

Proximal Policy Optimization

Spring, 2021

Outline

- 1 From ON-Policy to OFF-Policy
- 2 PPO/TRPO

Table of Contents

1 From ON-Policy to OFF-Policy

2 PPO/TRPO

Proximal Policy Optimization

① Policy Gradient → TRPO → ACKTR → PPO

- ① **TRPO**: Trust region policy optimization. Schulman, L., Moritz, Jordan, Abbeel. 2015
- ② **ACKTR**: Scalable trust-region method for deep reinforcement learning using Kronecker-factored approximation. Y. Wu, E. Mansimov, S. Liao, R. Grosse, and J. Ba. 2017
- ③ **PPO**: Proximal policy optimization algorithms. Schulman, Wolski, Dhariwal, Radford, Klimov. 2017

② Q-learning → DDPG → TD3 → SAC

- ① **DDPG**: Deterministic Policy Gradient Algorithms, Silver et al. 2014
- ② **TD3**: Addressing Function Approximation Error in Actor-Critic Methods, Fujimoto et al. 2018
- ③ **SAC**: Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor, Haarnoja et al. 2018

Proximal Policy Optimization

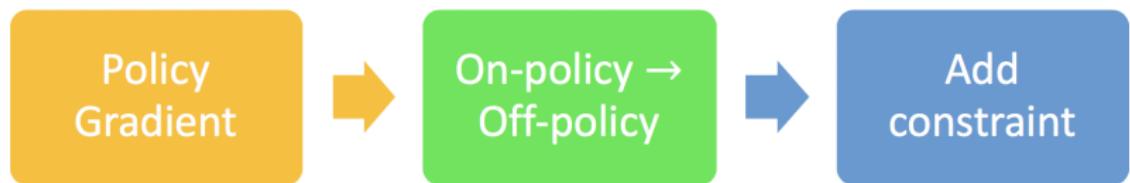
State-of-the-art RL methods are almost all policy-based

- ① **TRPO**: Schulman, L., Moritz, Jordan, Abbeel (2015). Trust region policy optimization
 - ① comment: Solid math proofs and guarantee, but hard to follow
- ② **ACKTR**: Y. Wu, E. Mansimov, S. Liao, R. Grosse, and J. Ba (2017). Scalable trust-region method for deep reinforcement learning using Kronecker-factored approximation.
 - ① comment: numeric optimization-based improvement, scalable to real-problems
- ③ **PPO**: Schulman, Wolski, Dhariwal, Radford, Klimov (2017). Proximal policy optimization algorithms
 - ① comment: Easy to read, elegant design of loss function, easy to implement, widely used

Proximal Policy Optimization

Proximal Policy Optimization (PPO)

default reinforcement learning algorithm at [OpenAI](#)



On-Policy vs. Off-Policy

From on-policy
to off-policy

Using the experience more than once

On-Policy vs. Off-Policy

On-policy v.s. Off-policy

- On-policy: The agent learned and the agent interacting with the environment is the same.
- Off-policy: The agent learned and the agent interacting with the environment is different.



阿光下棋



佐為下棋、阿光在旁邊看

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
-

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
 - Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.
-

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

Importance Sampling

$$E_{x \sim p} [f(x)]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

Importance Sampling

x^i is sampled from $p(x)$

$$E_{x \sim p} [f(x)] \approx \frac{1}{N} \sum_{i=1}^N f(x^i)$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

Importance Sampling

$$E_{x \sim p}[f(x)] \approx \frac{1}{N} \sum_{i=1}^N f(x^i)$$

x^i is sampled from $p(x)$

We only have x^i sampled from $q(x)$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

Importance Sampling

$$E_{x \sim p}[f(x)] \approx \frac{1}{N} \sum_{i=1}^N f(x^i)$$

x^i is sampled from $p(x)$

We only have x^i sampled from $q(x)$

$$= \int f(x)p(x)dx = \int f(x) \frac{p(x)}{q(x)} q(x)dx = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

Importance Sampling

$$E_{x \sim p}[f(x)] \approx \frac{1}{N} \sum_{i=1}^N f(x^i)$$

xⁱ is sampled from p(x)

We only have xⁱ sampled from q(x)

$$= \int f(x)p(x)dx = \int f(x) \frac{p(x)}{q(x)} q(x)dx = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

Importance weight

On-Policy vs. Off-Policy

Issue of Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\text{Var}_{x \sim p}[f(x)] \quad \text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\begin{aligned} \text{VAR}[X] \\ = E[X^2] - (E[X])^2 \end{aligned}$$

On-Policy vs. Off-Policy

Issue of Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\text{Var}_{x \sim p}[f(x)] \quad \text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$VAR[X]$$

$$= E[X^2] - (E[X])^2$$

$$\text{Var}_{x \sim p}[f(x)] = E_{x \sim p}[f(x)^2] - (E_{x \sim p}[f(x)])^2$$

On-Policy vs. Off-Policy

Issue of Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\text{Var}_{x \sim p}[f(x)] \quad \text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$VAR[X]$$

$$= E[X^2] - (E[X])^2$$

$$\text{Var}_{x \sim p}[f(x)] = E_{x \sim p}[f(x)^2] - (E_{x \sim p}[f(x)])^2$$

$$\text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}] = E_{x \sim q} \left[\left(f(x) \frac{p(x)}{q(x)} \right)^2 \right] - \left(E_{x \sim q} \left[f(x) \frac{p(x)}{q(x)} \right] \right)^2$$

On-Policy vs. Off-Policy

Issue of Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\text{Var}_{x \sim p}[f(x)] = \text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$VAR[X]$$

$$= E[X^2] - (E[X])^2$$

$$\text{Var}_{x \sim p}[f(x)] = E_{x \sim p}[f(x)^2] - (E_{x \sim p}[f(x)])^2$$

$$\text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}] = E_{x \sim q} \left[\left(f(x) \frac{p(x)}{q(x)} \right)^2 \right] - \left(E_{x \sim q} \left[f(x) \frac{p(x)}{q(x)} \right] \right)^2$$

$$= E_{x \sim p} \left[f(x)^2 \frac{p(x)}{q(x)} \right] - (E_{x \sim p}[f(x)])^2$$

On-Policy vs. Off-Policy

Issue of Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\text{Var}_{x \sim p}[f(x)] \quad \text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

$$\text{VAR}[X]$$

$$= E[X^2] - (E[X])^2$$

$$\text{Var}_{x \sim p}[f(x)] = E_{x \sim p}[f(x)^2] - (E_{x \sim p}[f(x)])^2$$

$$\text{Var}_{x \sim q}[f(x) \frac{p(x)}{q(x)}] = E_{x \sim q} \left[\left(f(x) \frac{p(x)}{q(x)} \right)^2 \right] - \left(E_{x \sim q} \left[f(x) \frac{p(x)}{q(x)} \right] \right)^2$$

$$= E_{x \sim p} \left[f(x)^2 \frac{p(x)}{q(x)} \right] - (E_{x \sim p}[f(x)])^2$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

$$\nabla \bar{R}_\theta = E_{\tau \sim p_{\theta'}(\tau)} \left[\frac{p_\theta(\tau)}{p_{\theta'}(\tau)} R(\tau) \nabla \log p_\theta(\tau) \right]$$

Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

$$\nabla \bar{R}_\theta = E_{\tau \sim p_\theta(\tau)} [R(\tau) \nabla \log p_\theta(\tau)]$$

- Use π_θ to collect data. When θ is updated, we have to sample training data again.
- Goal: Using the sample from $\pi_{\theta'}$ to train θ . θ' is fixed, so we can re-use the sample data.

$$\nabla \bar{R}_\theta = E_{\tau \sim p_{\theta'}(\tau)} \left[\frac{p_\theta(\tau)}{p_{\theta'}(\tau)} R(\tau) \nabla \log p_\theta(\tau) \right]$$

- Sample the data from θ' .
- Use the data to train θ many times.

Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}[f(x) \frac{p(x)}{q(x)}]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{P_\theta(s_t, a_t)}{P_{\theta'}(s_t, a_t)} A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

$$A^{\theta'}(s_t, a_t)$$

This term is from sampled data.

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{P_\theta(s_t, a_t)}{P_{\theta'}(s_t, a_t)} A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

$$A^{\theta'}(s_t, a_t)$$

This term is from sampled data.

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{P_\theta(s_t, a_t)}{P_{\theta'}(s_t, a_t)} A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} \frac{p_\theta(s_t)}{p_{\theta'}(s_t)} A^{\theta'}(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

$A^{\theta'}(s_t, a_t)$ This term is from sampled data.

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{P_\theta(s_t, a_t)}{P_{\theta'}(s_t, a_t)} A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} \frac{p_\theta(s_t)}{p_{\theta'}(s_t)} A^{\theta'}(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

$$A^{\theta'}(s_t, a_t)$$

This term is from sampled data.

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{P_\theta(s_t, a_t)}{P_{\theta'}(s_t, a_t)} A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} \frac{p_\theta(s_t)}{p_{\theta'}(s_t)} A^{\theta'}(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

$$J^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right]$$

On-Policy vs. Off-Policy

On-policy → Off-policy

Gradient for update

$$\nabla f(x) = f(x) \nabla \log f(x)$$

$$= E_{(s_t, a_t) \sim \pi_\theta} [A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n)]$$

$$A^{\theta'}(s_t, a_t)$$

This term is from sampled data.

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{P_\theta(s_t, a_t)}{P_{\theta'}(s_t, a_t)} A^\theta(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

$$= E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} \frac{p_\theta(s_t)}{p_{\theta'}(s_t)} A^{\theta'}(s_t, a_t) \nabla \log p_\theta(a_t^n | s_t^n) \right]$$

$$J^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right] \text{ When to stop?}$$

On-Policy vs. Off-Policy

Add Constraint

Table of Contents

- 1 From ON-Policy to OFF-Policy
- 2 PPO/TRPO

Proximal Policy Optimization

PPO / TRPO

Proximal Policy Optimization (PPO)

$$J_{PPO}^{\theta'}(\theta) = J^{\theta'}(\theta) - \beta KL(\theta, \theta')$$

$$\nabla f(x) = f(x)\nabla \log f(x)$$

$$J^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_{\theta}(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right]$$

Proximal Policy Optimization

PPO / TRPO

Proximal Policy Optimization (PPO)

$$J_{PPO}^{\theta'}(\theta) = J^{\theta'}(\theta) - \beta KL(\theta, \theta')$$

$$\nabla f(x) = f(x)\nabla \log f(x)$$

$$J^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_{\theta}(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right]$$

TRPO (Trust Region Policy Optimization)

$$J_{TRPO}^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_{\theta}(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right]$$

$$KL(\theta, \theta') < \delta$$

Proximal Policy Optimization

PPO / TRPO

θ cannot be very different from θ'

Constraint on behavior not parameters



Proximal Policy Optimization (PPO)

$$\nabla f(x) = f(x)\nabla \log f(x)$$

$$J_{PPO}^{\theta'}(\theta) = J^{\theta'}(\theta) - \beta KL(\theta, \theta')$$

$$J^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right]$$

TRPO (Trust Region Policy Optimization)

$$J_{TRPO}^{\theta'}(\theta) = E_{(s_t, a_t) \sim \pi_{\theta'}} \left[\frac{p_\theta(a_t | s_t)}{p_{\theta'}(a_t | s_t)} A^{\theta'}(s_t, a_t) \right]$$

$$KL(\theta, \theta') < \delta$$

Proximal Policy Optimization

PPO algorithm

- Initial policy parameters θ^0
- In each iteration
 - Using θ^k to interact with the environment to collect $\{s_t, a_t\}$ and compute advantage $A^{\theta^k}(s_t, a_t)$
 - Find θ optimizing $J_{PPO}(\theta)$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t)$$

$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

Update parameters
several times

Proximal Policy Optimization

PPO algorithm

- Initial policy parameters θ^0
- In each iteration
 - Using θ^k to interact with the environment to collect $\{s_t, a_t\}$ and compute advantage $A^{\theta^k}(s_t, a_t)$
 - Find θ optimizing $J_{PPO}(\theta)$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t)$$

$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

Update parameters
several times

- If $KL(\theta, \theta^k) > KL_{max}$, increase β
- If $KL(\theta, \theta^k) < KL_{min}$, decrease β

Proximal Policy Optimization

PPO algorithm

- Initial policy parameters θ^0
- In each iteration
 - Using θ^k to interact with the environment to collect $\{s_t, a_t\}$ and compute advantage $A^{\theta^k}(s_t, a_t)$
 - Find θ optimizing $J_{PPO}(\theta)$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t)$$

$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

Update parameters
several times

- If $KL(\theta, \theta^k) > KL_{max}$, increase β
- If $KL(\theta, \theta^k) < KL_{min}$, decrease β

Adaptive
KL Penalty

Proximal Policy Optimization

PPO algorithm

$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t)$$

PPO2 algorithm

$$J_{PPO2}^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \min \left(\frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t), \right.$$

$$\left. \text{clip} \left(\frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)}, 1 - \varepsilon, 1 + \varepsilon \right) A^{\theta^k}(s_t, a_t) \right)$$

Proximal Policy Optimization

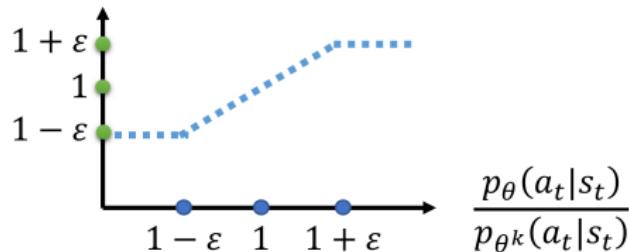
PPO algorithm

$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t)$$

PPO2 algorithm

$$J_{PPO2}^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \text{clip}\left(\frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)}, 1 - \varepsilon, 1 + \varepsilon\right) A^{\theta^k}(s_t, a_t)$$



Proximal Policy Optimization

PPO algorithm

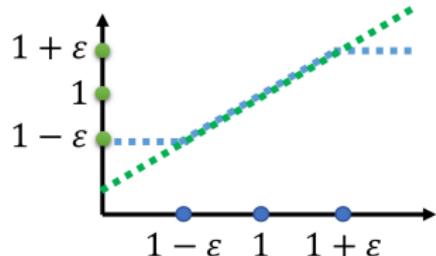
$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t)$$

PPO2 algorithm

$$J_{PPO2}^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \min \left(\frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t), \right.$$

$$\left. \text{clip} \left(\frac{p_\theta(a_t | s_t)}{p_{\theta^k}(a_t | s_t)}, 1 - \varepsilon, 1 + \varepsilon \right) A^{\theta^k}(s_t, a_t) \right)$$



Proximal Policy Optimization

PPO algorithm

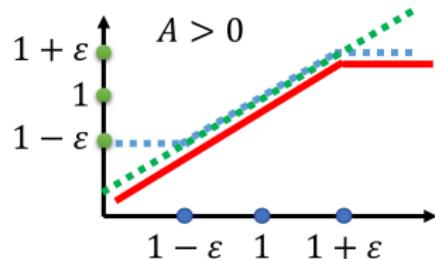
$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t|s_t)}{p_{\theta^k}(a_t|s_t)} A^{\theta^k}(s_t, a_t)$$

PPO2 algorithm

$$J_{PPO2}^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \min \left(\frac{p_\theta(a_t|s_t)}{p_{\theta^k}(a_t|s_t)} A^{\theta^k}(s_t, a_t), \right.$$

$$\left. \text{clip} \left(\frac{p_\theta(a_t|s_t)}{p_{\theta^k}(a_t|s_t)}, 1 - \varepsilon, 1 + \varepsilon \right) A^{\theta^k}(s_t, a_t) \right)$$



Proximal Policy Optimization

PPO algorithm

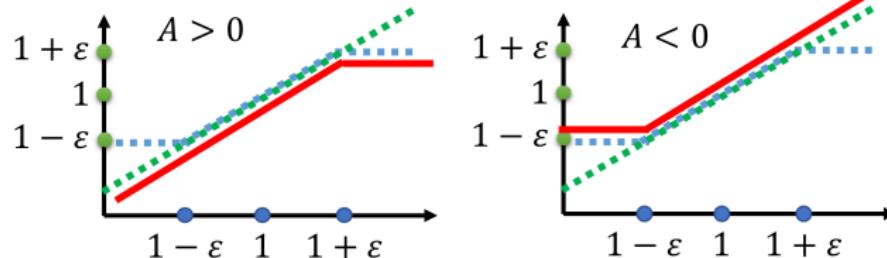
$$J_{PPO}^{\theta^k}(\theta) = J^{\theta^k}(\theta) - \beta KL(\theta, \theta^k)$$

$$J^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \frac{p_\theta(a_t|s_t)}{p_{\theta^k}(a_t|s_t)} A^{\theta^k}(s_t, a_t)$$

PPO2 algorithm

$$J_{PPO2}^{\theta^k}(\theta) \approx \sum_{(s_t, a_t)} \min \left(\frac{p_\theta(a_t|s_t)}{p_{\theta^k}(a_t|s_t)} A^{\theta^k}(s_t, a_t), \right.$$

$$\left. \text{clip} \left(\frac{p_\theta(a_t|s_t)}{p_{\theta^k}(a_t|s_t)}, 1 - \varepsilon, 1 + \varepsilon \right) A^{\theta^k}(s_t, a_t) \right)$$



Proximal Policy Optimization

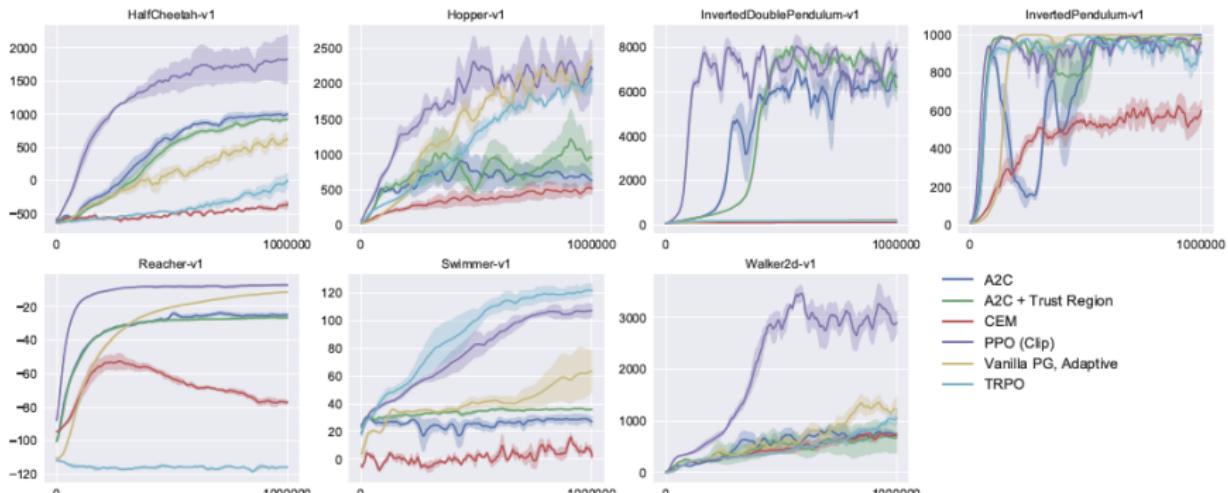


Figure 3: Comparison of several algorithms on several MuJoCo environments, training for one million timesteps.