# AgentPO: Enhancing Multi-Agent Collaboration via Reinforcement Learning

**Lin Sun[1,2], Chuang Liu[2] , Can Zhang[2] , Yubin Wu[2] , Weijia Lu[2] , Ning Wu[2]**

[1]MatrixRobotics MARI, [2]UAES AI Lab

## Problem Definition and Contribution

**Problem Definition:**

Multi-Agent Systems (MAS) struggle with optimizing agent interactions.
Manual workflows are unstable, and automated topology search is unscalable.

**Key Question:**

How to train agents to cooperate effectively within a fixed topology?

**Key Contribution:**

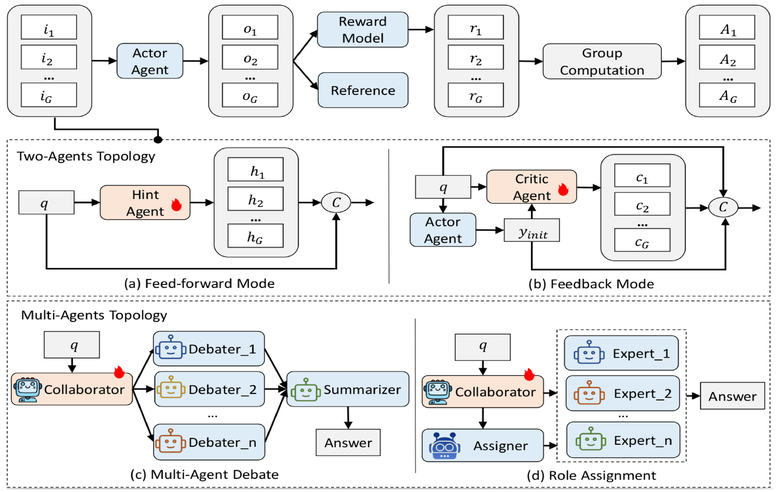
We propose AgentPO, a framework that optimizes collaborative dynamics, not agent architecture.

- **Efficient Optimization:**
- A lightweight Collaborator is trained via RL to guide a fixed, powerful Actor.
- **Strong Performance:**
- Outperforms strong baselines by up to +11.3% on reasoning tasks.
- **High Efficiency:**
- Achieves gains with only 500 training samples and 7.8% inference cost of prior methods.

## Method

**Process:**

- **Define Topology:**
- Choose a fixed interaction structure (e.g., Hint-Actor or Critic-Actor) that specifies how the Collaborator and Actor exchange information.
- **Generate Collaborative Signal:**
- The Collaborator (trained, lightweight) takes the problem context and produces an auxiliary signal (hint or critique) based on the topology.
- **Produce Final Answer:**
- The Actor (fixed, powerful) receives the original problem and the Collaborator's signal, then generates the final answer.
- **Evaluate & Reward:**
- The answer is compared to the ground truth using a binary reward (success/fail).
- **Optimize Collaborator:**
- Using Group Relative Policy Optimization (GRPO), the Collaborator's policy is updated to maximize the expected reward, learning to generate effective guidance.




## Experiments & Result

**Comparison with Baseline Methods.**

(1) AgentPO consistently outperforms all baseline algorithms.
(2) AgentPO scales effectively with the capability of the Actor model.

| Algorithm | AIME24 | Math500 | OlympiadBench | Minerva | AMC23 | Avg. | Δ |
|---|---|---|---|---|---|---|---|
| **Actor Model: Llama-3.2-3B-Instruct** | | | | | | | |
| CoT | 6.7 | 48.4 | 16.4 | 21.0 | 15.0 | 21.5 | +3.0 |
| Self-Consistency | 3.3 | 49.6 | 15.9 | 16.5 | **27.5** | 22.6 | +1.9 |
| Self-Refine | **10.0** | 33.6 | 12.7 | 15.1 | 12.5 | 16.8 | +7.7 |
| Multi-Agent Debate | 3.3 | 47.2 | 15.7 | 20.2 | 25.0 | 22.3 | +2.2 |
| Step-back Abstraction | 10.0 | 43.0 | 13.9 | 14.0 | 20.0 | 20.2 | +4.3 |
| Quality-Diversity | 3.3 | 40.0 | 14.2 | 15.1 | 17.5 | 18.0 | +6.5 |
| Role Assignment | 10.0 | 49.4 | **16.6** | 17.6 | 20.0 | 22.7 | +1.8 |
| EvoAgent | 10.0 | 33.4 | 12.6 | 13.2 | 17.5 | 17.3 | +7.2 |
| AgentPO (ours) | 6.7 | **52.2** | 16.1 | **22.4** | 25.0 | **24.5** | |
| **Actor Model: Llama-3.1-8B-Instruct** | | | | | | | |
| CoT | 3.3 | 43.0 | 15.0 | 18.4 | 17.5 | 19.4 | +12.1 |
| Self-Consistency | 3.3 | 50.6 | 17.2 | 19.5 | 25.0 | 23.1 | +8.4 |
| Self-Refine | 0.0 | 43.2 | 15.0 | 18.0 | 30.0 | 21.2 | +10.3 |
| Multi-Agent Debate | 6.7 | 50.2 | 15.6 | 24.3 | 25.0 | 24.4 | +7.1 |
| Step-back Abstraction | 3.3 | 44.2 | 14.4 | 18.8 | 20.0 | 20.1 | +11.4 |
| Quality-Diversity | 3.3 | 45.2 | 16.3 | 19.5 | 17.5 | 20.4 | +11.1 |
| Role Assignment | 6.7 | 47.4 | 15.9 | 24.3 | **35.0** | 25.9 | +5.6 |
| EvoAgent | 3.3 | 36.0 | 12.7 | 16.5 | 32.5 | 20.2 | +11.3 |
| AgentPO (ours) | **16.7** | **60.6** | **28.9** | **26.1** | 25.0 | **31.5** | |

**Comparsion between Collaborative and Actor Optimization.**

Collaborative optimization outperforms direct Actor fine-tuning.

| Algorithm | AIME24 | Math500 | OlympiadBench | Minerva | AMC23 | Avg. |
|---|---|---|---|---|---|---|
| Qwen2.5-Math-7B | 0.2 | 69.0 | 34.7 | 21.3 | 45.8 | 38.2 |
| **Actor Optimization** | | | | | | |
| SimpleRL-Zero-7B | 26.7 | 78.2 | 40.3 | 27.6 | 60.2 | 46.6 |
| Prime-Zero-7B | 16.7 | 83.8 | 40.9 | 36.0 | 62.7 | 48.0 |
| OpenReasoner-Zero-7B | 13.3 | 79.2 | 44.0 | 31.6 | 47.0 | 43.0 |
| **Collaborative Optimization** | | | | | | |
| AgentPO (ours) | 13.3 | 83.8 | 39.1 | 43.4 | 67.5 | **49.4** |

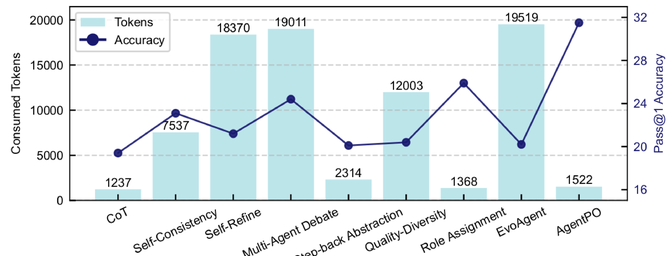
**Performance in Complex Multi-Agent Topologies.**

AgentPO consistently improves multi-agent systems.

| Algorithm | AIME24 | Math500 | OlympiadBench |
|---|---|---|---|
| Debate | 8.9 ± 3.8 | 46.3 ± 1.0 | 15.3 ± 1.2 |
| w/ AgentPO | **12.3 ± 2.6** | **49.6 ± 0.8** | **21.1 ± 1.4** |
| Δ | +3.4 | +3.3 | +5.8 |
| Role Assignment | 7.3 ± 2.9 | 48.3 ± 0.8 | 15.7 ± 0.6 |
| w/ AgentPO | **10.2 ± 1.7** | **52.2 ± 0.6** | **20.9 ± 0.8** |
| Δ | +2.9 | +3.9 | +5.2 |

**Inference Efficiency and Practicality.**

AgentPO achieves an optimal accuracy–cost trade-off, delivering a scalable and cost-effective solution for advanced reasoning in practice.




**Conclusion:**

(1) Optimizing how agents interact is more effective and scalable than searching for how to structure them.
(2) Collaborative optimization achieves better performance with far less training data than direct actor fine-tuning.
(3) AgentPO enables building hybrid multi-agent systems that combine powerful black-box API models with lightweight, locally trainable agents.

Project Code