Proceedings of CONF-SEML 2026 Symposium: Computational Analysis and Modeling in Complex Intelligent Systems
DOI: 10.54254/2755-2721/2026.GU33941

Task Complexity Determines the Effectiveness of PPO-Based Improvements in Reinforcement Learning: An Empirical and Analytical Study

Ruoyu Wu

School of Computer Science, University of Birmingham, Birmingham, UK
rxw553@student.bham.ac.uk

Abstract. Proximal Policy Optimization (PPO) is one of the most widely adopted reinforcement learning algorithms in both academic research and industrial applications. Its widespread popularity mainly comes from its outstanding training stability and simple implementation, which helps it avoid the common flaws of earlier policy gradient methods: unstable convergence and excessive sensitivity to hyperparameters. In recent years, the research community has put forward a large number of modified variants and targeted improvements to enhance PPO's learning efficiency, asymptotic performance and generalizability across different task environments. However, there is still a lack of systematic research on how the inherent complexity of a target task affects the actual effectiveness of these PPO improvements. In this paper, we try to fill this research gap by empirically explore how the task complexity would affect the effectiveness of PPO's improvements. This paper first conducted controlled evaluations of mainstream PPO variants in standard OpenAI Gym single-agent environments of explicitly graded difficulty, including low-complexity CartPole and medium-complexity LunarLander. The empirical results show that as task complexity increases, the performance reliability of PPO methods decrease significantly, and the performance gaps between different variants also widen noticeably. This paper further extend the analysis to multi-agent settings, which add extra challenges such as environmental non-stationarity and inter-agent coordination. Finally, this paper conclude that the effectiveness of PPO-based methods strongly depends on task complexity, which highlights that training stability and cross-environment adaptability are critical for developing PPO algorithms for complex and multi-agent scenarios.

Keywords: Reinforcement Learning, Proximal Policy Optimization, Task Complexity, Multi-Agent Reinforcement Learning, Generalization

1. Introduction

In recent years, reinforcement learning (RL) has achieved a lot of success specially in the problems of control and decision making. Proximal Policy Optimization (PPO) is one of the most used algorithms among the many that have been developed. Its simplicity and strong empirical performance made it one of the most used methods. PPO is a method that limits the policy updates

DOI: 10.54254/2755-2721/2026.GU33941

in order to improve the training stability by using a clipped objective function [1]. Most of the effectiveness of PPO is from the fact that it is able to keep control the exploration and exploitation and has a stable learning dynamics. Compared with the earlier policy gradient methods, PPO reduce the risk of having large and unstable update that can negatively affect the training performance. PPO is well suited for a wide range of applications like for example robotics control, game playing, and autonomous systems.

Although much progress has been made, Recent research has also proposed some improvements to make PPO better in performance as follows. For instance, dynamic entropy adjusting and smooth clipping mechanism improve exploration and stability [2], and adaptive penalty mechanism can give better control of policy update [3]. Besides that, the research also tried to apply PPO in multi-agent reinforcement learning setting in order to improve the coordination among agents [4]. These improvements are trying to solve the main shortcoming of the original PPO algorithm, such as not enough exploration in complicated environment, and not so stable in high dimension state space. As the reinforcement learning task will be more difficult in the coming time, especially in the case of more agents or the environment is dynamic, it will be important to have more robust and more adaptable PPO based method.

2. Introduction PPO and its variants

Besides the above basic improvements, there are some more advanced approaches. CMA-MAPPO combines the evolutionary strategies with PPO to improve the exploration in the sparse-reward environment [4]. By combining the idea of reinforcement learning with evolutionary optimization, this approach can improve the diversity of the policy updates and help to avoid the premature convergence, which is a common problem when the environment is complex. So we know that this approach would be very effective when we are in a situation, in which the exploration strategy of each one may not find the best solution in the sparse-reward environment. Besides, the survey study also shows the coordination and scalability of the multi-agent reinforcement learning system [5]. We know that when the number of agents is increasing, the number of the interactions grows too. Therefore, in this case, we need an algorithm which can deal with the communication, the cooperation, and the decentralized decision making. So the scalability and the coordination have become the important design of the modern multi-agent reinforcement learning method.

Other improvements are Truly PPO, which improves the objective function to get better convergence properties [6]. The idea of Truly PPO is to put more accurate constraints on how the policy is updated to make the training less unstable, and also the performance is more similar in different environments. This is also important when in the task the change of policy little, but the result may change little too. Augmented PPO methods improve more safety and robustness in training [7]. These methods usually adopt more constraints or regularization methods to make the training to not do unsafe or bad behaviors, especially in dangerous or in real world. By improving both the stability and the reliability, the augmented PPO methods make reinforcement learning more suitable for more complex and safety more important applications.

3. Effect of task complexity

The complexity of the task is very important for the performance of reinforcement learning. PPO based method can get the strong result for the cooperative multi-agent environment in many studies. But the more complex the environment is, the harder it is to keep the learning in a stable. One of the most difficult things is generalization. It has been found that the policy trained in the one

DOI: 10.54254/2755-2721/2026.GU33941

environment may not work well in the different environment [8]. In the theoretical work, it points out that the more complex the task is, the higher is the sample complexity, which directly affects the learning efficiency [9].

3.1. Performance on CartPole

CartPole is a classic control problem. The state space is low-dimensional and the reward signal is dense. Because the structure of the task is relatively simple, the agent can learn the relationship between its actions and the environment's response quickly. In this environment, PPO usually can get fast convergence and stable performance. Also, the state representation is very simple, the agent can explore the environment without too much uncertainty. In addition, the dynamics of system is also relatively simple, so the learning is stable. In this way, PPO can use gradient based update way to optimize its policy, and it does not need to consider the complex stabilization way.

The dense reward signal also makes learning easier, because during the training the agent will hear from its experience very often and very informative. This makes the credit assignment easier, because we know which actions give better. Thus, PPO will update the policy easily without using some fancy exploration strategy or any other sort of regularization. Also, because the complexity of the environment is very low, the variance of the policy is also small. Thus, the training is also very stable. We think that CartPole is a good benchmark to see the baseline performance, not to test some advanced algorithm improving.

Thus, in simple environments such as CartPole, extra PPO improvements might not be that helpful. In most cases, the plain PPO algorithm is already good enough to get close to the optimal performance, and extra improvements might just give us nothing.

3.2. Performance on LunarLander

Compared with the simple environments, the LunarLander is more complex than CartPole because it involves a higher dimensional state space, sparse rewards, and more unstable dynamics of . LunarLander is more complex than CartPole because the agent need to control the landing position, velocity, angle and fuel usage at the same time. The policy learning is more difficult because compared with simple environment, the increase of the dimensionality of state space introduces more uncertainty, the agent has more difficulty to explore all the states. In addition, the dynamics of lunar lander are more sensitive to small change of actions, this can make the thing have a large variation, this make the learning process more unstable and the difficulty of finding the optimal policy is bigger.

PPO in this environment also tends to be slower to converge, and the performance is also less stable. The reason is that in this environment, the rewards are not as common as in CartPole, probably the agent need more training experience before he can learn a good policy. In this sparse reward structure, it is more hard for the agent to know that which action would lead to a long time later. So the credit assign is more difficult as well. In addition, there is delayed reward and there is the control objective, the agent needs to consider many things at the time when he makes the decision, so the policy update have high variance. It often leads that the agent oscillates in the training or just converges to the bad policy.

In other words, the improvements related to exploration and stability are more useful when the task is more complex. For example, in such environment, techniques like better exploration strategy, better clipping, and better policy update can help to improve the learning performance. Hence, the

DOI: 10.54254/2755-2721/2026.GU33941

LunarLander is representative of the way that when we increase the task complexity, the standard PPO has a limit and we can see the usefulness of algorithmic improvement.

4. Generalization and multi-agent extension

Generalization is an important issue in reinforcement learning. For example, when we move from a simple single agent environment to more complicated and dynamical one. A method that performs well in a simple benchmark, may not perform the same in more difficult. In complex task, the feedback is often sparse, the state is much larger and the transition is more uncertain. In addition, the PPO based methods can be also extended to multi agent environments, and the extension works very well with the strong scalability and coordination performance [10]. The multi agent environments is a little bit difficult because that agents should learn in the time that other agents are also changing the behaviours. The environment is not stationary and it makes the stable learning more difficult.

4.1. Muti-agent reinforcement learning

In multi-agent reinforcement learning, each agent needs to take into account not only its own actions, but also the actions of the other agents. Therefore, the learning is much more complicated than in single agent environment. Because the state of the environment is affected by the joint action of the multi agents, the optimal policy of one agent may depend on the evolution of the other's strategies. So the coordination and the cooperation of the agents are parts of the learning of the multi-agent system. In addition, the multi-agent system is intrinsically non-stationary. In single agent reinforcement learning, the environment is usually fixed. In multi-agent, each agent is keep changing the policy, that is to say, the environment is dynamic for any agents. The non-stationary will bring a lot of difficulty for the traditional reinforcement learning algorithms, which may make the training is not stable, to oscillate or even diverge.

These interactions can also further destabilize the training, if the agents learn at different rates or have different goals. For instance, in an environment involving both cooperation and competition, there may be conflicts between the strategies of the agents, that in turn deteriorate the performance of the whole system. In addition, some of the agents may prematurely converge to a suboptimal policies, and in this way limit the whole system. Thus, the algorithms for the multi-agent environment should be stronger in stability and ability to adapt to changing of the interactions and the complex dependencies.

In the recent works to tackle these problems, they tried to work on the way of policy update, put forward the idea of the framework of centralized training with decentralized execution and try to design better way of communication and coordination for the agents. These works try to reduce the effect of non-stationarity and make the result more robust and general in the multi-agent complex environment.

4.2. Hungry Birds environment

The Hungry Birds environment can be viewed as a multi-agent environment with cooperation, competition, and resource management. Compared with the standard single agent task, the agents in this environment are in harder situations, that is, they have less resources and their decisions are affected by the dynamic of the other agents. For example, the agent may need to compete the resource which is very limited, but at the same time has to adapt the behavior of other agents, because the strategy of other agents are changing all the time. It is much harder for the learning. In

DOI: 10.54254/2755-2721/2026.GU33941

contrast to static environment, the state transitions of Hungry Birds are affected by the joint actions of many agents, which is a very dynamical and non-stationary environment. The environment itself changes when the agents learn and update the policies of themselves over time. It is harder for any of agent to converge to a stable and good strategy. Besides, there are both parts of cooperation and competition, that is, the agents need to trade off the gain of short time and the coordination in long time.

In such settings, PPO based methods have to face several challenges. First, the policy updates should be stable as the environment is continuously changing. Second, it should support coordination of the agents, in particular when it is better for the overall result if the agents cooperate with each other. Third, they should have strong generalization ability because the agents have to face many different situations and interactions which are not necessarily seen during the training. The complexity of the environment of Hungry Birds also shows that standard PPO is not very suitable for multi-agent. If we do not modify it in some way, e.g. with better exploration or with more robust policy update, PPO does not have to give a good and consistent performance. Therefore, Hungry Birds is a good benchmark in which to see how good the improvements of PPO is with the real and complex conditions.

Hence, Hungry Birds is a representative example for elucidating why the task complexity is important in the PPO based reinforcement learning. From the performance in such environments, it can be seen what are the strengths and drawbacks of different algorithmic ways, and what is the direction of further improvements in terms of stability, adaptability and scalability.

5. Discussion

The effectiveness of the PPO based improvements strongly depends on the complexity of the task. Generalization is still a big problem, since policies obtained in one environment do not generalize to others [11].

In simple environments, such as CartPole, PPO is usually able to learn stable and effective policies with relatively little difficulty. The environment dynamics are simple, and the reward signals are dense, and the agent is able to quickly associate actions with what is the result. In such situations, many of the PPO improvements provide only little. The same is true for simple environments, such as the one in CartPole. In simple environments, the environment dynamics are simple, and the reward signals are dense, and the agent is able to quickly associate actions with what is the result. As task becomes more complex, the same is true for the environment. In an environment, like LunarLander, or in the case of multi agent, the agent has to deal with sparse rewards, with bigger state space, and with more uncertain dynamics. All of these make the learning more difficult and often induce instabilities during training. In such cases the improvements to PPO, in particular the ones that are about exploration and stability have a more important role.

In multi agent environments also the problem is more difficult because of the non stationarity: the behaviour of each agent affects the environment that affects the other agents. It is harder for ppo to reach a policy where is not converging. As consequence is important to improve the generalization and the adaptability for the application of the methods based on ppo to complex and real problems.

6. Conclusion

In this paper, we have analyzed how the complexity of the task can impact the improvements of PPO based. For the future work, it is important to improve the stability, adaptability, and sample efficiency. In general, the results show that PPO does well in the simple environment, but when the

DOI: 10.54254/2755-2721/2026.GU33941

task is more complicated, the performance of PPO is bad. The improvements of PPO can help with these problems, but it depends on the characteristic of the environment. For example, the environment with sparse rewards, high dimensional state space, or the environment with many interacting agents should use the more advanced strategy.

Furthermore, the extension of PPO to multi-agent settings also points to the issue of coordination and robustness. We will leave for future work the methods which can make the stability in the non-stationary environment and also improve the generalization ability of learned policies. With these challenges, we can make the PPO based methods be more effective for more complex reinforcement learning tasks.

References

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms, " arXiv preprint, arXiv: 1707.06347, 2017.
[2] Sha, S., Liu, Y., and Lei, B., "Dynamic Proximal Policy Optimization: Enhancing PPO with Adaptive Entropy and Smooth Clipping, " Neurocomputing, vol. 674, pp. 132861–132861, 2026.
[3] Wang, H., "Improved PPO Algorithm Based on Adaptive Penalty Mechanism, " 2026.
[4] Su, B., "Research on Multi-Agent Reinforcement Learning and PPO-Based Improvements, " 2026.
[5] Khatami, A. H., "CMA-MAPPO: Integrating Covariance Matrix Adaptation Evolution Strategy with Multi-Agent Proximal Policy Optimization for Enhanced Exploration in Sparse-Reward Environments, " Swarm and Evolutionary Computation, vol. 102, pp. 102330–102330, 2026.
[6] C. Jiaju, Z. Chai, Z. Jue, Y. Hao, Y. Zheng, and D. Zhao, "A Survey of Cooperative Multi-Agent Reinforcement Learning for Multi-Task Scenarios, " Artificial Intelligence Science and Engineering, 2025.
[7] Y. Wang, H. He, and X. Tan, "Truly Proximal Policy Optimization, " in Proceedings of the AAAI Conference on Artificial Intelligence, 2020.
[8] J. Dai, J. Ji, L. Yang, Q. Zheng, and G. Pan, "Augmented Proximal Policy Optimization for Safe Reinforcement Learning, " in Proceedings of the AAAI Conference on Artificial Intelligence, 2023.
[9] C. Yu, A. Velu, E. Vinitsky, S. Gao, Y. Wang, A. Bayen, and Y. Wu, "The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games, " in Advances in Neural Information Processing Systems (NeurIPS), vol. 35, pp. 24611–24624, 2022.
[10] K. Cobbe, O. Klimov, C. Hesse, T. Kim, and J. Schulman, "Quantifying Generalization in Reinforcement Learning, " in Proceedings of the 36th International Conference on Machine Learning (ICML), 2019.
[11] E. Brunskill and L. Li, "Sample Complexity of Multi-Task Reinforcement Learning, " arXiv preprint arXiv: 1309.6821, 2013.