Poisoning Attacks on Multi-Agent Reinforcement Learning Systems

Chanyeok Choi[1], Jaehwan Cho[1], Youngmoon Lee[1]

***Abstract*— Humanoid robots are increasingly relying on reinforcement learning by building reward models aligned to humans, such as training language models to follow human instructions. However, multi-agent reinforcement learning systems such as robot teaming suffer large performance loss due to reward model anomaly, and their low observability makes anomaly detection challenging. This paper investigates the impact of poisoning attacks that exploit shared reward structures in multi-agent reinforcement learning, luring agents into reward traps. Specifically, we present a poisoning attack tailored for deep reinforcement learning in multi-agent setup, and evaluate its vulnerability on two representative reinforcement learning algorithms: PPO and SAC. Results demonstrate performance degradation of 18.7% (PPO) and 20.9% (SAC). While SAC showed a marginal decline in performance compared to PPO, PPO experienced a severe reward collapse under attack. This suggests that PPO is vulnerable to poisoning attacks, especially in multi-agent environments. These findings call for robust defense mechanisms against reward-based attacks in multi-agent reinforcement learning systems. Our experiment is available at: `https://github.com/RAISELab/MAPA`.**

I. Introduction

Humanoid robots and intelligent agents are increasingly trained by reinforcement learning from human feedback, a paradigm that builds reward models aligned with human preferences. This approach has gained traction in tasks such as training large language models to follow natural language instructions [1], where explicitly designing a reward function is difficult or infeasible. In such settings, human feedback serves as a proxy for the reward signal, enabling agents to learn goal-aligned, often crowd-sourced behavior without relying on manually labeled datasets. However, reinforcement learning systems, in return, may become vulnerable to reward model anomalies from crowdsourcing.

While existing reward anomalies and attacks focused on single-agent reinforcement learning [3], multi-agent reinforcement learning (MARL) has emerged as a powerful framework for coordinating multiple agents at scale [4] such as robot teaming. When combined with reinforcement learning from human feedback, multi-agent systems are particularly well suited for human and multi-agent interaction scenarios, where multiple agents collaborate with humans in complex, dynamic environments. However, MARL is vulnerable to reward model anomalies and training-time perturbations due to limited observability.

Distributed nature of MARL makes detecting anomalies or behavioral drift challenging, posing risks to trust and safety in MARL systems. In particular, reward poisoning attacks manipulate the reward function during training, leading agents to learn suboptimal policies. Techniques such as reward traps, which lure agents with artificially high rewards along deceptive paths, have been explored in single-agent settings. However, their impact in multi-agent environments remains underexplored.

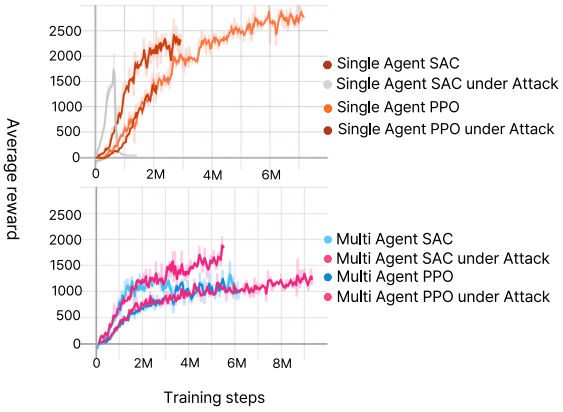

Fig. 1: Reward variations of PPO and SAC in single-agent and multi-agent environments under baseline and attack conditions, demonstrating attack conditions cause greater reward fluctuations and performance degradation, especially in multi-agent settings.

To address this gap, we propose a novel reward poisoning attack tailored for multi-agent robot interaction environments. Our attack targets the shared reward structures commonly found in MARL systems by inserting reward traps that exploit inter-agent coordination dynamics, leading to synchronized policy deviations. We evaluate its effectiveness using two widely adopted deep reinforcement learning algorithms under different configurations: PPO vs. SAC, multi-agent vs. single-agent, with vs. without attack.

Our results, visualized in Fig. 1 and detailed in Table 1, highlight that scenarios involving attacks consistently lead to performance degradation. In particular, PPO-based agents show a greater reduction in cumulative rewards compared to SAC-based agents, indicating that PPO is more susceptible to reward manipulation, an effect that is especially pronounced in multi-agent settings.

II. Poisoning Attack on Multi-Agent Learning

This study investigates policy and reward poisoning in environments with limited data, and introduces an attacker that strategically disrupts multi-agent learning dynamics. Fig. 2 illustrates the experimental environment designed to enable interaction between a crawler agent and an attacker agent. Within a 50x50 meter environment, the crawler's goal is

*This work was supported by Institute of Information and Communications Technology Planning and Evaluation (IITP) grant IITP-2025-RS-2020-II201741, RS-2022-00155885, RS-2024-00423071 funded by the Korea government (MSIT).

[1]All authors are with Department of Robotics, Hanyang University, Ansan, Republic of Korea {angledsugar, yh26175966, youngmoonlee}@hanyang. ac.kr

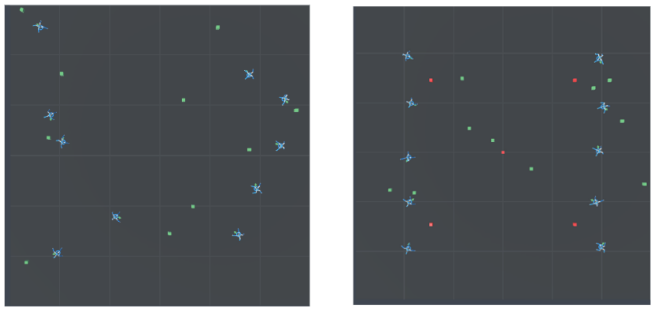

Fig. 2: Poisoning attack setup. The crawler agents move to the green target points to maximize rewards, while the attacker agents place high reward red lure points to mislead it. Left: baseline environment. Right: manipulated environment with attacker agents place lures disrupting the crawler's path.

to reach a green target point and maximize its cumulative reward. Meanwhile, the attacker interferes with the crawler's objective by generating lure points marked as red color at arbitrary locations. The reward rules are predefined during the environment design phase and are not altered for individual agents. Specifically, when the crawler reaches a lure point, it receives a reward of 100 points, while the attacker receives 1 point. Additionally, if the crawler removes a lure point, it is penalized with a reward of -1 point. In this setup, the roles, rewards, and objectives of both the crawler and the attacker are fixed within the environment, and each agent learned within this experimental setting.

One of the core components is reward manipulation, a strategy that targets agents in multi-agent reinforcement learning environments where behavior heavily depends on reward feedback by injecting premature rewards and subtly altering reward timing to induce suboptimal behavior while evading detection by conventional monitoring systems. Another core component is random behavior, which introduces environmental uncertainty by allowing the attacker to act unpredictably or disrupt key task elements. For example, the attacker may relocate the agent's goal object to random positions or move it to an unreachable height. This unpredictability hinders the agent's ability to learn stable policies and exposes its vulnerability under non-deterministic conditions.

A tempting reward attack is an adversarial strategy based on reward addiction, designed to mislead agents by inserting artificially high-reward locations away from their optimal paths. Instead of following the intended trajectory toward the goal, agents are lured by inflated rewards placed earlier in the environment, leading them to learn suboptimal behaviors. In multi-agent settings, this effect is amplified as multiple agents converge on these misleading reward spots, disrupting cooperation and goal achievement. To maximize the impact, high-reward points are randomly placed, increasing distraction and destabilizing policy learning. As a result, the attack significantly reduces learning efficiency and overall performance. This highlights the urgent need for robust detection and defense mechanisms against such sophisticated threats in multi-agent reinforcement learning.

III. Evaluation

This experiment successfully demonstrates the effectiveness of the attack. We evaluated it across various multi-agent and single-agent reinforcement learning environments using PPO and SAC algorithms. Table 1 presents the cumulative rewards of crawler and attacker agents under different scenarios.

In the multi-agent PPO, the crawler achieved 528.4, but under attack, it dropped to 429.4, with the attacker recording -2.903. In the multi-agent SAC setting, the reward decreased from 971.3 to 769.9, and the attacker received -2.449. In single-agent scenarios, the impact was more severe. The PPO agent's reward dropped from 647.5 to 302.5, and the attacker received 1. SAC showed the largest drop: from 1,276 to 23.93, with the attacker at -31.43.

These results confirm that our poisoning attack effectively disrupts agent learning, with PPO being more vulnerable than SAC, and multi-agent systems, while slightly more robust, still affected. This underscores the serious threat such attacks pose to real-world multi-agent systems like robot teaming.

TABLE I: Cumulative reward with and without attack.

Scenario	Agent	Step	Reward
Multi-Agent PPO	Crawler	1M	528.4
Multi-Agent PPO under Attack	Crawler	1M	**429.4**
Multi-Agent PPO under Attack	Attacker	1M	-2.903
Multi-Agent SAC	Crawler	1M	971.3
Multi-Agent SAC under Attack	Crawler	1M	**769.9**
Multi-Agent SAC under Attack	Attacker	1M	-2.449
Single-Agent PPO	Crawler	1M	647.5
Single-Agent PPO under Attack	Crawler	1M	**302.5**
Single-Agent PPO under Attack	Attacker	1M	1
Single-Agent SAC	Crawler	1M	1276
Single-Agent SAC under Attack	Crawler	1M	**23.93**
Single-Agent SAC under Attack	Attacker	1M	-31.43

IV. Conclusions

Multi-agent reinforcement learning systems are at the core of human-interactive robotics, but they are highly vulnerable to reward poisoning attacks during training. This risk is particularly severe in environments with strong inter-agent dependencies, where manipulated rewards can compromise the entire system. To mitigate this threat, it is essential to design not only robust reward functions but also early detection and defense mechanisms. Future research must prioritize not just performance, but also security and trustworthiness.

References

[1] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," in *NeurIPS*, 2022.

[2] J. Sorg, R. L. Lewis, and S. Singh, "Reward design via online gradient ascent," in *NeurIPS*, 2010.

[3] Y.-C. Lin, Z.-W. Hong, Y.-H. Liao, M.-L. Shih, M.-Y. Liu, and M. Sun, "Tactics of adversarial attack on deep reinforcement learning agents," in IJCAI, 2017.

[4] W. Du and S. Ding, "A survey on multi-agent deep reinforcement learning: from the perspective of challenges and applications," Artificial Intelligence Review, vol. 54, pp. 3215–3238, 2021.

[5] A. Rakhsha, X. Zhang, X. Zhu, and A. Singla, "Reward poisoning in reinforcement learning: Attacks against unknown learners in unknown environments," in *NeurIPS Workshop on Strategic Feedback (StratML)*, 2021.

[6] X. Zhang, Y. Ma, A. Singla, and X. Zhu, "Adaptive reward-poisoning attacks against reinforcement learning," in *ICML*, 2020.