Proximal Policy Optimization

Ruifan Yu

(ruifan.yu@uwaterloo.ca)

CS 885

June 20

UNIVERSITY OF WATERLOO

Proximal Policy Optimization (OpenAI)

"PPO has become the default reinforcement learning algorithm at OpenAI because of its ease of use and good performance"

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

https://arxiv.org/pdf/1707.06347
https://blog.openai.com/openai-baselines-ppo/

Policy Gradient (REINFORCE)

REINFORCE(s_0, π_θ)

Initialize π_θ to anything

Loop forever (for each episode)

Generate episode $\mathrm{s}_0, \mathrm{a}_0, \mathrm{r}_0, \mathrm{s}_1, \mathrm{a}_1, r_1, \dots, s_T, a_T, r_T$ with π_θ

Loop for each step of the episode $n = 0, 1, \dots, T$

$G_n \leftarrow \sum_{t=0}^{T-n} \gamma^t \, r_{n+t}$

Update policy: $\theta \leftarrow \theta + \alpha \, \gamma^n G_n \nabla \log \pi_\theta(a_n|s_n)$

Return π_θ

In practice, update on each batch(trajectory)

* Use the same notation in the paper

$$\max_\theta J(\pi_\theta) \doteq \mathop{\mathrm{E}}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^{\infty} \gamma^t r_t \right]$$

$$g = \nabla_\theta J(\pi_\theta) = \mathop{\mathrm{E}}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^{\infty} \gamma^t \nabla_\theta \log \pi_\theta(a_t|s_t) A^{\pi_\theta}(s_t, a_t) \right]$$

Problem?

- **Unstable update**
  - Step size is very important:
    - If step size is too large:
      - Large step → bad policy
      - Next batch is generated from current bad policy → collect bad samples
      - Bad samples → worse policy

      (compare to supervised learning: the correct label and data in the following batches may correct it)
    - If step size is too small: the learning process is slow

- **Data Inefficiency**
  - On-policy method: for each new policy, we need to generate a completely new trajectory
  - The data is thrown out after just one gradient update
  - As complex neural networks need many updates, this makes the training process very slow

Importance Sampling

Estimate one distribution by sampling from another distribution

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

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

$$= \int f(x)\frac{p(x)}{q(x)}q(x)dx$$

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

$$\approx \frac{1}{N}\sum_{i=1,x^i\in q}^{N} f(x^i)\frac{p(x^i)}{q(x^i)}$$

Data Inefficiency

Evaluate the gradient of current policy

Use previous samples?

$$\mathop{\mathrm{E}}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \nabla_\theta \log \pi_\theta(a_t|s_t) A^{\pi_\theta}(s_t, a_t)\right]$$

Like replay buffer in DQN

Data Inefficiency

Make it efficient

Avoid sampling from current policy

Can we estimate an expectation of one distribution without taking samples from it?

Importance Sampling in Policy Gradient

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

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

$$= E_{(s_t,\, a_t)\sim\pi_{\theta_{old}}}[\frac{\pi_\theta(s_t, a_t)}{\pi_{\theta_{old}}(s_t, a_t)} \nabla \log \pi_\theta(a_t|s_t)\, A(s_t, a_t)]$$

$$J(\theta) = E_{(s_t,\, a_t)\sim\pi_{\theta_{old}}}[\frac{\pi_\theta(s_t, a_t)}{\pi_{\theta_{old}}(s_t, a_t)} A(s_t, a_t)]$$

→ Surrogate objective function

Importance Sampling

Problem? No free lunch!

Two expectations are same, but we are using sampling method to estimate them
→ variance is also important

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

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

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

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

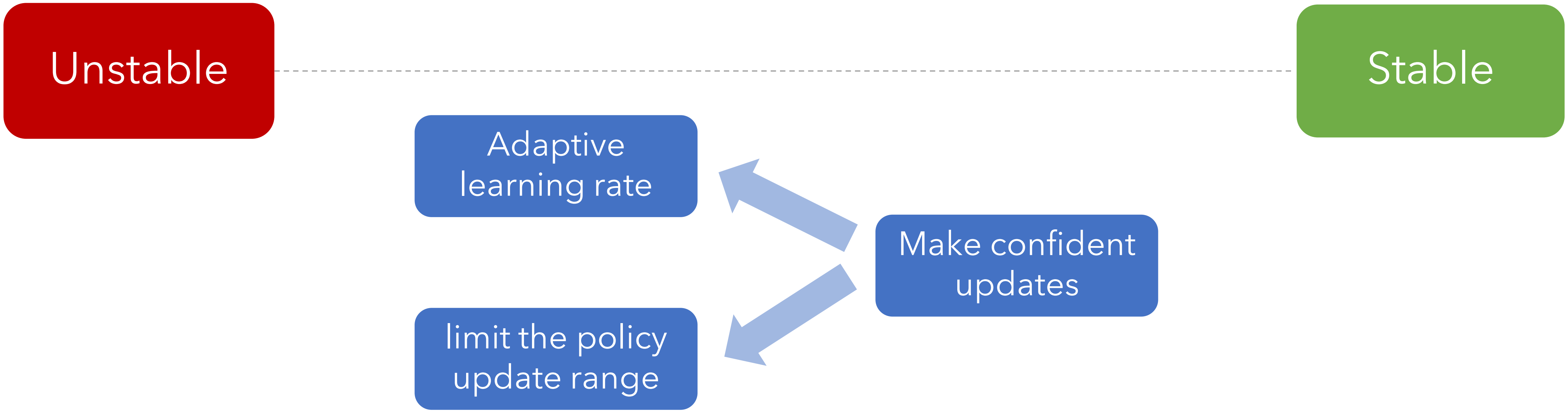
$$Var_{x\sim q}[f(x)\frac{p(x)}{q(x)}]$$

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

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

Price (Tradeoff): we may need to sample more data, if $\frac{p(x)}{q(x)}$ is far away from 1

Unstable Update



Can we measure the distance between two distributions?

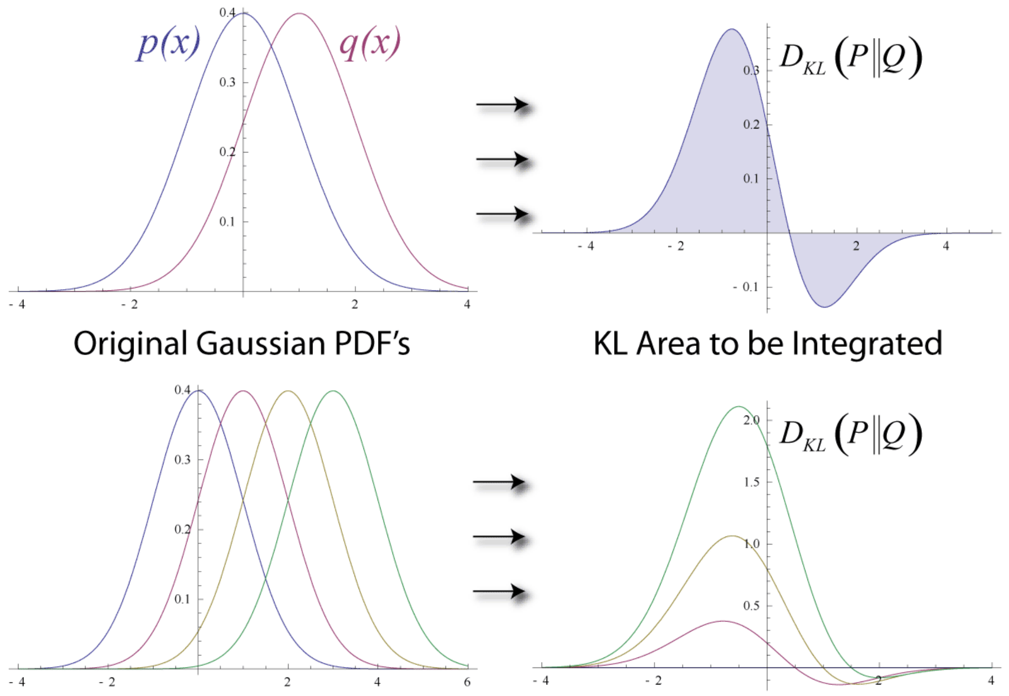
KL Divergence

Measure the distance of two distributions

$$D_{KL}(P||Q) = \sum_x P(x) log \frac{P(x)}{Q(x)}$$

KL divergence of two policies

$$D_{KL}(\pi_1||\pi_2)[s] = \sum_{a\in A} \pi_1(a|s) log \frac{\pi_1(a|s)}{\pi_2(a|s)}$$



* image: Kullback-Leibler divergence (Wikipedia)
https://en.wikipedia.org/wiki/Kullback-Leibler_divergence

Trust Region Policy Optimization (TRPO)

$$\underset{\theta}{\text{maximize}} \quad \hat{\mathbb{E}}_t \left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} \hat{A}_t \right]$$

$$\text{subject to} \quad \hat{\mathbb{E}}_t [\text{KL}[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)]] \leq \delta$$

Common trick in optimization: Lagrangian Dual

$$\underset{\theta}{\text{maximize}} \, \hat{\mathbb{E}}_t \left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} \hat{A}_t - \beta \, \text{KL}[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)] \right]$$

TRPO uses a hard constraint rather than a penalty because it is hard to choose a single value of β that performs well across different problems—or even within a single problem, where the characteristics change over the course of learning

Proximal Policy Optimization (PPO)

TRPO use conjugate gradient decent to handle the constraint

Hessian Matrix → expensive both in computation and space

Idea:
The constraint helps in the training process. However, maybe the constraint is not a strict constraint:
Does it matter if we only break the constraint just a few times?

What if we treat it as a "soft" constraint? Add proximal value to objective function?

PPO with Adaptive KL Penalty

$$L^{KLPEN}(\theta) = \hat{\mathbb{E}}_t \left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} \hat{A}_t - \beta \, \text{KL}[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)] \right]$$

Hard to pick β value → use adaptive β

Compute $d = \hat{\mathbb{E}}_t[\text{KL}[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)]]$

- If $d < d_{\text{targ}}/1.5$, $\beta \leftarrow \beta/2$
- If $d > d_{\text{targ}} \times 1.5$, $\beta \leftarrow \beta \times 2$

Still need to set up a KL divergence target value …

PPO with Adaptive KL Penalty

Algorithm 4 PPO with Adaptive KL Penalty

Input: initial policy parameters θ_0, initial KL penalty β_0, target KL-divergence δ
for $k = 0, 1, 2, ...$ **do**
  Collect set of partial trajectories $\mathcal{D}_k$ on policy $\pi_k = \pi(\theta_k)$
  Estimate advantages $\hat{A}_t^{\pi_k}$ using any advantage estimation algorithm
  Compute policy update

$$\theta_{k+1} = \arg\max_{\theta} \mathcal{L}_{\theta_k}(\theta) - \beta_k \bar{D}_{KL}(\theta || \theta_k)$$

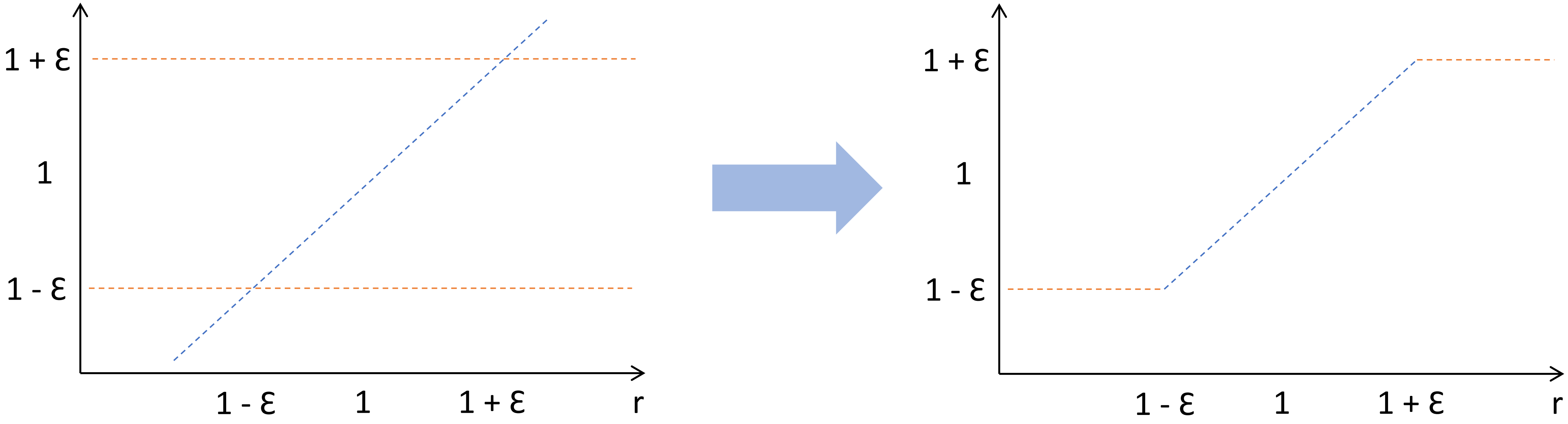
  by taking K steps of minibatch SGD (via Adam)
  if $\bar{D}_{KL}(\theta_{k+1} || \theta_k) \geq 1.5\delta$ **then**
    $\beta_{k+1} = 2\beta_k$
  else if $\bar{D}_{KL}(\theta_{k+1} || \theta_k) \leq \delta / 1.5$ **then**
    $\beta_{k+1} = \beta_k / 2$
  end if
end for

* CS294 Fall 2017, Lecture 13
http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_13_advanced_pg.pdf

PPO with Clipped Objective

$$\underset{\theta}{\text{maximize}} \quad \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}\hat{A}_t\right] \qquad r_t(\theta) = \frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}$$

Fluctuation happens when r changes too quickly → limit r within a range?



$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t\left[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)\right]$$

PPO with Clipped Objective

Algorithm 5 PPO with Clipped Objective

Input: initial policy parameters θ_0, clipping threshold ϵ

for $k = 0, 1, 2, \ldots$ **do**

Collect set of partial trajectories $\mathcal{D}_k$ on policy $\pi_k = \pi(\theta_k)$

Estimate advantages $\hat{A}_t^{\pi_k}$ using any advantage estimation algorithm

Compute policy update

$$\theta_{k+1} = \arg\max_{\theta} \mathcal{L}_{\theta_k}^{CLIP}(\theta)$$

by taking K steps of minibatch SGD (via Adam), where

$$\mathcal{L}_{\theta_k}^{CLIP}(\theta) = \underset{\tau \sim \pi_k}{\mathrm{E}} \left[\sum_{t=0}^{T} \left[\min(r_t(\theta)\hat{A}_t^{\pi_k}, \text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right) \hat{A}_t^{\pi_k}) \right] \right]$$

end for

* CS294 Fall 2017, Lecture 13
http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_13_advanced_pg.pdf

PPO in practice

$$L_t^{CLIP+VF+S}(\theta) = \hat{\mathbb{E}}_t\left[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S[\pi_\theta](s_t)\right]$$

Surrogate objective function

a squared-error loss for "critic"

$$(V_\theta(s_t) - V_t^{\mathrm{targ}})^2$$

entropy bonus to ensure sufficient exploration

encourage "diversity"

* c1, c2: empirical values, in the paper, c1=1, c2=0.01

Performance

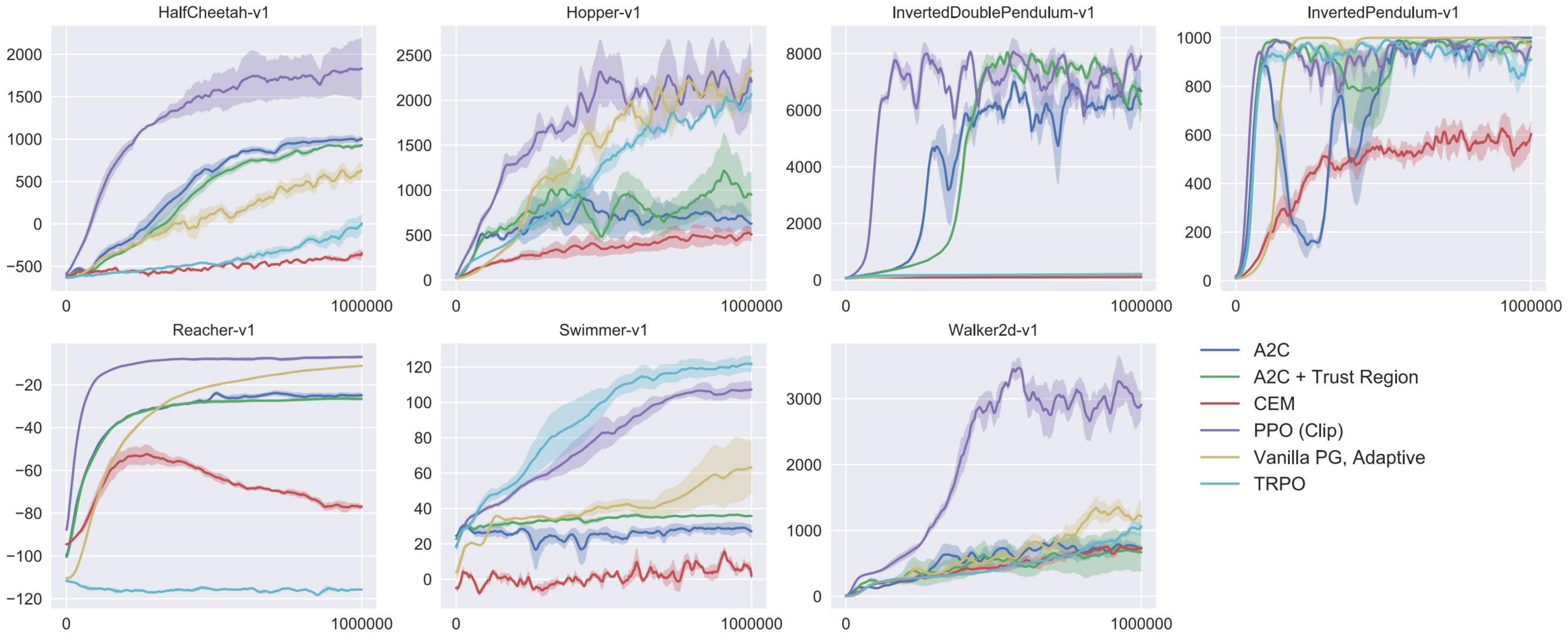
No clipping or penalty:	$L_t(\theta) = r_t(\theta)\hat{A}_t$
Clipping:	$L_t(\theta) = \min(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta)), 1-\epsilon, 1+\epsilon)\hat{A}_t$
KL penalty (fixed or adaptive)	$L_t(\theta) = r_t(\theta)\hat{A}_t - \beta\,\mathrm{KL}[\pi_{\theta_{\mathrm{old}}}, \pi_\theta]$

Results from continuous control benchmark. Average normalized scores (over 21 runs of the algorithm, on 7 environments)

algorithm	avg. normalized score
No clipping or penalty	-0.39
Clipping, $\epsilon = 0.1$	0.76
Clipping, $\epsilon = 0.2$	**0.82**
Clipping, $\epsilon = 0.3$	0.70
Adaptive KL $d_{\mathrm{targ}} = 0.003$	0.68
Adaptive KL $d_{\mathrm{targ}} = 0.01$	0.74
Adaptive KL $d_{\mathrm{targ}} = 0.03$	0.71
Fixed KL, $\beta = 0.3$	0.62
Fixed KL, $\beta = 1.$	0.71
Fixed KL, $\beta = 3.$	0.72
Fixed KL, $\beta = 10.$	0.69

Performance

Results in MuJoCo environments, training for one million timesteps



Related Works

[1] *Emergence of Locomotion Behaviours in Rich Environments*

Distributed PPO

Interesting fact: this paper is published before PPO paper

DeepMind got this idea from OpenAI's talking in NIPS 2016

[2] *An Adaptive Clipping Approach for Proximal Policy Optimization*

PPO-λ

Change the clipping range adaptively

[1] https://arxiv.org/abs/1707.02286
[2] https://arxiv.org/abs/1804.06461

END

Thank you