Reinforcement Learning with Derivative-Free Exploration[*]

Extended Abstract

Xiong-Hui Chen
Nanjing University
Nanjing, China
chenxh@lamda.nju.edu.cn

Yang Yu
Nanjing University
Nanjing, China
yuy@lamda.nju.edu.cn

ABSTRACT

Effective exploration is key to sample-efficient reinforcement learning. While the most popular general approaches (e.g., ϵ-greedy) for exploration are still of low efficiency, derivative-free optimization also invents efficient ways of exploration for better global search, which reinforcement learning usually desires for. In this paper, we introduce a derivative-free based exploration called DFE as a general efficient exploration method for early-stage reinforcement learning. DFE overcomes the disadvantage of optimization inefficiency and pool scalability in pure derivative-free optimization based reinforcement learning methods. Our experiments show DFE is an efficient and general exploration method through exploring trajectories with DFE in deterministic off-policy method DDPG and stochastic off-policy method ACER algorithms, and applying in Atari and Mujoco, which represent a high-dimensional discrete-action environment and a continuous control environment.

KEYWORDS

reinforcement learning; derivative-free optimization; exploration

ACM Reference Format:
Xiong-Hui Chen and Yang Yu. 2019. Reinforcement Learning with Derivative-Free Exploration. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 3 pages.

1 INTRODUCTION

Exploration is essential in reinforcement learning [9], which determines the efficiency of convergence and the quality of the outcome policy. We presume that searching better performance trajectories in global is critical to do efficient exploration since it helps to jump out of local traps. We also notice that exploration is not only required by reinforcement learning but also needed in derivative-free optimization, where smart global exploration mechanisms have been designed, such as the Bayesian optimization [7] and the classification-based optimization [2, 11]. However, as they don't rely on any gradient information of data, pure derivative-free optimization methods often have poor efficiency than derivative-based algorithms.

In this paper, we propose the *derivative-free exploration* (DFE) for reinforcement learning, which regards the exploration problem as an extra optimization problem and adopts the exploration components in the state-of-the-art derivative-free optimization for improving the efficiency of reinforcement learning. Instead of both exploring and learning with derivative-free optimization directly, DFE just uses derivative-free optimization to *guide exploration via its smart exploration mechanism.* In DFE, derivative-free optimization searches for high-performance exploration policy in global way via set sampling trajectories performance as the fitness value of that exploration policy. Thus, we can avoid the drawback of exploration locality in derivative-based methods. Besides, by optimizing target policy with one of derivative-based off-policy policy gradient algorithm (e.g., DDPG) via those exploration trajectories, we can avoid the disadvantage of optimization inefficiency in pure derivative-free optimization.

2 DERIVATIVE-FREE LEARNING OF POLICY

A policy representation method should be designed to learn an exploration policy with derivative-free optimization in DFE. At present, derivative-free optimization learns a policy via adjusting network parameters [3, 6, 8]. It's effective but will take two problems: first, the influence of adjusting parameters is unpredictable. Some parameters may have little effect on output actions while others may change a lot. Besides, as the neural network becomes more complex, we need to search rapidly increased parameters, which directly affects scalability.

We propose a policy learning method called *Direct Action Optimization* (DAO) to solve these problems. DAO is a modification based on the derivative-free optimization framework to learn policy. In parameterized policy case, policy can be represented with a state-action pair data set $\mathcal{D} := \{(s, a)\}$ via supervised learning:

$$\mathcal{L}(\theta_{\pi_{\text{explore}}}) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[(\pi_{\text{explore}}(s) - a)^2]$$

In DAO process, a set of states are sampled and fixed from replay buffer. Action labels (corresponding to each state) are regarded as the parameters to learn. The derivative-free optimization can learn a policy by tunning those action labels.

There are two advantages to use DAO in DFE framework: first, it effectively impacts neural network because actions to states which sampled from replay buffer are more associated with policy performance. Second is reducing learning parameter dimensions. For complex network, the needed number of state-action pairs is less than that of network parameters. So it has better scalability and is optimized easily.

However, better exploration policy is hard to be found since target policy is improved by RL optimization algorithm gradually. Thus we construct exploration policy in a residual way (inspired by ResNet [1]). In particular, exploration policy regards target policy as a baseline policy, and adds another network term π_{DF} (optimized

[*] The full version of this work is titled "Derivative-Free Exploration for Reinforcement Learning".

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

by DAO) to bias the output of that baseline policy (e.g., bias the action value of baseline policy for deterministic policy and the weights for Softmax policy). Instead of doing exploration with the output of π_{DF} directly, as a bias term, the network complexity of π_{DF} to construct a better policy is reduced. We name this *Policy Combination Technique* (PCT).

3 DERIVATIVE-FREE EXPLORATION

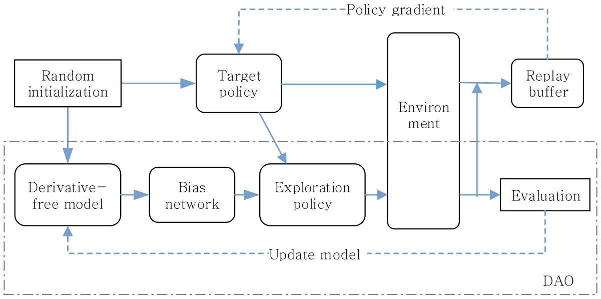


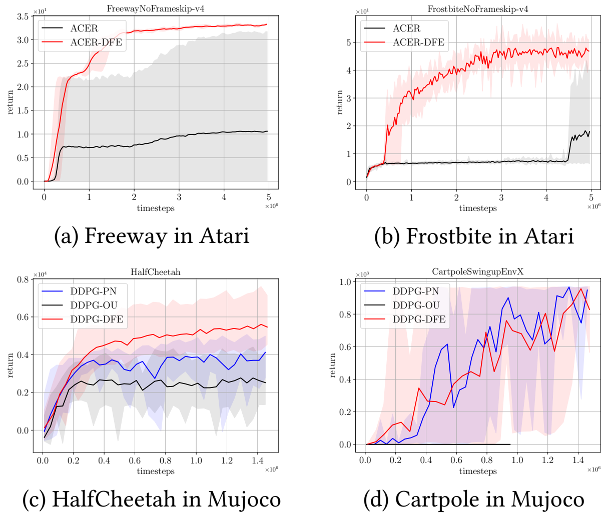
Figure 1: The overview of DFE implementation

We select *sequential randomized coordinate shrinking* (SRACOS) [2] as the implementation of derivative-free optimization, which is one of state-of-the-art classification-based derivative-free optimization methods. Classification based methods often learn a classification model h to classify solutions (i.e., exploration policy in our case) into two set, positive or negative, and then sample from positive areas.

Now the overview of DFE can be seen in Figure 1. A target policy and a bias network are used to construct an exploration policy. Therein, target policy is what optimized by an off-policy policy gradient algorithm and the bias network generated by the SRACOS model. SRACOS fixes states and searches corresponding actions from a randomized coordinate shrinking region which is constructed by discriminator h. The SRACOS model needs a fitness value for each group of action labels to update h. The fitness value is set to the average long-term reward running with the exploration policy created by these actions. At the same time, target policy and exploration policy run to get trajectories in parallel. All trajectories are stored in two replay buffers respectively. An off-policy policy gradient algorithm is used to optimize target policy with these trajectories.

4 EXPERIMENTS

We combine DFE with DDPG [4] method (called DDPG-DFE) in continuous control tasks (Mujoco) and ACER [10] in discrete-action environments (Atari), since DDPG is a deterministic-policy RL algorithm and ACER is a stochastic-policy RL algorithm.



(a) Freeway in Atari

(b) Frostbite in Atari

(c) HalfCheetah in Mujoco

(d) Cartpole in Mujoco

Figure 2: Illustration of the average cumulative reward with timesteps. Shaded areas include the max/min score in all seeds.

Results in ACER We test 42 Atari games with ACER-DFE and the vanilla ACER method. Figure 2 (a) and (b) depicts the result in Atari. Overall, around 76% (32/42) games perform better or equal to the original method, and around 56% (23/42) games get better performance over 15%. Near 28% (12/42) games have significant performance improvement over 50%. The definition of performance improvement percentage is as follows:

$$p(\pi_{\text{DFE}}, \pi_{\text{comp}}) = \frac{\text{eval}(\pi_{\text{DFE}}) - \text{eval}(\pi_{\text{rand}})}{\text{eval}(\pi_{\text{comp}}) - \text{eval}(\pi_{\text{rand}})} \times 100\% \qquad (1)$$

where eval(π) function evaluates the average return in several episodes running with policy π. eval(π_{rand}) denotes the performance of random policy in this environment.

Results in DDPG Figure 2 (c) and (d) depicts Mujoco result on DDPG-DFE. DDPG-OU denotes DDPG method exploring with Ornstein-Uhlenbeck process [4] and DDPG-PN denotes DDPG method exploring with the parameter noise method [5]. Our experiments show that DDPG-DFE outperforms vanilla DDPG-OU *in all environments*. While the DDPG-PN method can also improve performance, DDPG-DFE performs significantly better (not worse at least) than DDPG-PN method. In particular, 50% (4/8) tasks get better performance *over 90%* compared with DDPG-OU, while the percentage is 25% (4/8) compared with DDPG-PN.

5 CONCLUSION

In this work, we propose *Derivative-free Exploration* (DFE) to solve the general exploration problem. DFE solves the inefficiency problem of pure derivative-free optimization and increases the scalability, thus making it possible to apply in high-dimensional environments. Experiments show that DFE is applicable to both stochastic policy and deterministic policy; besides, can work well in high-dimensional discrete-action and continuous control tasks. It is worth to do further research on regarding the exploration problem as an extra optimization problem with the goal of finding better exploration policy in global.

REFERENCES

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 29th. IEEE Conference on Computer Vision and Pattern Recognition.* 770–778.

[2] Yi-Qi Hu, Hong Qian, and Yang Yu. 2017. Sequential Classification-Based Optimization for Direct Policy Search. In *Proceedings of the 31st. AAAI Conference on Artificial Intelligence.* 2029–2035.

[3] Shauharda Khadka and Kagan Tumer. 2018. Evolution-Guided Policy Gradient in Reinforcement Learning. (2018).

[4] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* (2015).

[5] Matthias Plappert, Rein Houthooft, Prafulla Dhariwal, Szymon Sidor, Richard Y. Chen, Xi Chen, Tamim Asfour, Pieter Abbeel, and Marcin Andrychowicz. 2018. Parameter Space Noise for Exploration. In *Proceedings of the 6th. International Conference on Learning Representations*. Vancouver, Canada.

[6] Tim Salimans, Jonathan Ho, Xi Chen, and Ilya Sutskever. 2017. Evolution Strategies as a Scalable Alternative to Reinforcement Learning. *arXiv preprint arXiv:1703.03864* (2017).

[7] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando De Freitas. 2015. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* 104, 1 (2015), 148–175.

[8] Felipe Petroski Such, Vashisht Madhavan, Edoardo Conti, Joel Lehman, Kenneth O Stanley, and Jeff Clune. 2017. Deep neuroevolution: genetic algorithms are a competitive alternative for training deep neural networks for reinforcement learning. *arXiv preprint arXiv:1712.06567* (2017).

[9] Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction.* MIT Press.

[10] Ziyu Wang, Victor Bapst, Nicolas Heess, Volodymyr Mnih, Rémi Munos, Koray Kavukcuoglu, and Nando de Freitas. 2017. Sample Efficient Actor-Critic with Experience Replay. In *Proceedings of the 5th. International Conference on Learning Representations*. Barcelona, Spain.

[11] Yang Yu, Hong Qian, and Yi-Qi Hu. 2016. Derivative-Free Optimization via Classification. In *Proceedings of the 30th. AAAI Conference on Artificial Intelligence.* 2286–2292.