Proceedings of the 4th International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/2026.38466

A Study of Randomization-Based Improvement in CartPole

Yifei Wang

School of Mathematics, Shanghai University of Finance and Economics, Shanghai, China
2023111662@stu.sufe.edu.cn

Abstract. This paper studies whether reset randomization can improve the robustness of PPO in the CartPole-v1 task. CartPole is easy to solve under the default setting, but that easy setting does not always show whether a learned policy can still perform well when the starting state becomes less favorable. To address this issue, the study keeps the PPO algorithm unchanged and focuses on the environment reset process. A simple curriculum randomization strategy is used: the initial-state bound is widened gradually during training, from +/-0.05 to +/-0.10 and then to +/-0.15. The experiment compares baseline PPO and randomized PPO under a multi-seed, multi-bound evaluation design. In addition to training curves, both methods are tested under stronger reset bounds up to +/-0.40. The results show that both methods reach ceiling-level performance under mild settings, so the difference is hard to observe there. Once the evaluation becomes more difficult, the randomized method shows better robustness at part of the harder range, although the advantage is not consistent at every bound. The study suggests that randomization can be useful in CartPole, but its value depends on whether the evaluation protocol is strong enough to reveal meaningful differences.

Keywords: Cartpole, reinforcement learning, proximal policy optimization, randomization

1. Introduction

Reinforcement learning (RL) studies how an agent learns to make decisions through repeated interaction with an environment. Instead of being given the correct answer in advance, the agent observes the current state, takes an action, receives a reward, and gradually adjusts its policy to improve long-term return [1]. CartPole is one of the most common introductory benchmarks because it is simple, fast, and easy to understand. In this task, the agent moves a cart left or right to keep a pole balanced upright. Although the task is straightforward, it shows the basic RL loop clearly and is often used to evaluate the behavior and stability of control algorithms [2].

Proximal Policy Optimization (PPO) is now one of the most widely used methods in reinforcement learning because it usually trains well and remains relatively stable [3]. PPO limits how much the policy can change in a single update, which helps reduce the instability seen in many earlier policy-gradient methods. It is also commonly combined with Generalized Advantage Estimation (GAE), which provides a practical balance between bias and variance during learning [4]. Open-source libraries such as Stable-Baselines3 have made PPO easier to reproduce and compare across environments [5].

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.38466

Narvekar et al. examined curriculum learning as a way to let agents begin with easier tasks and move gradually toward harder ones [6]. Chen et al. showed that domain randomization can improve generalization by exposing a model to a wider range of conditions during training [7]. Henderson et al. and Agarwal et al. further showed that deep reinforcement learning results can be highly sensitive to evaluation settings, random seeds, and statistical reporting [8, 9]. These studies provide useful background for the present work, but they also leave an important question open. In a simple benchmark such as CartPole, a policy may reach the maximum score under the default reset setting without being tested carefully under harder initial states.

This paper studies randomization in CartPole through a multi-seed and multi-bound design. Rather than asking only whether PPO can solve the default benchmark, it asks whether changes in the reset randomization range affect performance, stability, and robustness. This question matters because a perfect score under the default task does not necessarily mean that the learned policy is robust. It also offers a practical way to examine how the size of the initial-state range influences the final result.

2. Methodology

2.1. CartPole-v1 environment

This study uses Gymnasium CartPole-v1 as the experimental environment. Figure 1 shows a simple schematic of the CartPole system used in this task. CartPole is a classic control task in which the agent moves a cart left or right to keep a pole balanced upright. The observation space contains four continuous state variables: cart position, cart velocity, pole angle, and pole angular velocity. The action space is discrete and contains only two actions, corresponding to pushing the cart left or right. At each time step, the agent receives a reward of +1 as long as the pole stays balanced and the episode has not terminated. An episode ends when the pole angle exceeds the failure threshold, the cart moves outside the allowed range, or the maximum episode length of 500 steps is reached. Because the task is simple, fast to train, and easy to visualize, it works well for studying how reset randomization affects both learning and evaluation [2, 10].

Under the standard Gymnasium setting, the initial state variables are sampled from a narrow range near zero. Most training episodes therefore begin under relatively easy conditions, with the cart close to the center and the pole already near vertical. A policy can therefore reach a perfect score under the default setting while still being tested only in a narrow part of the valid state space. This feature of CartPole matters here because it creates a gap between standard benchmark success and broader robustness.

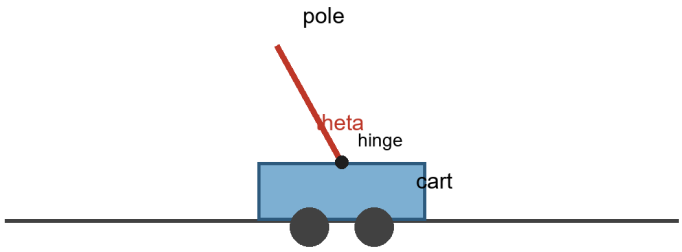


Figure 1. Simple schematic of the CartPole-v1 environment (picture credit: original)

DOI: 10.54254/2755-2721/2026.38466

2.2. PPO method

Proximal Policy Optimization (PPO) is the baseline reinforcement learning method used in this study. The implementation is based on Stable-Baselines3 with the MlpPolicy setting [3, 5]. PPO is an on-policy method that updates the policy using collected trajectories while limiting how much the policy can change in a single update. That constraint helps make training more stable than in many earlier policy-gradient methods. In this experiment, the baseline training seed is fixed at 42 to keep the run reproducible, and the total training budget is 50,000 time steps.

The main hyperparameters are learning rate 3×10^{-4}, rollout length n_steps=1024, batch size 64, and discount factor gamma=0.99. A Monitor wrapper is added to record full-episode rewards during training, and a custom callback stores the reward of each completed episode. This makes it possible to draw a training curve and examine the full learning process, from unstable early exploration to later stable performance. After training, the saved model is evaluated in deterministic episodes. During evaluation, the model always chooses the greedy action rather than sampling from the policy distribution, because the purpose is to measure control quality rather than action noise.

2.3. Randomization-based improvement

The optimization introduced in this study does not replace PPO with a new algorithm. It changes how the environment generates starting states and asks whether broader randomization can serve as a practical way to improve robustness. The method is a simple curriculum-style randomization scheme. In implementation terms, a wrapper is added around the original CartPole environment and intercepts the reset process. The initial-state sampling range is widened step by step during training instead of staying fixed at the default narrow setting.

The wrapper tracks the cumulative number of interaction steps. During the first third of training, the reset bound remains at +/-0.05, which is close to the default CartPole setting and lets the agent learn basic balancing behavior first. In the middle stage, after one third of the training budget has been used, the reset bound expands to +/-0.10. In the late stage, after two thirds of the total steps have been used, the bound expands again to +/-0.15. At the start of each episode, these low and high values are passed into Gymnasium reset options so that the environment samples a wider initial state for cart position, velocity, pole angle, and angular velocity.

This design is useful for two reasons. First, the modification is deliberately limited. The neural network structure, reward definition, action space, PPO objective, and update rule all remain unchanged. The experiment therefore asks whether a different training distribution can help the same PPO method handle a wider set of initial states. Second, the staged design avoids the main weakness of full randomization. If the environment becomes too difficult from the beginning, an immature policy may fail too quickly to collect useful trajectories. Starting from an easier range and widening the reset bound gradually follows the logic of curriculum learning while keeping the intervention limited to the reset mechanism [6].

2.4. Evaluation protocol

To make the comparison between methods clearer, the evaluation range is extended beyond the mild setting used in the basic test. The trained models are still compared through their training curves, but the main robustness test is no longer limited to reset bounds up to +/-0.20. Instead, both trained models are evaluated under a stronger sequence of reset bounds: +/-0.05, +/-0.10, +/-0.15, +/-0.20, +/-0.25, +/-0.30, +/-0.35, and +/-0.40. For each bound, 40 deterministic episodes are run, and the

DOI: 10.54254/2755-2721/2026.38466

mean return, standard deviation, and minimum return are recorded. This choice follows recent work that emphasizes stronger evaluation and more careful statistical reporting in reinforcement learning [9, 11].

The purpose of this stronger protocol is straightforward. The test condition should be difficult enough to show whether baseline PPO and randomized PPO still behave similarly once the initial states move farther away from the usual easy region. The experiment therefore goes beyond asking whether the model can solve default CartPole. It tracks how performance changes as the reset distribution becomes progressively more difficult. In short, the baseline uses standard PPO on default CartPole, while the improved method keeps PPO unchanged, adds a controlled curriculum randomization of the initial-state range, and evaluates both methods under a more demanding test protocol.

3. Results and discussion

Figure 2 shows the baseline training curve for PPO on CartPole. In the early stage, episode rewards remain low and fluctuate because the agent has not yet learned an effective balancing strategy. In the middle stage, the moving average begins to rise more clearly, showing that the policy is starting to connect state information with better left-right control. In the late stage, the curve rises sharply and reaches the solved region above the 475 threshold. Under the default evaluation setting, the baseline model achieves a reward of 500 in every deterministic test episode, confirming that the standard PPO implementation is enough to solve the default CartPole benchmark.

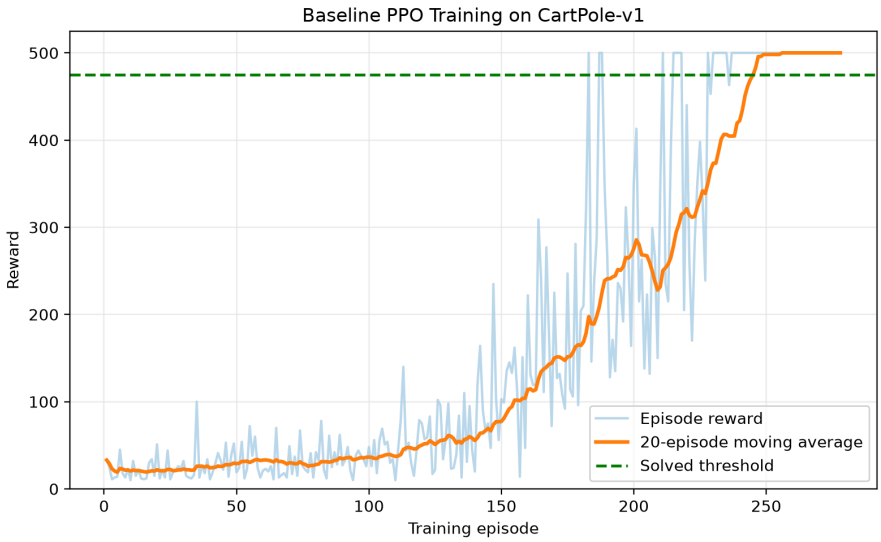


Figure 2. Baseline PPO training curve on CartPole-v1 (picture credit: original)

Figure 3 shows the training curve of the curriculum-randomized PPO. It reaches the same perfect result under the mild evaluation setting. Its training curve is slightly noisier in the later middle stage and early late stage, which is expected because the model is gradually exposed to harder starting states and therefore sees a more diverse set of trajectories during training. Randomization does not automatically raise the final score under easy conditions, but it does change the structure of the training experience and prepares the model for a broader range of initial states.

DOI: 10.54254/2755-2721/2026.38466

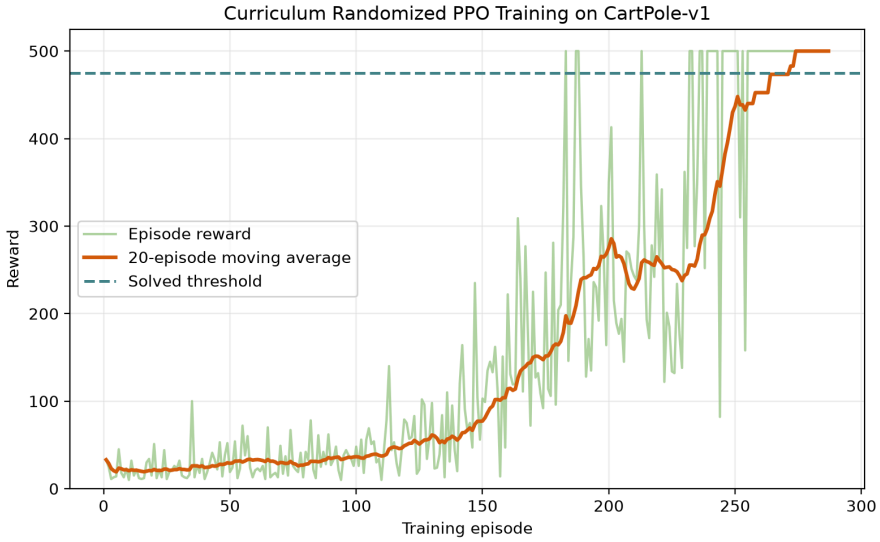


Figure 3. Curriculum randomized PPO training curve on CartPole-v1 (picture credit: original)

Figure 4 presents the mean return of the two methods under stronger reset bounds. When the evaluation bounds are extended beyond +/-0.20, the difference between the two methods becomes easier to see. From +/-0.05 to +/-0.20, both methods still achieve a mean return of 500.00, which shows that this part of the evaluation range is too easy to separate them. At +/-0.25, however, randomized PPO reaches a higher mean return than baseline PPO, 437.63 versus 412.68, and it also has a smaller standard deviation, 165.03 versus 189.60. At +/-0.35, the gap grows again: randomized PPO reaches 325.35, while baseline PPO falls to 238.03. These results suggest that curriculum randomization can improve robustness under some moderately difficult out-of-distribution starts.

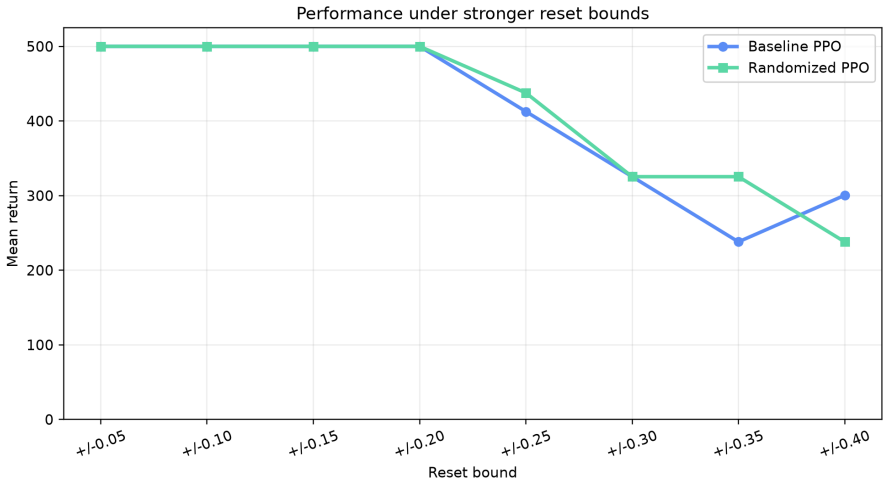


Figure 4. Mean return of baseline PPO and randomized PPO under stronger reset bounds (picture credit: original)

Figure 5 shows the bar comparison together with the variability of the results at each bound. The stronger test also shows that the advantage of randomization is not consistent across every hard condition. At +/-0.30, both methods have the same mean return, 325.35. At +/-0.40, baseline PPO performs slightly better, with a mean return of 300.40 compared with 238.03 for randomized PPO.

DOI: 10.54254/2755-2721/2026.38466

Both methods also show very large standard deviations and can fail completely in some episodes, as indicated by the minimum return of 1.00. The fixed curriculum used here therefore improves robustness in part of the difficult range, but it does not produce a policy that dominates under every extreme condition.

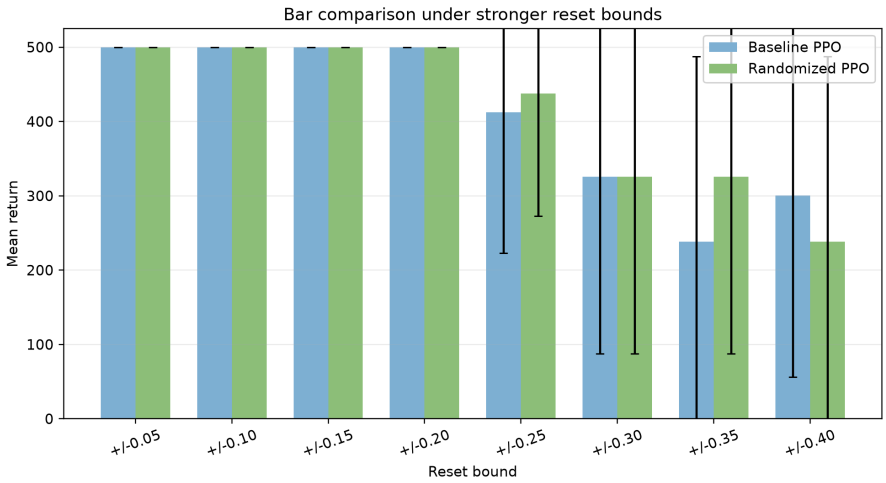


Figure 5. Bar comparison with standard deviations under stronger reset bounds. The black lines are error bars showing the standard deviation over 40 deterministic evaluation episodes at each reset bound (picture credit: original)

The experimental results suggest that the value of randomization becomes clearer under genuinely challenging reset conditions. Under mild bounds, both baseline PPO and randomized PPO remain at ceiling-level performance, so final reward alone reveals very little difference. Once the reset bounds become larger, the randomized method begins to show an advantage in part of the harder range. This suggests that broader exposure to initial states can improve robustness when the policy is tested on more demanding starts.

However, the randomized method used here is a fixed three-stage curriculum rather than an adaptive scheme, so its advantage appears only in part of the harder range. The strongest claim supported by the present results is therefore modest: curriculum randomization can improve robustness under some stronger initial-state conditions when the evaluation protocol is demanding enough. In this study, randomization is useful, but that usefulness becomes visible only after the experiment moves beyond the easy ceiling region of CartPole [12].

4. Conclusion

This study examined whether a simple randomization-based change could improve PPO on CartPole without modifying the core learning algorithm. The answer is mixed but still meaningful. Under the default and mildly randomized settings, both baseline PPO and curriculum-randomized PPO achieved near-ceiling performance, which means the standard CartPole benchmark alone was too easy to show a clear difference. After the reset bounds were extended to harder ranges, the effect of randomization became easier to observe. In part of the more difficult range, especially at intermediate stronger bounds, the randomized method achieved higher mean returns and lower variability, which suggests better robustness to unfamiliar initial conditions. At the same time, the improvement was not universal. Under the hardest tested condition, the randomized method did not outperform the baseline consistently. This means the present three-stage curriculum should be

DOI: 10.54254/2755-2721/2026.38466

understood as a useful but limited improvement rather than a complete solution. The main contribution of the paper is therefore twofold. First, it shows that reset randomization is a practical and easy-to-implement way to strengthen CartPole training and testing. Second, it shows that evaluation design matters: if the test remains too easy, meaningful differences in robustness can stay hidden. Future work can build on this by trying wider bounds, adaptive curricula, or more detailed seed-based comparisons.

References

[1] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press. Cambridge, MA.
[2] Farama Foundation. (2026). Cart pole. *Gymnasium Documentation*, Version 1.3.0. Accessed 21 September 2026.
[3] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv Preprint*, arXiv:1707.06347.
[4] Schulman, J., Moritz, P., Levine, S., Jordan, M., & Abbeel, P. (2016). High-dimensional continuous control using generalized advantage estimation. *arXiv Preprint*, arXiv:1506.02438.
[5] Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-Baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research, 22*, 1–8.
[6] Narvekar, S., Peng, B., Leonetti, M., Sinapov, J., Taylor, M. E., & Stone, P. (2020). Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research, 21*, 1–50.
[7] Chen, X., Hu, J., Jin, C., Li, L., & Wang, L. (2022). Understanding domain randomization for sim-to-real transfer. *arXiv Preprint*, arXiv:2110.03239.
[8] Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., & Meger, D. (2018). Deep reinforcement learning that matters. *Proceedings of the AAAI Conference on Artificial Intelligence, 32*, 3207–3214.
[9] Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A., & Bellemare, M. G. (2021). Deep reinforcement learning at the edge of the statistical precipice. *Advances in Neural Information Processing Systems, 34*.
[10] Towers, M., Kwiatkowski, A., Terry, J., et al. (2024). Gymnasium: A standard interface for reinforcement learning environments. *arXiv Preprint*, arXiv:2407.17032.
[11] Eimer, T., Lindauer, M., & Raileanu, R. (2023). Hyperparameters in reinforcement learning and how to tune them. *Proceedings of the 40th International Conference on Machine Learning, Proceedings of Machine Learning Research, 202*, 9104–9149.
[12] Curtis, A., Li, E., Noseworthy, M., Gothoskar, N., Chitta, S., Li, H., Kaelbling, L. P., & Carey, N. E. (2025). Flow-based domain randomization for learning and sequencing robotic skills. *Proceedings of the 42nd International Conference on Machine Learning, Proceedings of Machine Learning Research, 267*, 11692–11709.