Optimistic Proximal Policy Optimization

Takahisa Imagawa [1] **Takuya Hiraoka** [1 2] **Yoshimasa Tsuruoka** [1 3]

Abstract

Reinforcement Learning, a machine learning framework for training an autonomous agent based on rewards, has shown outstanding results in various domains. However, it is known that learning a good policy is difficult in a domain where rewards are rare. We propose a method, optimistic proximal policy optimization (OPPO) to alleviate this difficulty. OPPO considers the uncertainty of the estimated total return and optimistically evaluates the policy based on that amount. We show that OPPO outperforms the existing methods in a tabular task.

1. Introduction

Reinforcement learning is a framework to learn a good policy in terms of total expected extrinsic rewards by interacting with an environment. It has shown super-human performance in the game of Go and in Atari games (Mnih *et al.*, 2015; Silver *et al.*, 2017). In the early days, RL algorithms such as Q-learning, and state-action-reward-state-action (SARSA) (Sutton *et al.*, 1998), and recently, more sophisticated algorithms have been proposed. Among the latter, proximal policy optimization (PPO) is one of the most popular algorithms, because it can be used in a variety of tasks such as Atari games and robotic control tasks (Schulman *et al.*, 2017).

However, learning a good policy is difficult when the agent rarely receives extrinsic rewards. Existing methods alleviate this problem by adding another type of reward called intrinsic reward. For example, as an intrinsic reward, Pathak *et al.* (2017) and Burda *et al.* (2019a) use prediction error of the next state, and Burda *et al.* (2019b) use evaluation of state novelty. However, these methods are not based on solid theoretical backgrounds.

[1]National Institute of Advanced Industrial Science and Technology, Tokyo, Japan [2]NEC Central Research Laboratories, Kanagawa, Japan [3]The University of Tokyo, Tokyo, Japan. Correspondence to: Takahisa Imagawa <imagawa.t@aist.go.jp>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

Uncertainty Bellman exploration (UBE) is another method to alleviate the sparse reward problem, which has a more solid theoretical background (O'Donoghue *et al.*, 2017). UBE evaluates the value of a policy higher when the estimation of the value is more uncertain, like in "optimism in face of uncertainty" in multi-armed bandit problems (Bubeck *et al.*, 2012). O'Donoghue *et al.* (2017) showed a relationship between the local uncertainty and the uncertainty of the expected return and applied the uncertainty estimation to SARSA.

We apply the idea of UBE to PPO and propose a new algorithm named optimistic PPO (OPPO) which evaluates the uncertainty of the total return of a policy and updates the policy in the same way as PPO. By updating the policy like PPO, its policy is expected to be stable, and this allows OPPO to evaluate the uncertainty of estimated values in states that are far from the current state.

2. Background

2.1. Uncertainty Bellman Equation and Exploration

Markov decision processes (MDPs) are models of sequential decision-making problems. In this paper, we focus on an MDP with a finite horizon, state, and action space. An MDP is defined as a tuple, $\langle \mathcal{S}, \mathcal{A}, r, T, \rho, H \rangle$, where $\mathcal{S}$ is a set of possible states, $\mathcal{A}$ is a set of possible actions; and r is a reward function $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$, which defines the expected reward when the action is taken at the state; T is a transition function $\mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$, which defines the transition probability to the next state when the action is taken at the current state; ρ is a probability distribution of the initial state, and $H \in \mathbb{N}$ is the horizon length of the MDP, i.e. the number of actions until the end of an episode.

The objective of an agent/learner is to learn a good policy in terms of expected total return. Formally, policy $\pi_\theta(a|s)$ $(s \in \mathcal{S}, a \in \mathcal{A})$ is the probability of taking action a at state s, where θ is a set of parameters that determines the probability (for the sake of simplicity, we often omit θ). The Q-value $Q^{h,\pi}_{(s,a)}$, $(Q^{H+1,\pi}_{(s,a)} := 0)$ is an expected total return when the agent is at state s, time-step h, takes action a, and follows policy π after taking action a.

Let us assume the Bayesian setting of Q-value estimation, where there are priors and posteriors over the mean reward

function r and the transition function T. Let $\hat{r}$ be the sampled reward function, $\hat{T}$ be the sampled transition function from prior or posterior, and $\mathcal{F}_\tau$ be the sigma-algebra of all data (e.g. states, actions, rewards) earned by τ times sampling. It is known that there exists a unique $\hat{Q}^{h,\pi}_{(s,a)}$ that satisfies the Bellman equation,

$$\hat{Q}^{h,\pi}_{(s,a)} = \hat{r}(s,a) + \sum_{s',a'} \pi(a|s)\hat{T}(s,a,s')\hat{Q}^{h+1,\pi}_{(s',a')}, \quad (1)$$

for all s and a, for $h = 0, \ldots, H$, where $\hat{Q}^{H+1,\pi}_{(s,a)} = 0$. O'Donoghue *et al.* (2017) extend this Bellman equation to the variance/uncertainty of $\hat{Q}^{h,\pi}_{(s,a)}$.

To prove theoretical results, let us assume that the state transition of the MDP is a directed acyclic graph (DAG) and that expected reward $r(s,a)$ is bounded for all states and actions. We denote the conditional variance of a random variable x as

$$\mathbf{var}_\tau x := \mathbb{E}((x - \mathbb{E}(x|\mathcal{F}_\tau)|\mathcal{F}_\tau)^2. \quad (2)$$

We denote the maximum of Q-value as $Q_{\max}$ and $\nu_\tau(s,a)$ as

$$\mathbf{var}_\tau \hat{r}(s,a) + {Q_{\max}}^2 \sum_{s'} \frac{\mathbf{var}_\tau \hat{T}(s,a,s')}{T_\tau(s,a,s')}, \quad (3)$$

where $T_\tau(s,a,s') := \mathbb{E}_{\hat{T}}[\hat{T}(s,a,s')|\mathcal{F}_\tau]$. The Q-value satisfies the following equation (O'Donoghue *et al.*, 2017).

Theorem 1. *For any policy π, there exists a unique $Q^{h,\pi}_{2,\tau}$ that satisfies the uncertainty Bellman equation,*

$$Q^{h,\pi}_{2,\tau}(s,a) = \nu_\tau(s,a) + \sum_{s',a'} \pi(a'|s')T_\tau(s,a,s')Q^{h+1,\pi}_{2,\tau}(s',a') \quad (4)$$

for all (s,a) and $h = 1, \ldots, H$, where $Q^{H+1,\pi}_{2,\tau} = 0$, and $Q^{h,\pi}_{2,\tau} \geq \mathbf{var}_\tau \hat{Q}^{h,\pi}$ point-wise.

This theorem shows a relationship between the local uncertainty, $\nu_\tau(s,a)$ and the uncertainty of estimated Q-values.

For convenience of discussion in later sections, we introduce some notations. Let us denote the solution of the Bellman equation,

$$Q^{h,\pi}_{1,\tau}(s,a) = r_\tau(s,a) + \sum_{s',a'} \pi(a'|s')T_\tau(s,a,s')Q^{h+1,\pi}_{1,\tau}(s',a') \quad (5)$$

as $Q^{h,\pi}_{1,\tau}$, where the estimated mean reward, $r_\tau(s,a)$ is $\mathbb{E}_{\hat{r}}[\hat{r}(s,a)|\mathcal{F}_\tau]$. For $i = 1, 2$,

$$V^{h,\pi}_{i,\tau}(s) := \sum_a \pi(a|s)Q^{h,\pi}_{i,\tau}(s,a), \quad (6)$$

$$A^{h,\pi}_{i,\tau}(s,a) := Q^{h,\pi}_{i,\tau}(s,a) - V^{h,\pi}_{i,\tau}(s), \quad (7)$$

$$\eta_{i,\tau}(\pi) := \sum_s \rho(s)V^{0,\pi}_{i,\tau}(s). \quad (8)$$

To estimate $\nu_\tau(s,a)$, O'Donoghue *et al.* (2017) start from the case where the domain is tabular. Let $n_{s,a}$ denote the number of times action a is chosen at state s and let σ_r^2 denote the variance of a reward sampled from the reward distribution. We assume that the reward distribution and its prior is Gaussian, and the prior over the transition function is Dirichlet; then

$$\mathbf{var}_\tau \hat{r}(s,a) \leq \sigma_r^2/n_{sa}, \quad (9)$$

$$\sum_{s'} \mathbf{var}_\tau \hat{T}(s,a,s')/T_\tau(s,a,s') \leq |\mathcal{S}_{s,a}|/n_{sa}, \quad (10)$$

where $|\mathcal{S}_{s,a}|$ is the number of next states reachable from (s,a). Thus, there exists a constant C_u which satisfies $\nu_\tau(s,a) \leq \frac{C_u}{n_{s,a}}$, e.g. $C_u = \sigma_r^2 + Q^2_{\max}|\mathcal{S}_{s,a}|$. Since this exact upper bound is too loose in most cases, UBE heuristically chooses C_u instead of using the parameter assured to satisfy the bound. In a domain other than the tabular, UBE extends the discussion above and uses pseudo-counts to estimate the local uncertainty. O'Donoghue *et al.* (2017) applied UBE to SARSA (Sutton *et al.*, 1998), which is a more primitive algorithm than Proximal Policy Optimization.

2.2. Proximal Policy Optimization

Proximal Policy Optimization (PPO) is a simplified version of trust region policy optimization (TRPO)[1]. Although TRPO shows promising results in control tasks (Schulman *et al.*, 2015a), PPO empirically shows better results in most cases (Schulman *et al.*, 2017). PPO uses a clipped variable as follows, so as not to change policy drastically.

$$L(\theta) = \bar{\mathbb{E}}_h \left[\min\left(l_h(\theta)\bar{A}^h, \text{clip}\left(l_h(\theta), 1-\epsilon, 1+\epsilon\right)\bar{A}^h\right)\right], \quad (11)$$

where θ is the parameters of the policy, h is time-step, $l_h(\theta)$ is $\frac{\pi_\theta(a_h|s_h)}{\pi_{\theta_{\text{old}}}(a_h|s_h)}$, $\bar{A}^h$ is the estimated advantage value, e.g. the estimated value of $A^{h,\pi}_{1,\tau}(s_h,a_h)$ in this paper, and $\bar{\mathbb{E}}_h[\cdot]$ is the empirical average over a batch of samples. The clipping function $\text{clip}(x, 1-\epsilon, 1+\epsilon)$ means $x = 1+\epsilon$ if $x > 1+\epsilon$ and $x = 1-\epsilon$ if $x < 1-\epsilon$. PPO samples the data by executing actions for T time-steps following the policy and repeating it N times. PPO updates the policy by maximizing $[L -$ prediction error of V-value $+$ entropy of policy$]$ in the data.

2.3. Exploration Based on Intrinsic Reward

Random network distillation (RND) is recently proposed for alleviating the problem of sparse reward (Burda *et al.*,

[1] While the original TRPO and PPO are formulated under the assumption that the policy is run for an MDP with an infinite horizon, they have recently been extended in the case of finite horizon (Azizzadenesheli *et al.*, 2018), which is the same setting as ours.

2019b). It has shown outstanding performance in Atari games. RND uses two neural networks called a target network f_t and a predictor network f_p. Each network maps state/observation x to its value $f_t(x)$ or $f_p(x)$. The networks are randomly initialized, and the target network's parameters are fixed, on the other hand, the predictor learns the outputs of the target. The intrinsic reward for observation x is defined as the difference of output $||f_t(x) - f_p(x)||^2$. As a reward, RND uses [extrinsic one + intrinsic one], instead of using only the extrinsic one. RND uses the reward defined above and learns a policy like PPO. RND updates the policy to maximize [PPO's objective − differences of outputs of the networks] in the batch data. It is expected that more observations lead to smaller differences of the outputs, which means the intrinsic reward is smaller. In RND, the intrinsic rewards can be seen as a kind of pseudo-count bonus. However, there is no theoretical discussion about how this bonus should be used.

There are other methods for exploration by the intrinsic rewards. To calculate the intrinsic rewards, Bellemare *et al.* (2016) used context tree switching, and Ostrovski *et al.* (2017) used pixcelCNN. However, those methods depend on visual heuristics and are not straightforward to apply to other tasks than Atari games, e.g. control tasks whose inputs are sensor data. Ecoffet *et al.* (2019) proposed an another method for exploration, which is based on memorization and random search rather than intrinsic reward. Although it shows state-of-the-art performance on Montezuma's Revenge, it is also not straightforward to extend the method to other tasks. Tang *et al.* (2017) proposed a method similar to RND which evaluates the state novelty by using a hash function.

3. Optimistic Proximal Policy Optimization

We propose optimistic proximal policy optimization (OPPO), which is a variant of PPO. OPPO optimizes a policy based on optimistic evaluation of the expected return where the evaluation is optimistic by the amount of the uncertainty of the expected return.

First, we explain its theoretical background. We denote the optimistic value of policy $\tilde{\eta}(\pi)_\tau$ as below:

$$\tilde{\eta}_\tau(\pi) := \eta_{1,\tau}(\pi) + 2\beta\sqrt{\eta_{2,\tau}(\pi)}, \tag{12}$$

where $\beta > 0$ is a hyper-parameter for exploration. Setting the high value to β means emphasizing exploration more than exploitation. Let us denote the value of policy $\hat{\eta}(\pi)$ as $\sum_{s,a} \rho(s)\pi(a|s)\hat{Q}^{0,\pi}_{(s,a)}$. Then the following corollary is derived from Theorem 1.

Corollary 1.

$$\mathbf{var}_\tau\left(\hat{\eta}(\pi)\right) \leq \eta_{2,\tau}(\pi) \tag{13}$$

This corollary shows that $\eta_{2,\tau}(\pi)$ is an upper bound of the uncertainty of the expected return of π. In general, more data lead to more accurate estimation, and this means lower $\nu_\tau(s,a)$ and $\eta_{\tau,2}(\pi)$. Especially if $\nu_\tau(s,a) = 0$, $\eta_{\tau,2}(\pi) = 0$. Also, $0 \leq \mathbf{var}_\tau\left(\hat{\eta}(\pi)\right) \leq \eta_{2,\tau}(\pi)$. Therefore, the difference of $\mathbf{var}_\tau\left(\hat{\eta}(\pi)\right)$ and $\eta_{2,\tau}(\pi)$ decreases to zero as the number of data increases. These facts show that evaluating $\mathbf{var}_\tau\left(\hat{\eta}(\pi)\right)$ by $\eta_{2,\tau}(\pi)$ is reasonable. Besides, $\eta_{1,\tau}(\pi)$ is an estimation of the mean of $\hat{\eta}(\pi)$. Thus, $\tilde{\eta}(\pi)_\tau$ is a form that the estimated return plus its uncertainty and seeking a policy which maximizes $\tilde{\eta}(\pi)_\tau$ is reasonable in terms of "optimism in face of uncertainty".

However, it is difficult to find policy π' which maximizes $\tilde{\eta}_\tau(\pi')$ by directly evaluating $\tilde{\eta}_\tau(\pi')$. Thus, following PPO, OPPO approximates $\tilde{\eta}_\tau(\pi')$ based on the current policy π. Let $\mathcal{L}_\tau(\pi, \pi')$ denote

$$\tilde{\eta}_\tau(\pi) + \sum_{h,s,a} \rho^\pi_h(s)\pi'(a|s)\left(A^{h,\pi}_{1,\tau}(s,a) + \beta\frac{A^{h,\pi}_{2,\tau}(s,a)}{\sqrt{\eta_{2,\tau}(\pi)}}\right). \tag{14}$$

Then the following equations are satisfied.

Theorem 2. *For any parameters of policy ϕ,*

$$\mathcal{L}_\tau(\pi_\phi, \pi_\phi) = \tilde{\eta}_\tau(\pi_\phi) \tag{15}$$

$$\nabla_\theta \mathcal{L}_\tau(\pi_\phi, \pi_\theta)|_{\theta=\phi} = \nabla_\theta \tilde{\eta}_\tau(\pi_\theta)|_{\theta=\phi} \tag{16}$$

Theorem 2 means that $\tilde{\eta}_\tau(\pi')$ can be approximated by $\mathcal{L}_\tau(\pi, \pi')$ with enough accuracy if π and π' are not very different. Therefore, OPPO chooses the next policy π' so as to increase the estimated value of $\mathcal{L}_\tau(\pi, \pi')$ with regularizing the 'similarity' between π and π' by the clipping function introduced in section 2.2.

The objective function of OPPO is the same as L in Equation (11), except that OPPO uses $\tilde{A}^h$ instead of $\bar{A}^h$ in the equation, where $\tilde{A}^h$ is

$$A_1(s_h, a_h) + \beta A_2(s_h, a_h)/\sqrt{\eta_2 + c}. \tag{17}$$

Parameter $c \geq 0$ is introduced for stabilizing the estimation when $\eta_2(\pi)$ is nearly zero. Note that Theorem 2 is valid if the square root in equations (12), (14) are either $\sqrt{\eta_{2,\tau}(\pi)}$ or $\sqrt{\eta_{2,\tau}(\pi) + c}$. The terms, η_2, $A_1(s,a)$ and $A_2(s,a)$, are the estimated values of $\eta_{2,\tau}(\pi)$, $A^{h,\pi}_{1,\tau}(s,a)$ and $A^{h,\pi}_{2,\tau}(s,a)$, respectively, which are calculated based on generalized advantage function estimation (Schulman *et al.*, 2015b). We show the details in A.2. The other parts of the objective function of OPPO are prediction error of V-values and entropy of policy, which are the same as PPO.

Note that simply adding the bonuses ${n_{s,a}}^{-1/2}$ to the extrinsic rewards instead of adding bonuses like UBE and OPPO may be overly optimistic, as shown in an example in

O'Donoghue *et al.* (2017), although ordinary count-based exploration is based on the bonuses (Bellemare *et al.*, 2016; Ostrovski *et al.*, 2017; Tang *et al.*, 2017).

OPPO can be combined with an arbitrary estimator of the local uncertainty. For example, the local uncertainty can be directly evaluated by bootstrap sampling of the reward and transition functions, like the estimators of Q-values in Osband *et al.* (2016). In this paper, instead of the model-based approach, we take a model-free one for simplicity. We use the RND bonus of state s' as the local uncertainty of (s, a) pair, where s' is the next state after (s, a). Although the networks in RND can be easily extended to evalute novelty of (s, a) pair instead of s', we follow the RND original imprementations for a simple and clear comparison. We discuss the difference between the local uncertainty evaluations in A.3. In this case, OPPO is equivalent to RND, if $\beta^2 = c$ and $c \to \infty$. Testing OPPO with various local uncertainty estimators is left for future work. We also tested OPPO with local uncertainties based on exact visitation counts of s' i.e., $\frac{1}{n_{s'}}$.

4. Experiments

4.1. Tabular Domain

First, we examine the efficiency of the proposed algorithms in a tabular domain where visitation counts are easily calculated. We used a domain called a bandit tile. A bandit tile is a kind of a grid world with two tiles exist on which the agent receives a stochastic reward. We show an example of a bandit tile in figure 1. In the figure, 'G' represents the tile and 'S' represents possible initial positions of the agent. The initial position is stochastically chosen among the two 'S' tiles. The reward is sampled from a Gaussian distribution. The mean reward of each 'G' tile is 0.5 and 0.3 and its variance is 0.5. The episode ends when the agent reaches the 'G' tile or 100 time-steps are passed.

We compared OPPO with the bonus based on exact visitation counts to OPPO, RND, and PPO. Figure 2 shows that OPPO is more efficient than RND and also suggests that we can improve OPPO if there is a proper method to estimate local uncertainty.

4.2. Atari Domain

Next, we show experimental results on more complex tasks, Atari games, popular testbeds for reinforcement learning. It has been pointed out that Atari games are deterministic, which is not appropriate for being testbeds, so we added randomness by sticky action (Machado *et al.*, 2018). In the sticky action environment, the current chosen action are executed with the probability $1 - \zeta$ while the most previous action is repeated with the probability ζ. We set $\zeta = 1/4$. We chose six games (Frostbite, Freeway, Solaris, Venture, Montezuma's Revenge, and Private Eye) to evaluate the proposed method and run algorithms until 100 million time-steps in Frostbite and 50 million in the other games. OPPO was more effective than RND at Frostbite in terms of learning speed, although the difference is not so salient as that in the tabular case. The details are shown in figure 4 in Appendix.

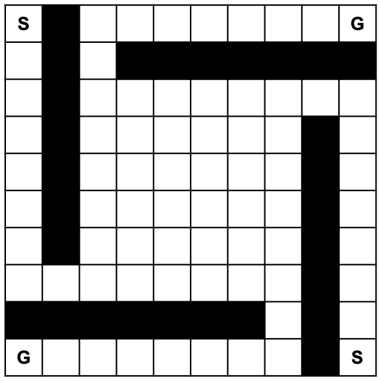


Figure 1: Example of bandit tile domain

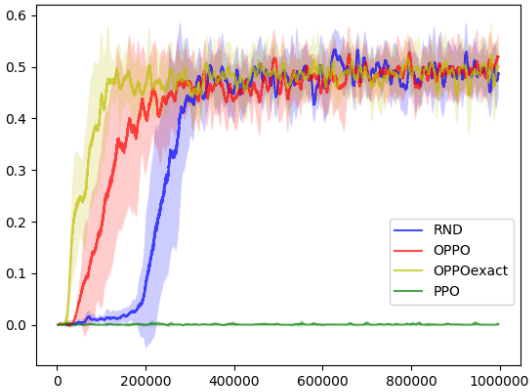


Figure 2: Moving average $\pm$ standard deviation of epsode rewards in bandit tile domain with 10 seeds until 1M time-steps

5. Conclusion

We have proposed a new algorithm, optimisitic proximal policy optimization (OPPO) to alleviate the sparse reward problem. OPPO is an extension of proximal policy optimization and considers uncertainty of estimation of expected total returns instead of simply estimating the returns. OPPO optimistically evaluates the values of policies by the amount of uncertainty and improves the policy like PPO. Experimental results show that OPPO learns more effectively than the existing method, RND, in a tabular domain.

Acknowledgments

Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

References

Kamyar Azizzadenesheli, Manish Kumar Bera, and Animashree Anandkumar. Trust region policy optimization of pomdps. *arXiv preprint arXiv:1810.07900*, 2018.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pages 1471–1479, 2016.

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. *Seventh International Conference on Learning Representations*, 2019.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *Seventh International Conference on Learning Representations*, 2019.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O Stanley, and Jeff Clune. Go-explore: a new approach for hard-exploration problems. *arXiv preprint arXiv:1901.10995*, 2019.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of international conference on Machine learning*, volume 2, pages 267–274, 2002.

Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Brendan O'Donoghue, Ian Osband, Remi Munos, and Volodymyr Mnih. The uncertainty bellman equation and exploration. *arXiv preprint arXiv:1709.05380*, 2017.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In *Advances in neural information processing systems*, pages 4026–4034, 2016.

Georg Ostrovski, Marc G Bellemare, Aäron van den Oord, and Rémi Munos. Count-based exploration with neural density models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2721–2730. JMLR. org, 2017.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, volume 2017, 2017.

John Schulman, Sergey Levine, Pieter Abbeel, Michael I Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of international conference on Machine learning*, volume 37, pages 1889–1897, 2015.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.

Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # Exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in neural information processing systems*, pages 2753–2762, 2017.

Hado P van Hasselt, Arthur Guez, Matteo Hessel, Volodymyr Mnih, and David Silver. Learning values across many orders of magnitude. In *Advances in Neural Information Processing Systems*, pages 4287–4295, 2016.

A. Details of Proposed Method

A.1. Proofs

Corollary 1 is derived from the following relations.

Proof.

$$\mathbf{var}_\tau\left(\hat{\eta}(\pi)\right) = \mathbf{var}_\tau\left(\sum_{s,a}\rho(s)\pi(a|s)\hat{Q}^{0,\pi}_{(s,a)}\right) \tag{18}$$

$$\leq \sum_{s,a}\rho(s)\pi(a|s)\mathbf{var}_\tau\left(\hat{Q}^{0,\pi}_{(s,a)}\right) \tag{19}$$

$$\leq \sum_{s,a}\rho(s)\pi(a|s)Q^{0,\pi}_{2,\tau}(s,a)) \tag{20}$$

$$= \eta_{2,\tau}(\pi) \tag{21}$$

The first inequality is derived from Jensen's inequality, and the second one is derived from Theorem 1. □

For convenience, we introduce some additional notations. Let $\rho^\pi_h(s)$ denote the probability of the agent being at state s at time-step h under the condition $s_0 \sim \rho(\cdot), a_h \sim \pi(\cdot|s_h), s_{h+1} \sim T_\tau(s_h, a_h, \cdot)$ for $h \geq 0$ and expectation under the condition as $\mathbb{E}_{s_0,a_0,\cdots\sim\tilde{\pi}}[\cdot]$. Theorem 2 is derived from the following relations.

Proof. Firstly, we show that $\eta_{i,\tau}(\pi)$ satisfies the following equations,

$$\eta_{i,\tau}(\pi') - \eta_{i,\tau}(\pi) = \sum_{h,s,a}\rho^{\pi'}_h(s)\,\pi'(a|s)\,A^{h,\pi}_{i,\tau}(s,a), \tag{22}$$

which is almost the same as the equations shown in (Kakade and Langford, 2002; Schulman *et al.*, 2015a). Equation (22) is derived as below:

$$\eta_{i,\tau}(\pi') - \eta_{i,\tau}(\pi) = \mathbb{E}_{s_0,a_0,\ldots,\sim\pi'}\left[\sum_{h=0}^{H} r(s_h,a_h) - V^{0,\pi}_{i,\tau}(s_0)\right] \tag{23}$$

$$= \mathbb{E}_{s_0,a_0,\ldots,\sim\pi'}\left[\sum_{h=0}^{H}\left\{r(s_h,a_h) + V^{h+1,\pi}_{i,\tau}(s_{h+1}) - V^{h,\pi}_{i,\tau}(s_h)\right\}\right] \tag{24}$$

$$= \mathbb{E}_{s_0,a_0,\ldots,\sim\pi'}\left[\sum_{h=0}^{H} A^{h,\pi}_{i,\tau}(s_h,a_h)\right] \tag{25}$$

$$= \sum_{h,s,a}\rho^{\pi'}_h(s)\pi'(a|s)A^{h,\pi}_{i,\tau}(s,a). \tag{26}$$

The first equation is derived from the definition of η and the fact that sampling of the initial state only depends on $\rho(\cdot)$, the second one $V^{H+1} = 0$. The third one and the forth one are derived from the definition of $A^{h,\pi}_{i,\tau}$ and $\mathbb{E}_{s_0,a_0,\ldots,\sim\pi'}[\cdot]$, respectively.

For simplicity, we denote π_ϕ as π. By the fact that $\sum_a \pi(a|s)A^{h,\pi}_{i,\tau}(s,a) = 0$ $(i = 1, 2)$,

$$\mathcal{L}_\tau(\pi,\pi) - \tilde{\eta}_\tau(\pi) = \sum_{h,s,a}\rho^\pi_h(s)\pi(a|s)\left(A^{h,\pi}_{1,\tau}(s,a) + \beta\frac{A^{h,\pi}_{2,\tau}(s,a)}{\sqrt{\eta_{2,\tau}(\pi)}}\right) \tag{27}$$

$$= 0. \tag{28}$$

Also,

$$\nabla_\theta \mathcal{L}_\tau(\pi_\phi, \pi_\theta)|_{\theta=\phi} - \nabla_\theta \tilde{\eta}_\tau(\pi_\theta)|_{\theta=\phi} = \nabla_\theta \sum_{h,s,a} \rho_h^\pi(s)\pi_\theta(a|s) \left(A_{1,\tau}^{h,\pi}(s,a) + \beta \frac{A_{2,\tau}^{h,\pi}(s,a)}{\sqrt{\eta_{2,\tau}(\pi)}} \right)\Bigg|_{\theta=\phi}$$

$$- \nabla_\theta \sum_{h,s,a} \rho_h^{\pi_\theta}(s)\pi_\theta(a|s) \left(A_{1,\tau}^{h,\pi}(s,a) + \beta \frac{A_{2,\tau}^{h,\pi}(s,a)}{\sqrt{\eta_{2,\tau}(\pi)}} \right)\Bigg|_{\theta=\phi} \tag{29}$$

$$= - \sum_{h,s} \nabla_\theta \rho_h^{\pi_\theta}(s)|_{\theta=\phi} \sum_a \pi(a|s) \left(A_{1,\tau}^{h,\pi}(s,a) + \beta \frac{A_{2,\tau}^{h,\pi}(s,a)}{\sqrt{\eta_{2,\tau}(\pi)}} \right) \tag{30}$$

$$= 0. \tag{31}$$

The first equation is derived from equation (22). □

A.2. Algorithm

In the batch data, we denote the state, action, and reward at time-step h $(0 \le h \le T)$ and sampled by actor n $(0 \le n \le N-1)$ are $s_h^{(n)}, a_h^{(n)}$, and $r_h^{(n)}$, respectively. Let $r_{1,h}^{(n)}$ denote $r_h^{(n)}$ and $r_{2,h}^{(n)}$ denote the local uncertainty of $(s_h^{(n)}, a_h^{(n)})$. A_i $(i = 1, 2)$ in equation (17) is calculated as below:

$$A_i(s_l^{(n)}, a_l^{(n)}) = \sum_{h=l}^{T-1} (\gamma^i \lambda)^{h-l} \left\{ \gamma^i V_i(s_{h+1}^{(n)}) + r_{i,h}^{(n)} - V_i(s_h^{(n)}) \right\}, \tag{32}$$

where V_i is an estimator of $V_{i,\tau}^\pi$ and γ is a discount factor. The discount factor is often used even if the horizon is finite, so we follow the ordinary implementations. η_2 is calculated as below:

$$\eta_2 = \sum_{n=0}^{N-1} V_2(s_0^{(n)}) + A_2(s_0^{(n)}, a_0^{(n)}) \tag{33}$$

Pseudo code is shown at Algorithm 1.

A.3. Local Uncertainty Estimation

Let $\nu(s')$ denote the local uncertainty based on the next state s' after (s, a) pair. OPPO uses $\nu(s')$ as the local uncertainty of (s, a) instead of $\nu(s, a)$. There is a small gap between the discussion and the implementation of OPPO. However, using $\nu(s')$ is reasonable if the state transition is a tree, a graph without cycles. Using $\nu(s')$ means using the average of $\nu(s')$ as the local uncertainty of (s, a). This can be approximated by $\sum_{s'} T(s, a, s')\nu(s')$. In the tree case, $n_{s'}$ can be approximated by $T(s, a, s')n_{s,a}$. Thus, if $\nu(s') \approx \frac{1}{n_{s'}}$, the local uncertainty of (s, a) can be approximated by $\sum_{s'} T(s, a, s')\nu(s') \approx \sum_{s'} T(s, a, s')\frac{1}{n_{s'}} \approx \sum_{s'} \frac{1}{n_{s,a}} = \frac{|\mathcal{S}_{s,a}|}{n_{s,a}}$. This means that $\nu(s, a)$ can be approximated by the average of $\nu(s')$, if $C_u = |\mathcal{S}_{s,a}|$.

B. Further Investigation in Tabular Domain

To confirm the validity of using RND bonus as visitation counts, we measured a ratio $\frac{\text{RND bonus}}{1/n_{s'}}$ to check if it is stable at around one in the bandit tile domain. Figure 3 shows that the ratio was around 1 for millions of time-steps, although it was high at the beginning and nearly zero at the end. It can be considered that OPPO is worse than OPPO with the exact count bonus by the amount of the overvaluation, and that the undervaluation was not harmful because it occured after learning the policy to the best tile.

C. Details of Results in Atari Games

We compared OPPO with RND in the six Atari games. In the original RND implementation, a reward clipping technique which transforms negative/positive extrinsic reward to $\{-1, 1\}$ is used, so we also used this technique in OPPO and RND.

Algorithm 1 OPPO

initialize the parameters of the policy network, the V-value estimators and the local uncertainty estimator.
for $\tau = 0, \ldots$ **do**
  for $n = 0, \ldots N-1$ **do**
    for $t = 0, \ldots T-1$ **do**
      make a batch data τ by sampling action $a_t^{(n)}$ from $\pi(\cdot|s_t^{(n)})$, exectuting $a_t^{(n)}$, and receiving next state $s_{t+1}^{(n)}$, extrinsic reward $r_t^{(n)}$, and the local uncertainty of $(s_t^{(n)}, a_t^{(n)})$.
    end for
  end for
  update policy so as to maximize the objective function of OPPO based on the data.
end for

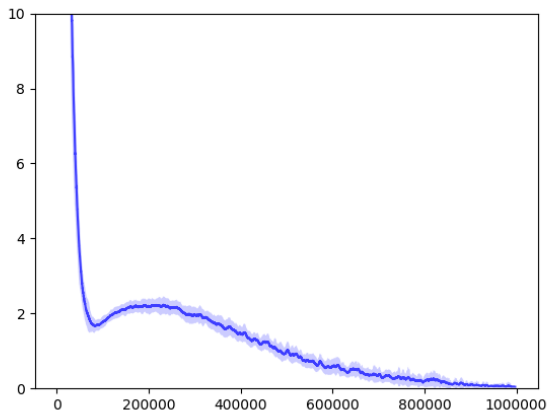
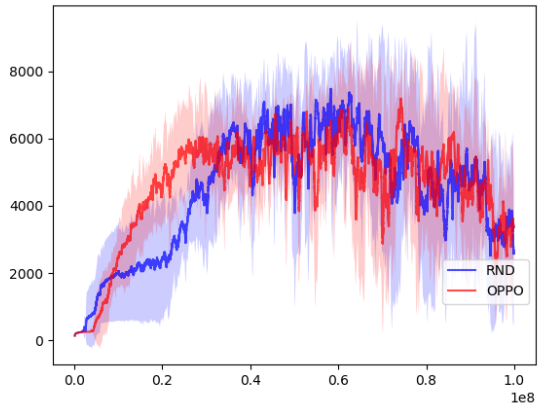


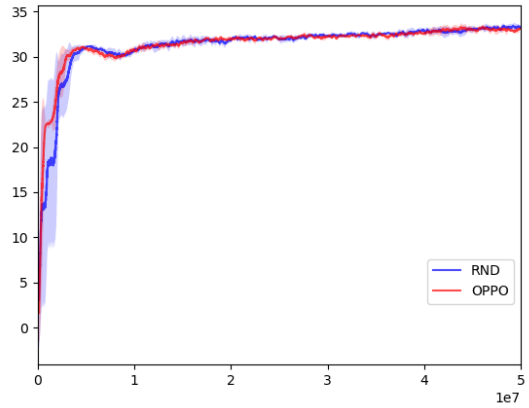
Figure 3: Moving average of the average of $\frac{\text{RND bonus}}{1/n_{s'}}$ in batch data.

Note that we use a frame skipping technique, and the number of the frame skips is four; so one time-step is equal to or less than four frames (it is less than four if the episode ends at a skipped frame).

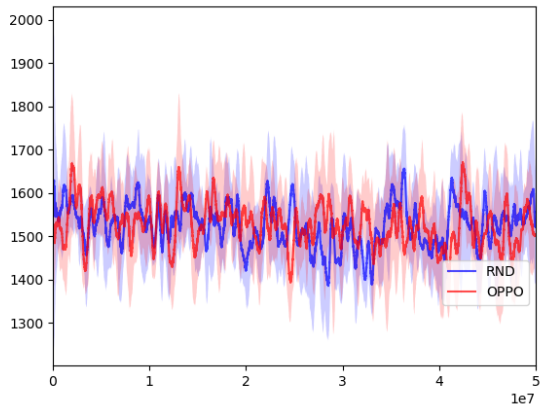
Figure 4 shows that OPPO learns more effectively than RND in Frostbite although there is only slight difference with the other games. Also, Figure 4 shows that exrinsic rewards decrease in Frostbite. One of the reason for the decrease may be the reward clipping, although further investigation is needed to confirm that. By the reward clipping, the agent learns a policy to receive positive rewards with high frequency, not high returns. The agent may learn the policy with the same frequency of rewards but with a small total return, as it receives data. Note that there are small and large rewards in Frostbite, and that a novel states leads to a higher reward in most Atari games (Burda *et al.*, 2019a). This problem can be alleviated by rescaling the reward by considering the amount of reward, e.g. PopArt (van Hasselt *et al.*, 2016), which is left for future work.



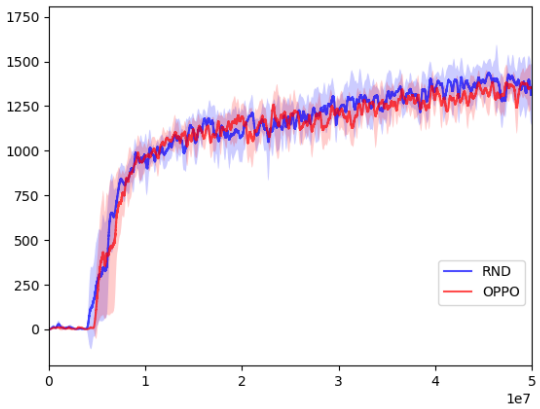
(a) Frostbite



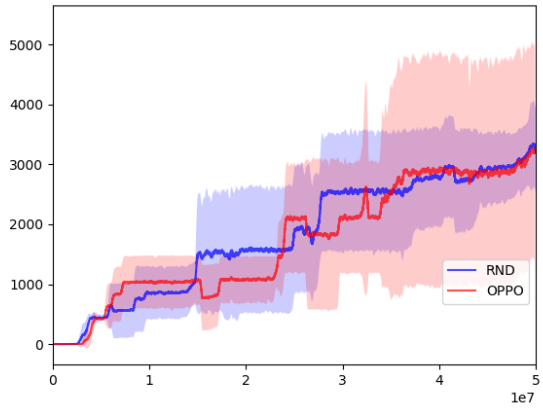
(b) Freeway



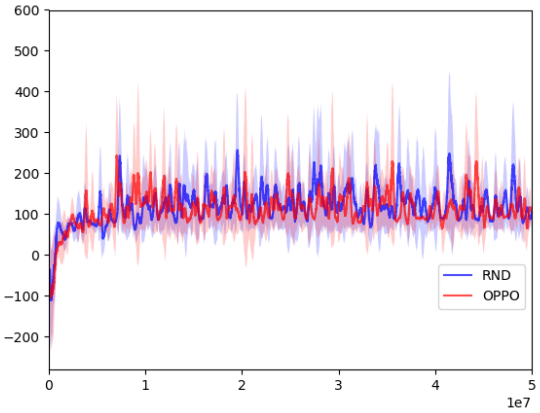
(c) Solaris



(d) Venture



(e) Montezuma's Revenge



(f) Private Eye

Figure 4: Moving average $\pm$ standard deviation of episode rewards with 5 seeds until 50M time-steps (100M time-steps in Frostbite)