 2024.

[3] Mirhoseini A, et al. A graph placement methodology for fast chip design. Nature, 594(7862):207–212, 2021.

[4] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. ICLR, 2017.

[5] Veličković P, et al. Graph attention networks. ICLR, 2018.

[6] Schulman J, et al. Proximal policy optimization algorithms. arXiv:1707.06347, 2017.