Nightmares of Policy Optimization

Sanjiban Choudhury

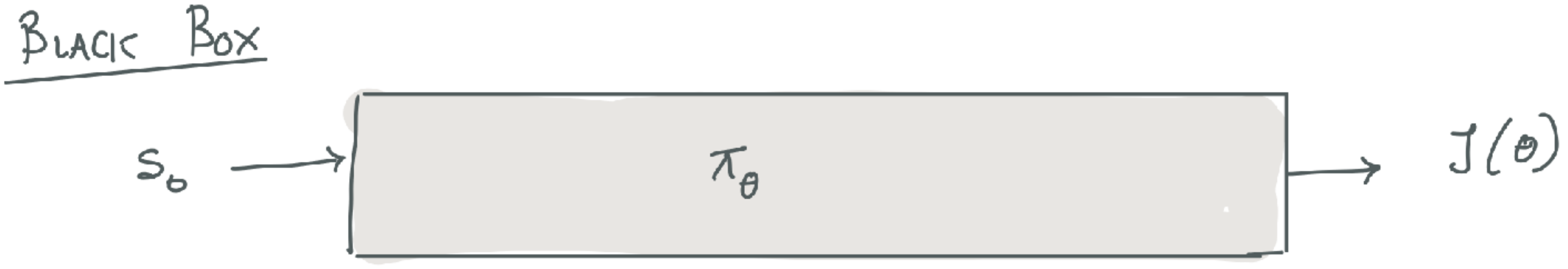
Cornell Bowers CIS
Computer Science

Switch from costs to rewards

All optimal control / planning literature written as costs

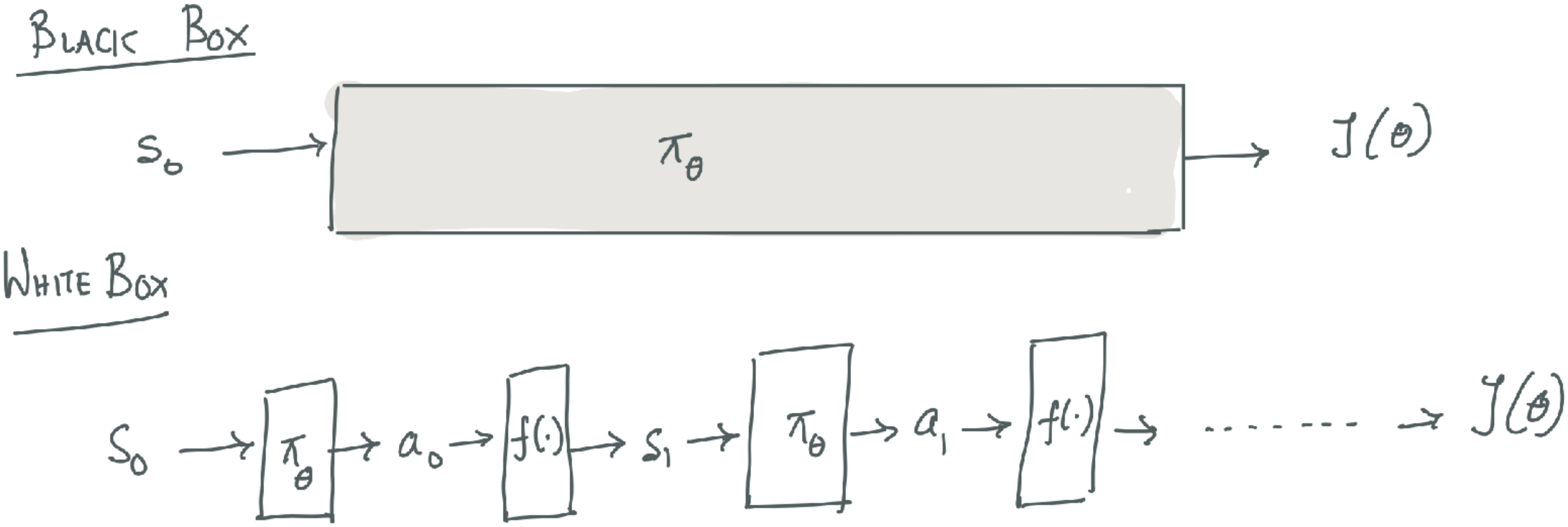
All RL literature written as rewards

We assumed black-box policies ...

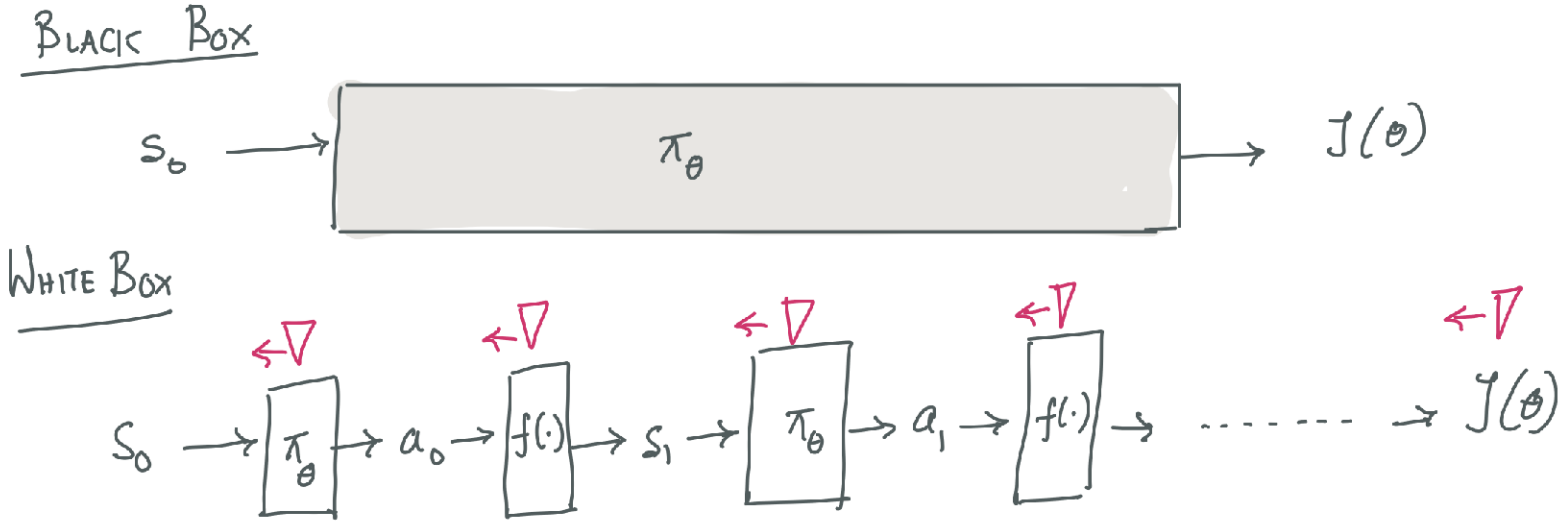


Have we redacted too much?
Form No. I
REPORT MADE AT
TITLE
Mr. [redacted]
SYNOPSIS OF FACTS:
Subject
employment as a worke
chief [redacted] for t
[redacted]
DETAILS:
BACKGROUND
Birth
SUBJECT: [redacted]
I. [redacted]
of [redacted] is being s[redacted]
[redacted] and biological study at [redacted] in the areas of [redacted] work
until [redacted], [redacted]. The pr[redacted]
2. This [redacted]
into the [redacted] areas, [redacted] l effects of [redacted].
continue to be [redacted], [redacted] load
[redacted] ors will
3. The [redacted] and [redacted]
[redacted] The [redacted] estimated to in the
[redacted] quired, under [redacted]

Black-box vs White-box vs Gray-box



Black-box vs White-box vs Gray-box



How can we take gradients if we don't know the dynamics?

The Likelihood Ratio Trick!

REINFORCE

Algorithm 20: The REINFORCE algorithm.

Start with an arbitrary initial policy π_θ

while *not converged* **do**

Run simulator with π_θ to collect $\{\xi^{(i)}\}_{i=1}^{N}$

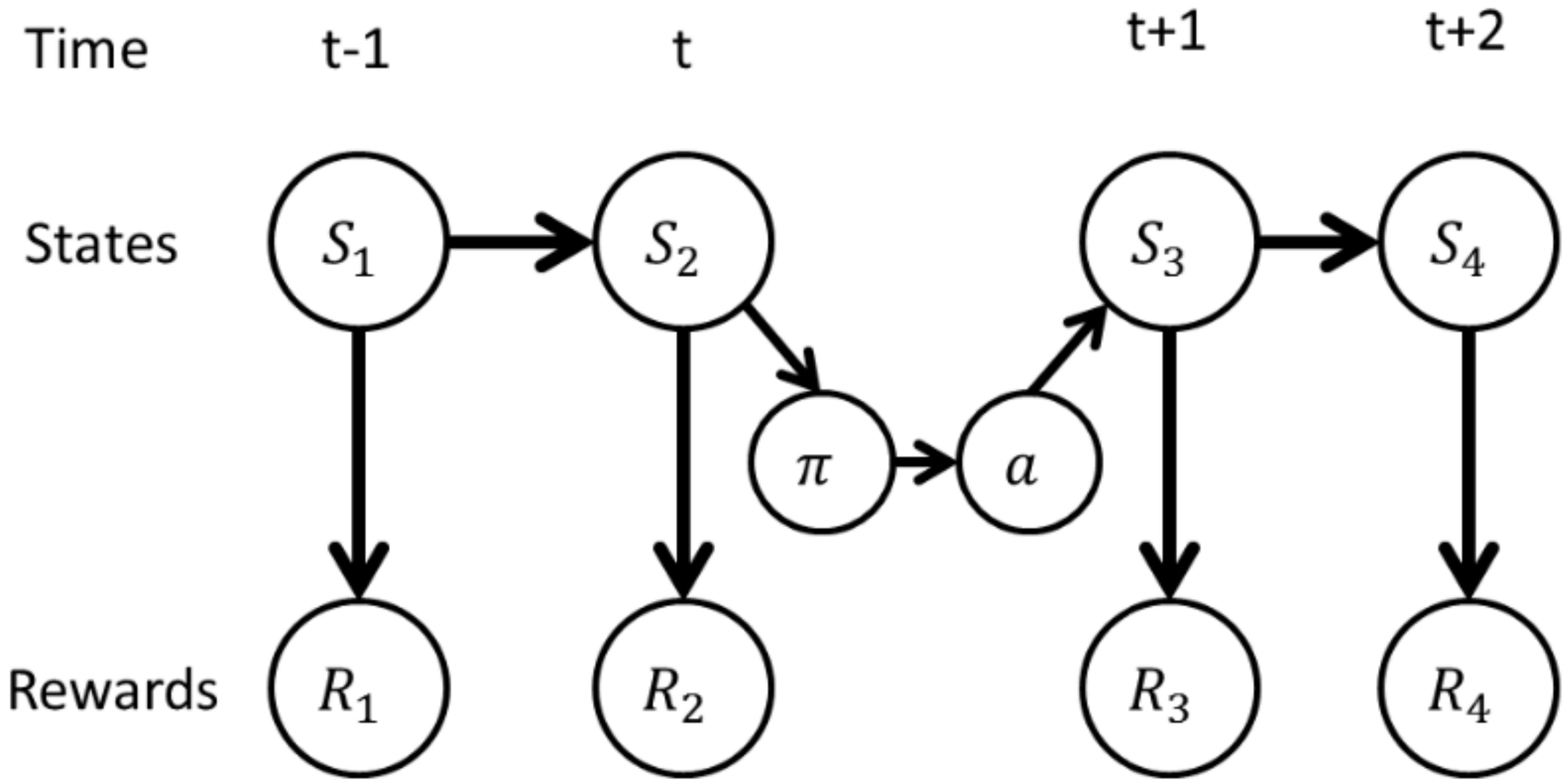
Compute estimated gradient

$$\widetilde{\nabla}_\theta J = \frac{1}{N} \sum_{i=1}^{N} \left[\left(\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta \left(a_t^{(i)} | s_t^{(i)} \right) \right) R(\xi^{(i)}) \right]$$

Update parameters $\theta \leftarrow \theta + \alpha \, \widetilde{\nabla}_\theta J$

return π_θ

Causality: Can actions affect the past?



The Policy Gradient Theorem

$$\begin{aligned}\nabla_\theta J &= E_{p(\xi|\theta)}\left[\sum_{t=0}^{T-1}\left(\nabla_\theta \log \pi_\theta(a_t|s_t)\left(\sum_{t'=0}^{t-1} r(s_{t'},a_{t'}) + \sum_{t'=t}^{T-1} r(s_{t'},a_{t'})\right)\right)\right] \\ &= E_{p(\xi|\theta)}\left[\sum_{t=0}^{T-1}\left(\nabla_\theta \log \pi_\theta(a_t|s_t)\sum_{t'=t}^{T-1} r(s_{t'},a_{t'})\right)\right],\end{aligned}$$

$$\nabla_\theta J = E_{p(\xi|\theta)}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t)\, Q^{\pi_\theta}(s_t,a_t)\right]$$

$$\boxed{\nabla_\theta J = E_{s\sim d^{\pi_\theta}(s),\, a\sim\pi_\theta(a|s)}\left[\nabla_\theta \log \pi_\theta(a|s) Q^{\pi_\theta}(s,a)\right]}$$

Life is good!

This solves everything ...

The Three Nightmares of Policy Optimization

Nightmare 1:

Variance

When Q values for all rollouts in a batch are high?

$$\boxed{\nabla_\theta J = E_{s\sim d^{\pi_\theta}(s), a\sim\pi_\theta(a|s)} \left[\nabla_\theta \log \pi_\theta(a|s) Q^{\pi_\theta}(s,a)\right]}$$

Recall that one of the reasons for the high variance is that the algorithm does not know how well the trajectories perform compared to other trajectories. Therefore, by introducing a baseline for the total reward (or reward to go), we can update the policy based on how well the policy performs compared to a baseline

Solution: Subtract a baseline!

$$\nabla_\theta J = E_{d^{\pi_\theta}(s)} E_{\pi_\theta(a|s)} \left[\nabla_\theta \log(\pi_\theta(a|s) \left(Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s) \right) \right].$$

We can prove that this does not change the gradient

$$\nabla_\theta J = E_{d^{\pi_\theta}(s)} E_{\pi_\theta(a|s)} \left[\nabla_\theta \log(\pi_\theta(a|s) A^{\pi_\theta}(s,a) \right]$$

But turns Q values into advantage (which is lower variance)

Justify the move to advantage using PDL!

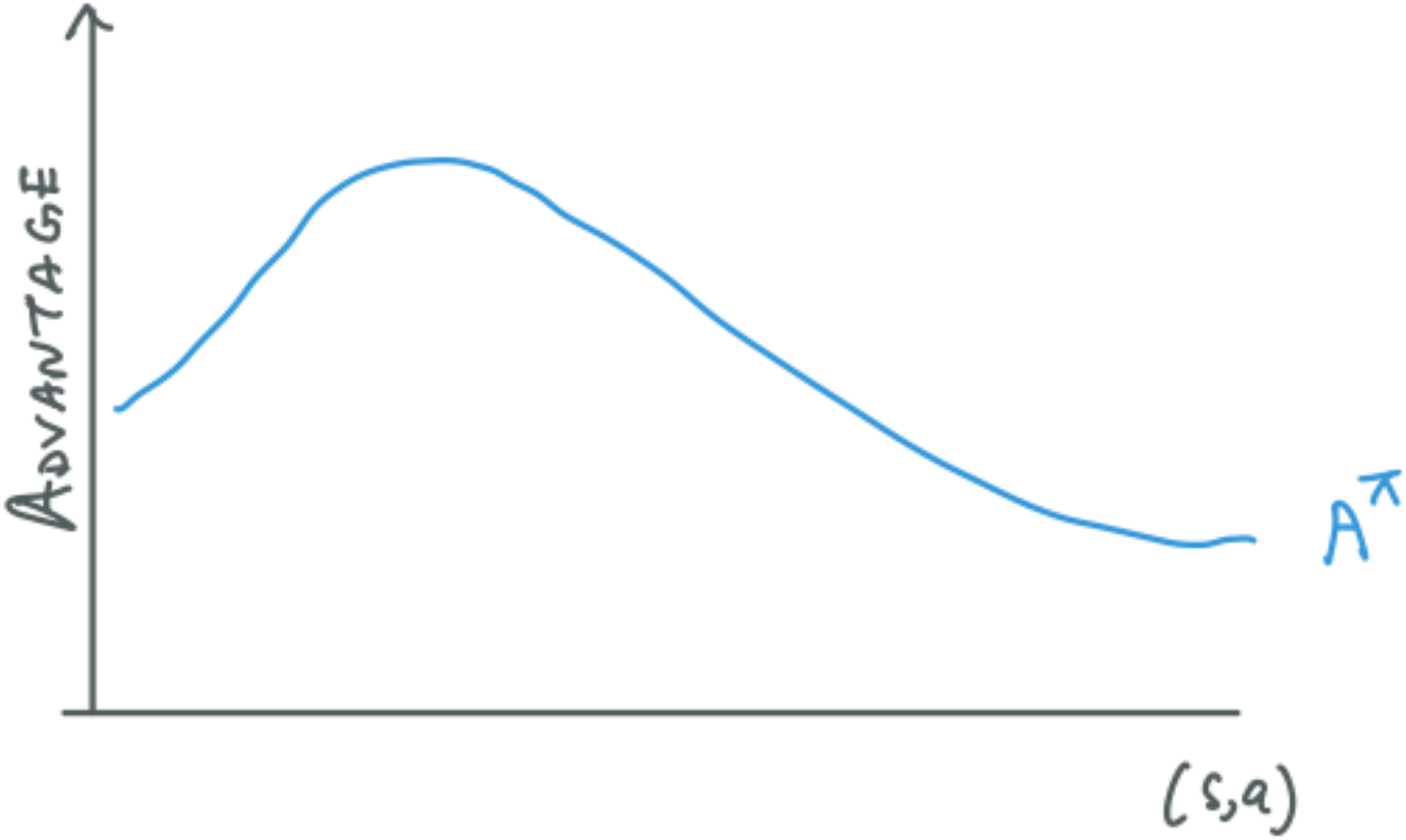
Nightmare 2:

Distribution Shift

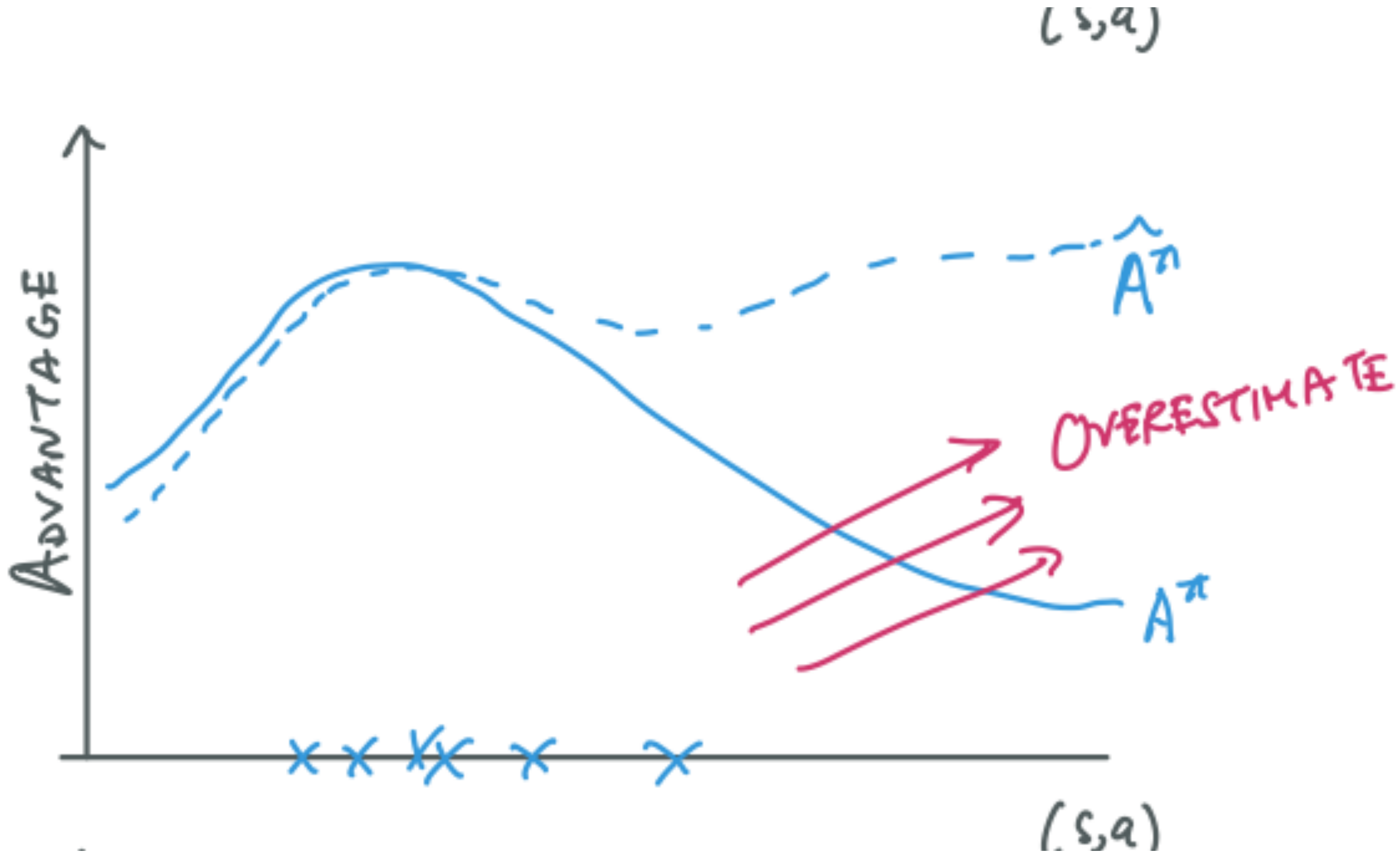
What happens if your step-size is large?

$$\nabla_\theta J = E_{d^{\pi_\theta}(s)} E_{\pi_\theta(a|s)} \left[\nabla_\theta \log(\pi_\theta(a|s) A^{\pi_\theta}(s,a)\right]$$

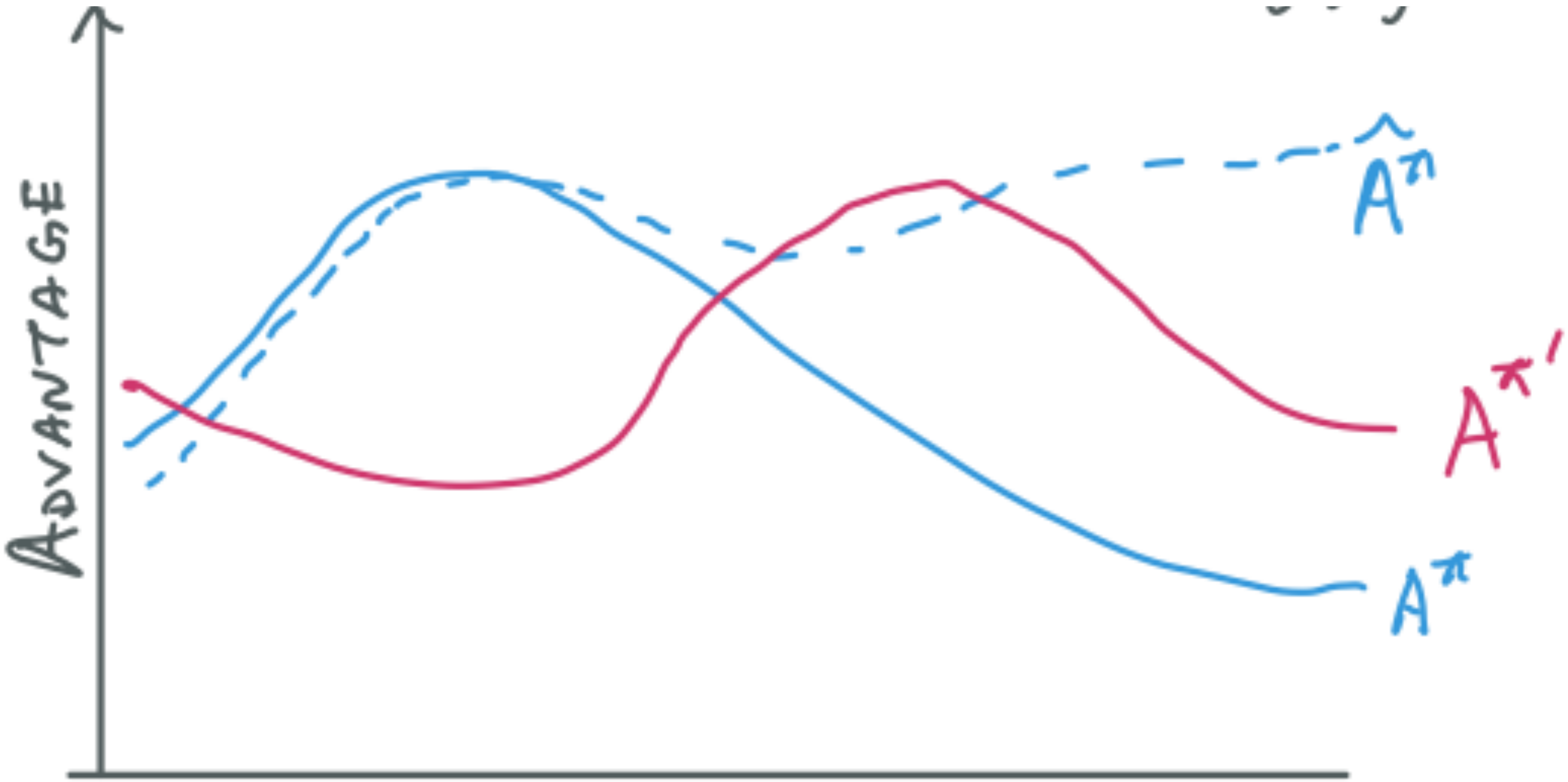
The problem of distribution shift



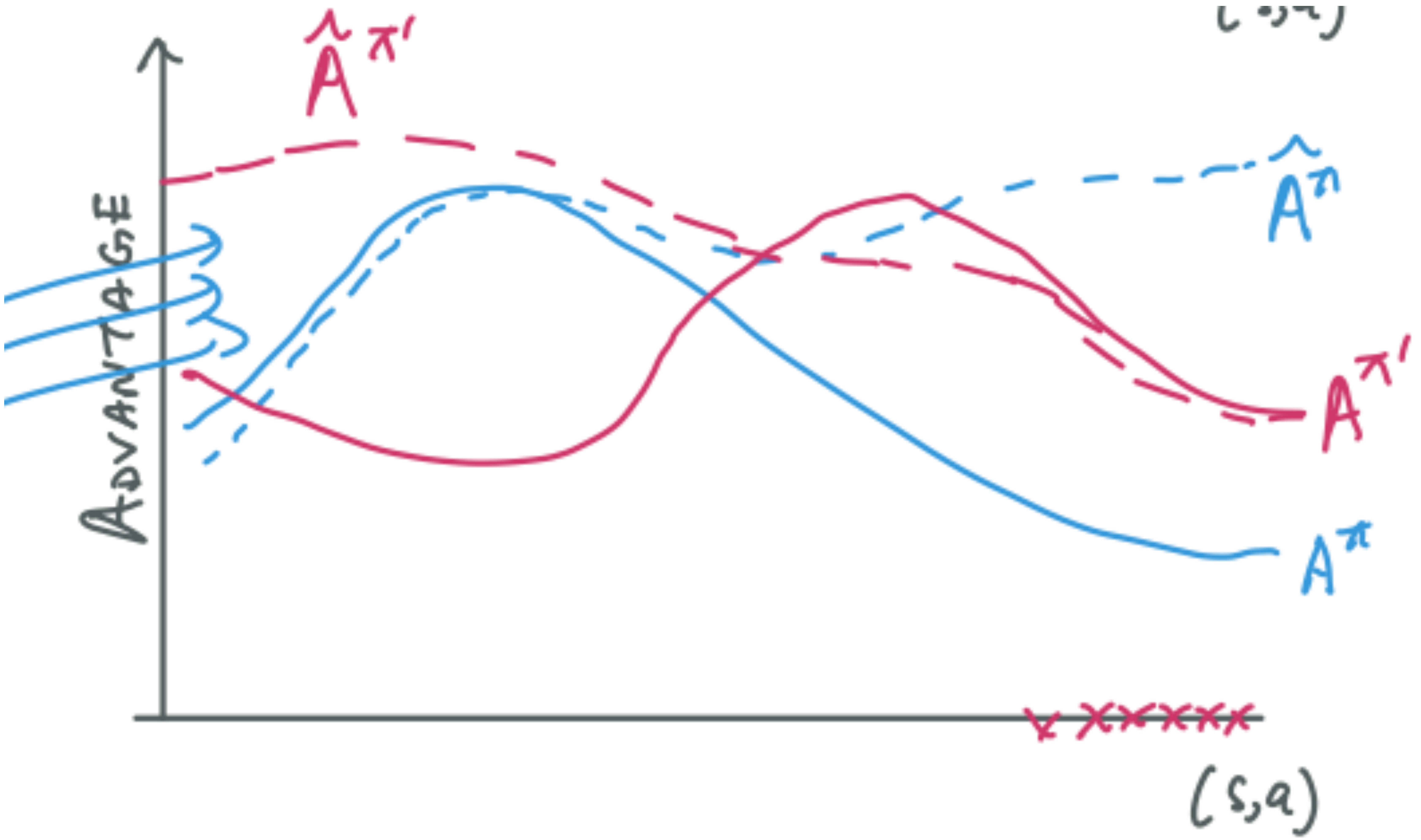
The problem of distribution shift



The problem of distribution shift



The problem of distribution shift



How does distribution shift manifest?

The true performance difference

$$J(\pi') - J(\pi) = \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d^t_{\pi'}} A^\pi(s, \pi'(s))$$

(New) (Old)

What our estimator currently approximates

$$\sum_{t=0}^{T-1} \mathbb{E}_{\boxed{s \sim d^t_{\pi}}} A^\pi(s, \pi'(s))$$

Slowly change policies

Keep d^t_{π} close to $d^t_{\pi'}$

Idea: Update distributions slowly

Does this simply mean do gradient descent with a small step size?

Does gradient descent keep distribution change small?

Gradient Descent is simply Steepest Descent with L2 norm

$$\max_{\Delta\theta} J(\theta + \Delta\theta) \qquad s.t. \qquad \|\Delta\theta\| \leq \epsilon$$

Does this ensure $d_{\pi_{\theta+\Delta\theta}}$ and d_{π_θ} are close??

What if we change norms?

Gradient Descent is simply Steepest Descent with L2 norm

$$\max_{\Delta\theta} J(\theta + \Delta\theta) \qquad s.t. \qquad \|\Delta\theta\| \leq \epsilon \qquad \longrightarrow \qquad \Delta\theta = \nabla_\theta J(\theta)$$

What would update look like for another norm?

$$\max_{\Delta\theta} J(\theta + \Delta\theta) \qquad s.t. \qquad \Delta\theta^\top G(\theta)\Delta\theta \leq \epsilon \qquad \longrightarrow \qquad \Delta\theta = \frac{1}{2\lambda} G^{-1}(\theta) \nabla_\theta J(\theta)$$

What's a good norm for distributions?

What is a good norm for distributions?

$$\max_{\Delta\theta} J(\theta + \Delta\theta)$$

$$\text{s.t. } KL(P(\theta + \Delta\theta) \,||\, P(\theta)) \leq \epsilon$$

What is a good norm for distributions?

$$\max_{\Delta\theta} J(\theta + \Delta\theta)$$

$$\cancel{\text{s.t. } KL(P(\theta + \Delta\theta)\,||\,P(\theta)) \leq \epsilon}$$

$$\text{s.t. } \Delta\theta^T G(\theta) \Delta\theta \leq \epsilon$$

Fischer Information Matrix

$$G(\theta) = E_{p_\theta}\left[\nabla_\theta \log(p_\theta)\, \nabla_\theta \log(p_\theta)^\top\right]$$

"Natural" Gradient Descent

Start with an arbitrary initial policy π_θ

while *not converged* **do**

Run simulator with π_θ to collect $\{\xi^{(i)}\}_{i=1}^N$

Compute estimated gradient

$$\tilde{\nabla}_\theta J = \frac{1}{N}\sum_{i=1}^{N}\left[\left(\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta\left(a_t^{(i)}|s_t^{(i)}\right)\right)R(\xi^{(i)})\right]$$

$$\boxed{\tilde{G}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left[\nabla_\theta \log \pi_\theta(a_i|s_i)\nabla_\theta \log \pi_\theta(a_i|s_i)^\top\right]}$$

Update parameters $\theta \leftarrow \theta + \alpha \boxed{\tilde{G}^{-1}(\theta)} \tilde{\nabla}_\theta J$.

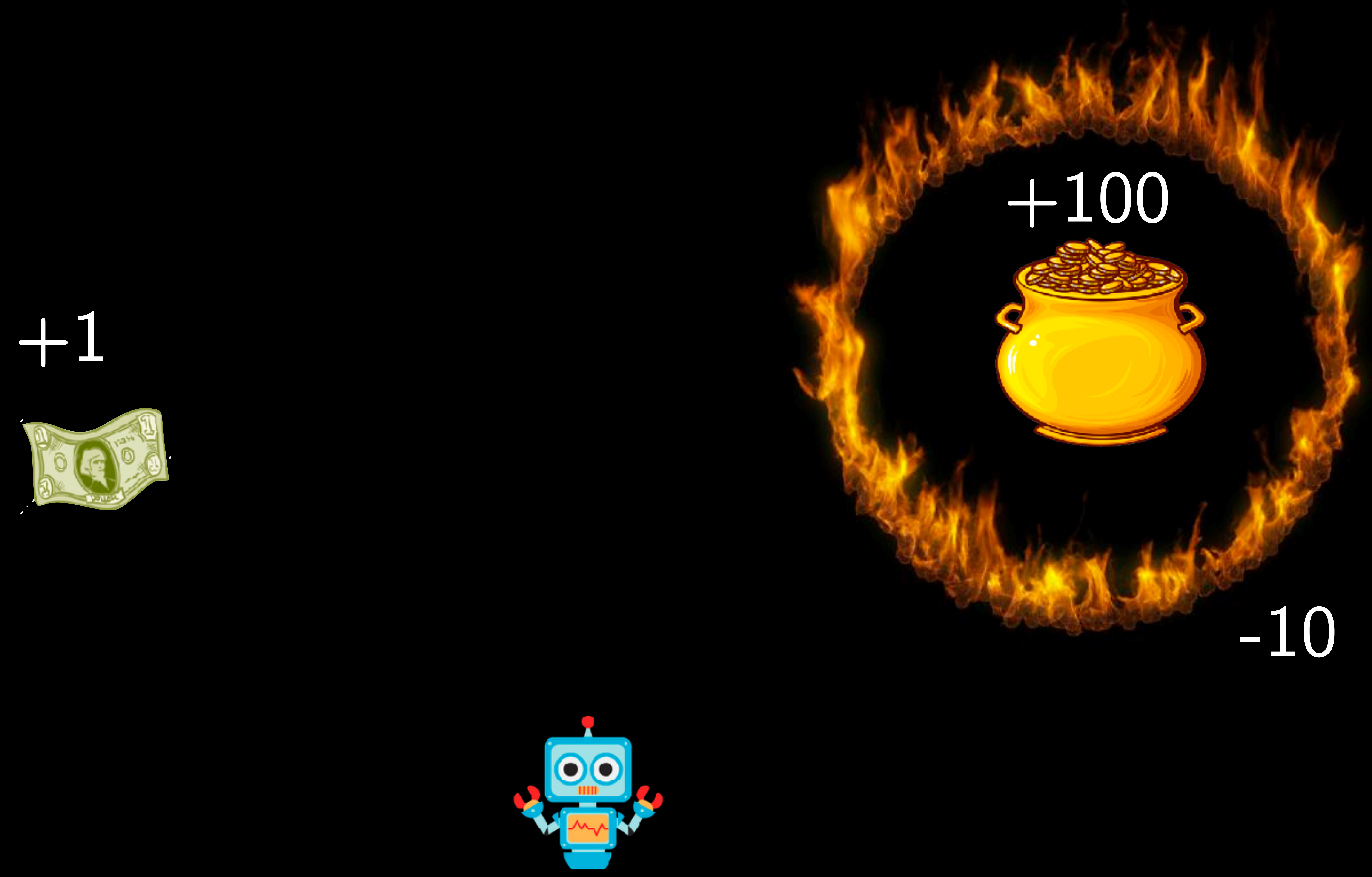
return π_θ

Modern variants are TRPO, PPO, etc

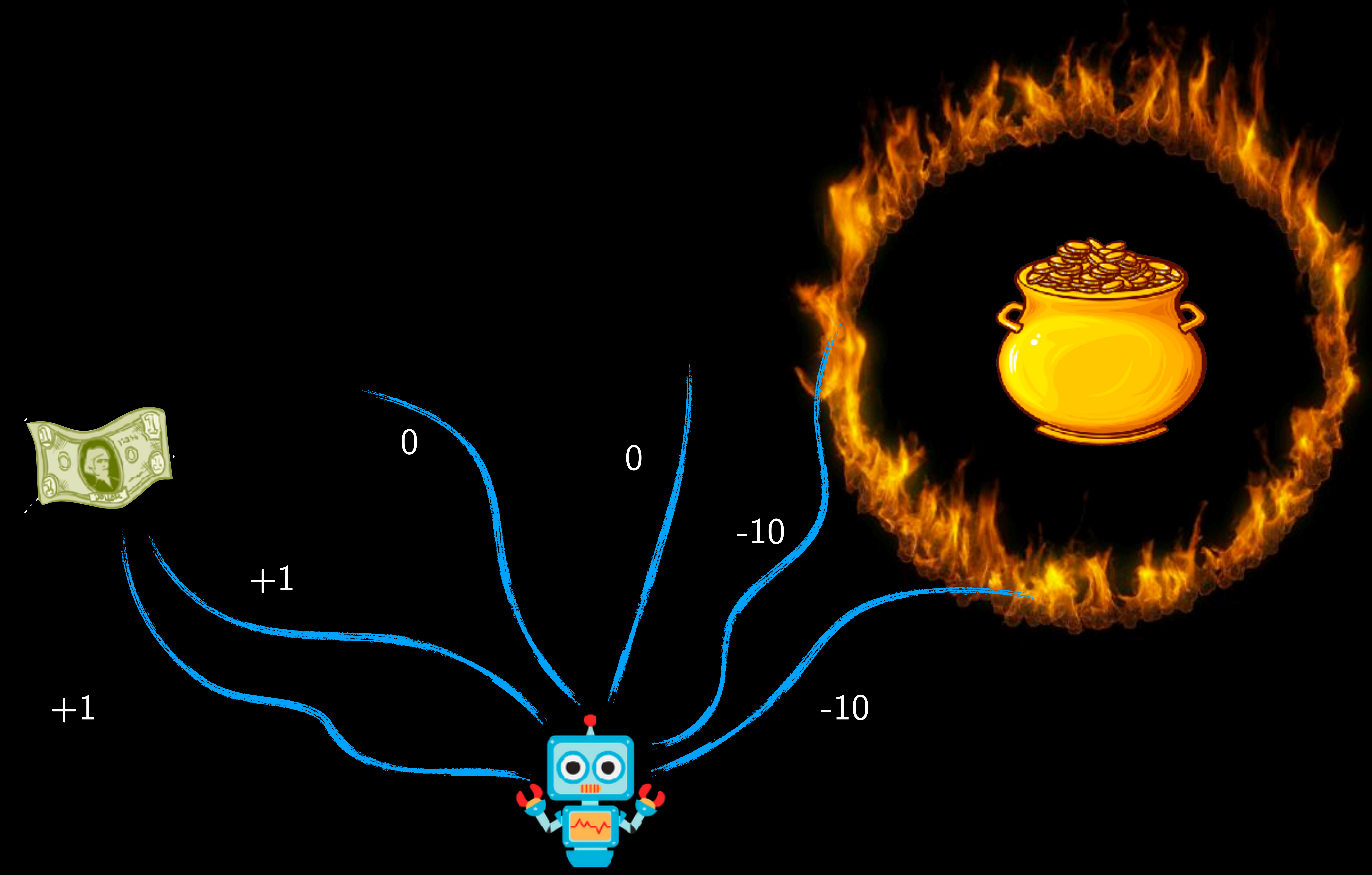
Nightmare 3:

Local Optima

The Ring of Fire

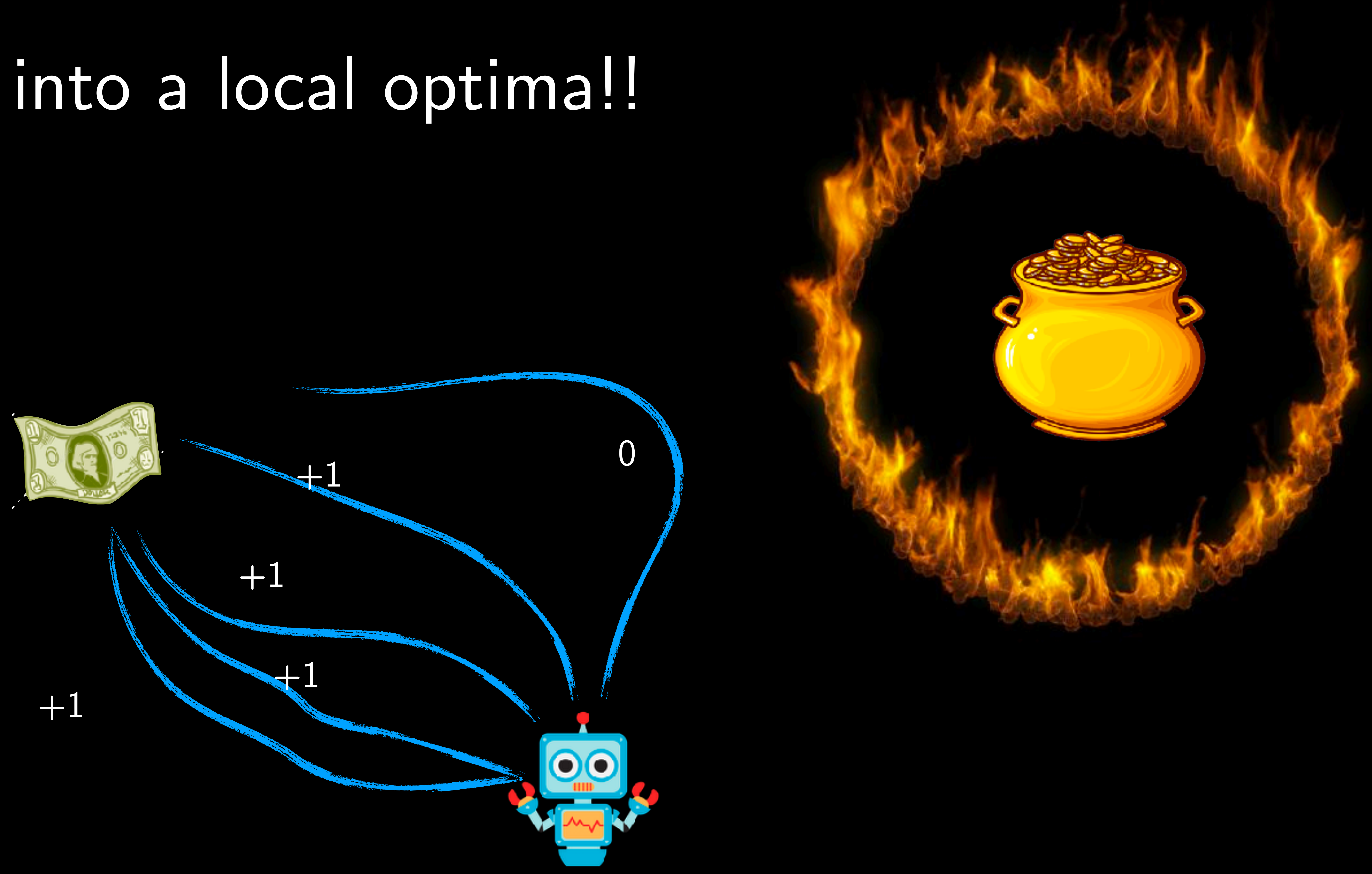


The Ring of Fire



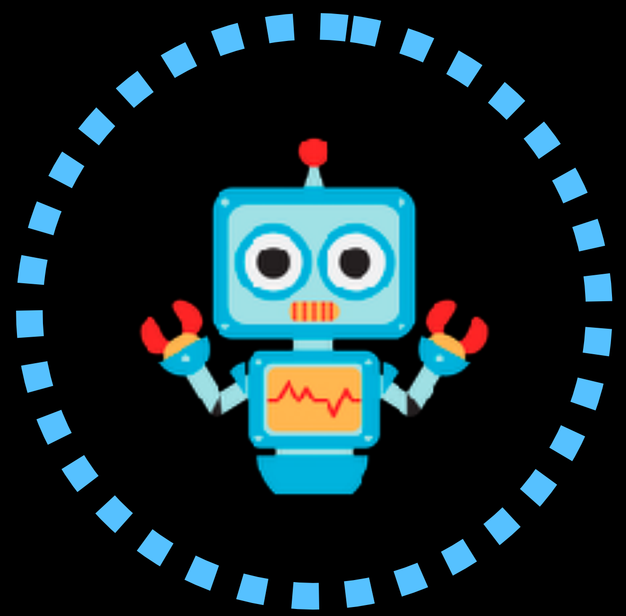
The Ring of Fire

Get's sucked into a local optima!!

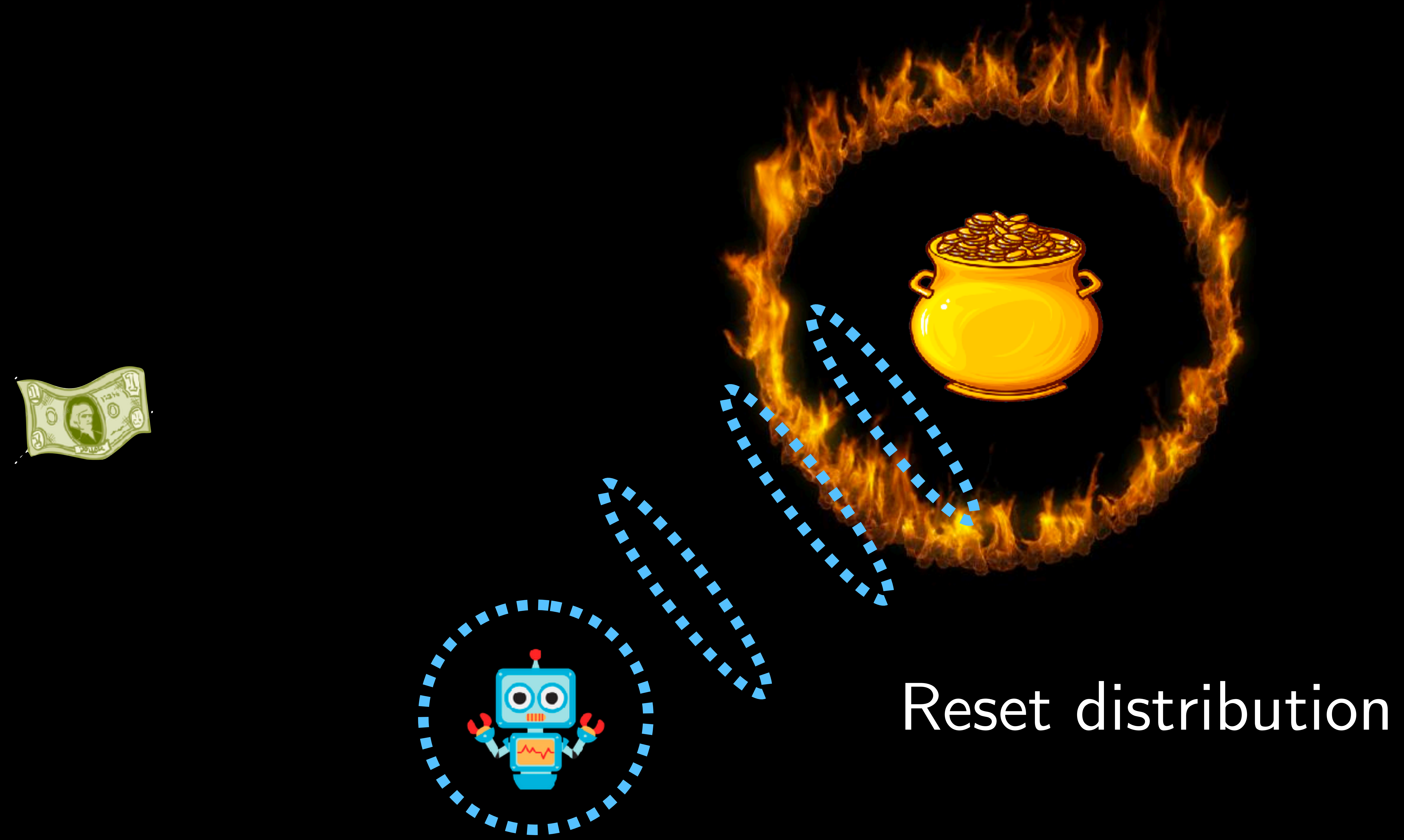


Idea: What if we had a "good reset distribution?"

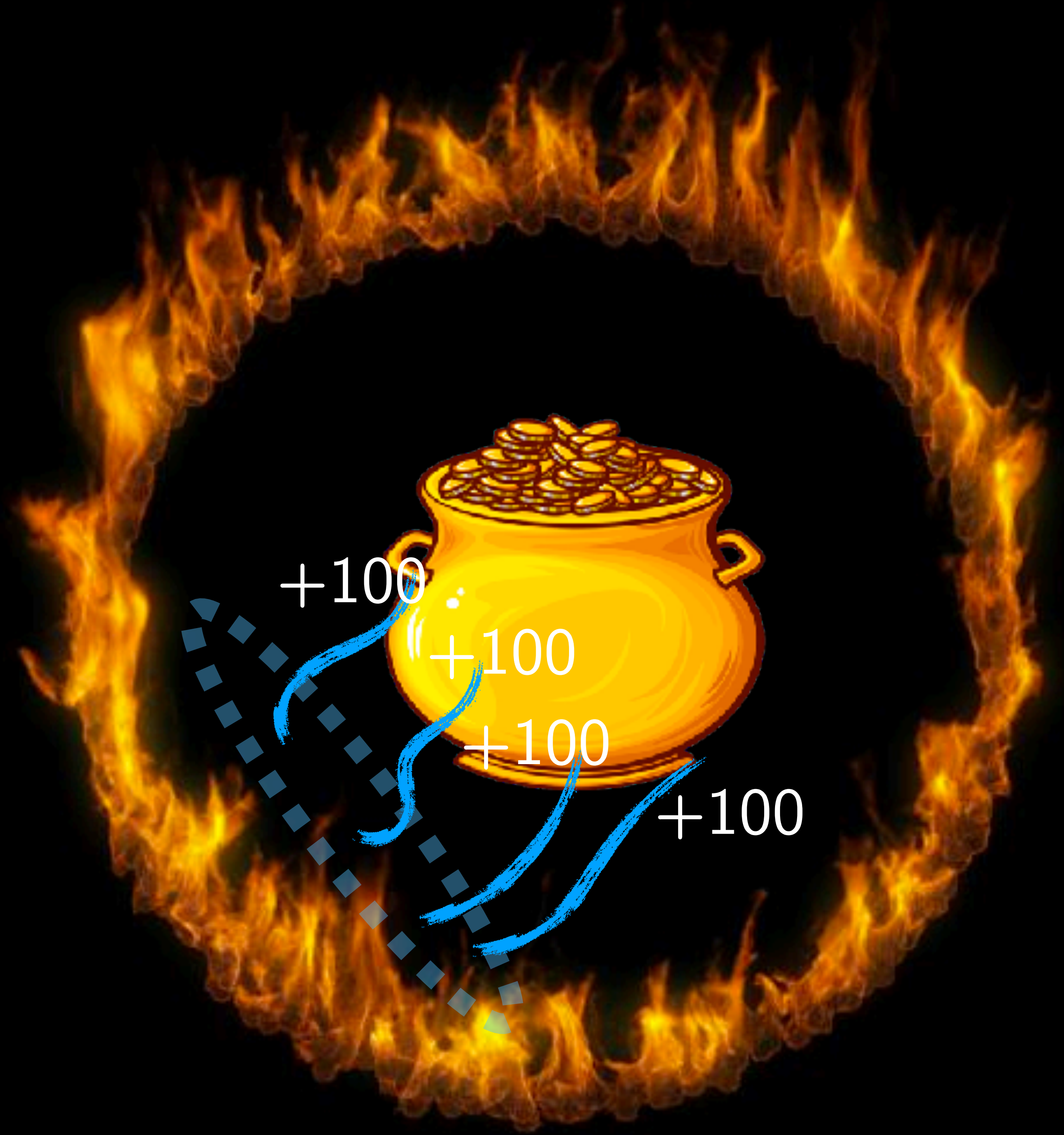
Start distribution



Idea: What if we had a "good reset distribution?"

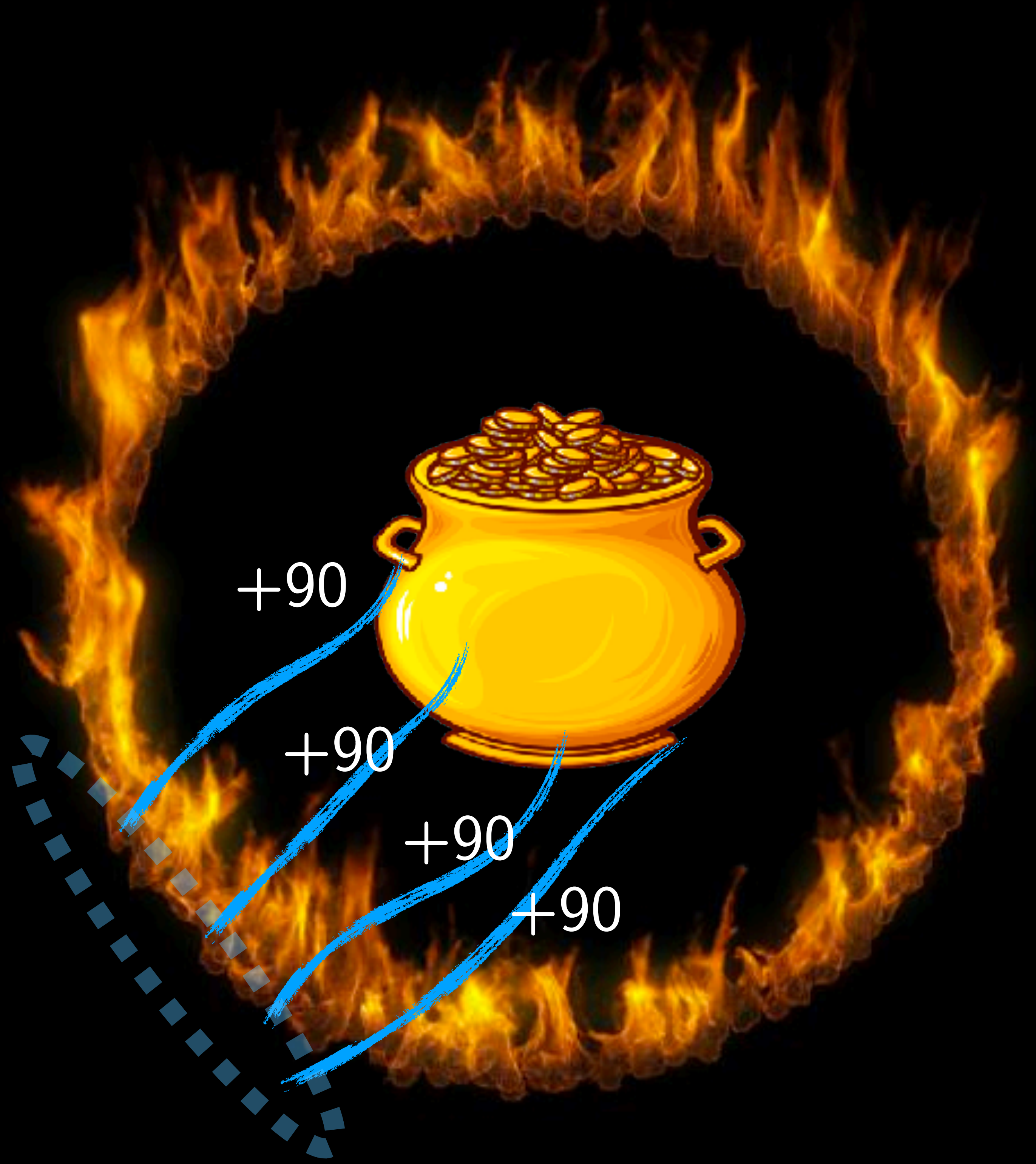


Idea: What if we had a “good reset distribution?”



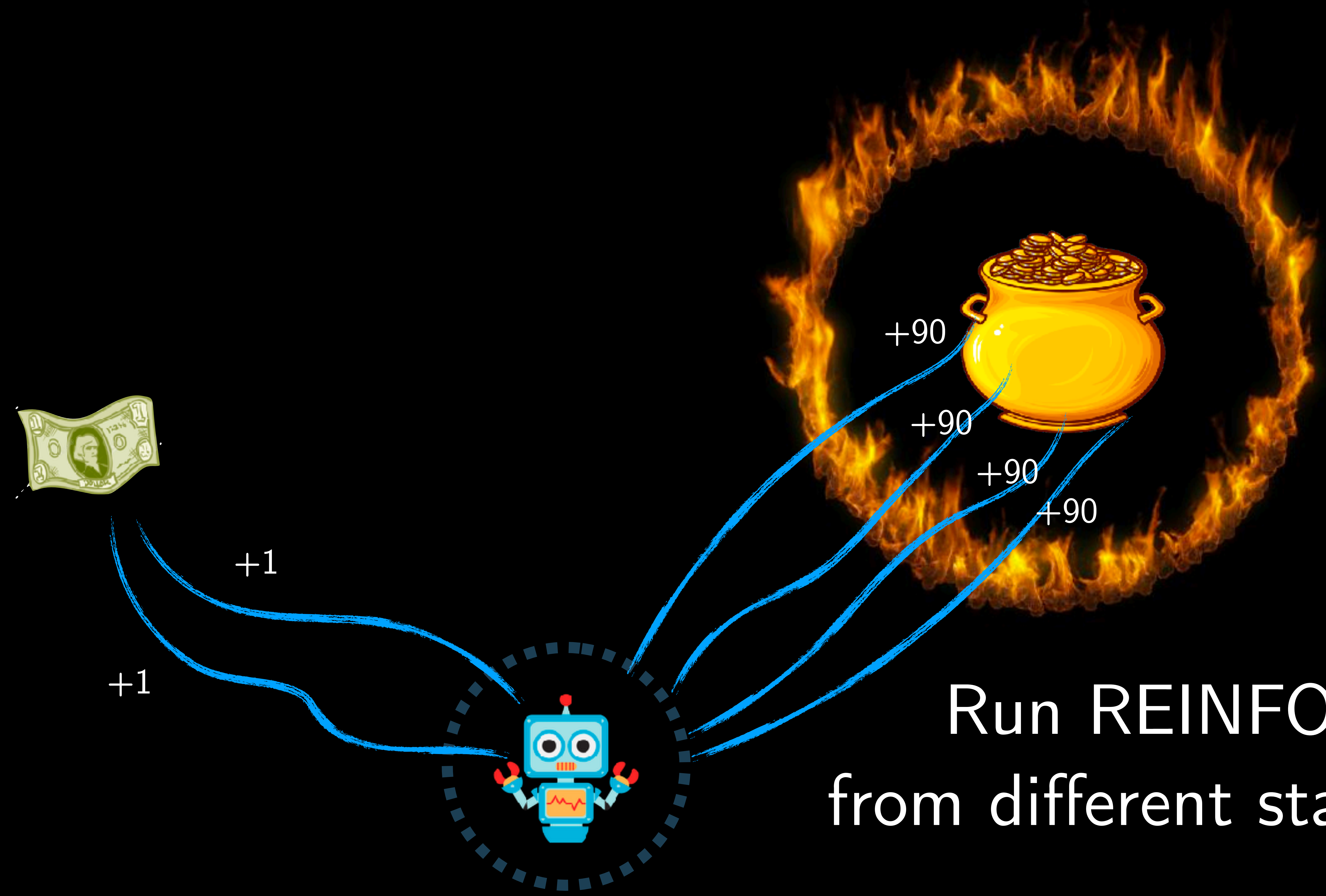
Run REINFORCE
from different start states

Idea: What if we had a “good reset distribution?”



Run REINFORCE
from different start states

Idea: What if we had a "good reset distribution?"



Run REINFORCE
from different start states

Solution: Use a good "reset" distribution

Choose a reset distribution $\mu(s)$ instead of start state distribution

Try your best to "cover" states the expert will visit

Justify using the PDL!

tl;dr

The Policy Gradient Theorem

$$\nabla_\theta J = E_{p(\xi|\theta)} \left[\sum_{t=0}^{T-1} \left(\nabla_\theta \log \pi_\theta(a_t|s_t) \left(\sum_{t'=0}^{t-1} r(s_{t'}, a_{t'}) + \sum_{t'=t}^{T-1} r(s_{t'}, a_{t'}) \right) \right) \right]$$
$$= E_{p(\xi|\theta)} \left[\sum_{t=0}^{T-1} \left(\nabla_\theta \log \pi_\theta(a_t|s_t) \sum_{t'=t}^{T-1} r(s_{t'}, a_{t'}) \right) \right],$$

$$\nabla_\theta J = E_{p(\xi|\theta)} \left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t)\, Q^{\pi_\theta}(s_t, a_t) \right]$$

15

1. High Variance: Subtract baseline
2. Distribution Shift: *Natural* Gradient Descent
3. Local Optima: Use Reset Distribution