PPO
& Importance Sampling

Lucas Janson and Sham Kakade

CS/Stat 184: Introduction to Reinforcement Learning
Fall 2023

Today

- Recap++
- Proximal Policy Optimization (PPO)
  - Importance Sampling
- Exploration?
- PG review

Recap++

(M=1) PG with a Learned Baseline:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
    1. Sup. Learning: Using N trajectories sampled under π_{θ^k}, estimate a baseline $\widetilde{b}_h$
       $\widetilde{b}(s) \approx V_h^{\theta^k}(s)$
    2. Obtain a trajectory $\tau \sim \rho_{\theta^k}$
       Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h)\left(R_h(\tau) - \widetilde{b}(s_h)\right)$
    3. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_\theta J(\theta^k)$

Note that regardless of our choice of $\widetilde{b}_h(s)$, we still get unbiased gradient estimates.

The Performance Difference Lemma (PDL)

- Let $\rho_{\widetilde{\pi},s}$ be the distribution of trajectories from starting state s acting under π. (we are making the starting distribution explicit now).
- For any two policies π and $\widetilde{\pi}$ and any state s,

$$V^{\widetilde{\pi}}(s) - V^{\pi}(s) = \mathbb{E}_{\tau \sim \rho_{\widetilde{\pi},s}} \left[\sum_{h=0}^{H-1} A_h^{\pi}(s_h, a_h) \right]$$

Comments:

- Helps us think about error analysis, instabilities of fitted PI, and sub-optimality.
- Helps to understand algorithm design (TRPO, NPG, PPO)
- This also motivates the use of “local” methods (e.g. policy gradient descent)

Trust Region Policy Optimization (TRPO)

1. Init π_0
2. For $k = 0, \ldots K$:

$$\theta^{k+1} = \arg\max_{\theta} \mathbb{E}_{s_0, \ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi_\theta(s_h)} A^{\pi^k}(s_h, a_h) \right]$$

$$\text{s.t. } KL\left(\rho_{\pi^k} | \rho_{\pi_\theta}\right) \leq \delta$$

3. Return π_K

- We want to maximize local advantage against π_{θ^k}, but we want the new policy to be close to π_{θ^k} (in the KL sense)
- **How do we implement this with sampled trajectories?**

TRPO is locally equivalent to the NPG

TRPO at iteration k:

$$\max_{\theta} \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h\sim\pi_\theta(s_h)} A^{\pi^k}(s_h, a_h) \right]$$

$$\text{s.t. } KL\left(\rho_{\pi^k} | \rho_{\pi_\theta}\right) \leq \delta$$

Intuition: maximize local adv subject to being incremental (in KL);

→ First-order Taylor expansion at θ^k

→ second-order Taylor expansion at θ^k

$$\max_{\theta} \nabla_\theta J(\pi_{\theta^k})^\top (\theta - \theta^k)$$

$$\text{s.t. } (\theta - \theta^k)^\top F_{\theta^k} (\theta - \theta^k) \leq \delta$$

(Where F_{θ^k} is the "Fisher Information Matrix")

NPG: A "leading order" equivalent program to TRPO:

1. Init π_0
2. For $k = 0, \ldots K$:

$$\theta^{k+1} = \arg\max_{\theta} \nabla_\theta J(\pi_{\theta^k})^\top (\theta - \theta^k)$$

$$\text{s.t. } (\theta - \theta^k)^\top F_{\theta^k} (\theta - \theta^k) \le \delta$$

3. Return π_K

- Where $\nabla_\theta J(\pi_{\theta^k})$ is the gradient at θ^k and
- F_θ is (basically) the Fisher information matrix at $\theta \in \mathbb{R}^d$, defined as:

$$F_\theta := \mathbb{E}_{\tau \sim \rho_\theta}\left[\nabla_\theta \ln \rho_\theta(\tau) \left(\nabla_\theta \ln \rho_\theta(\tau) \right)^\top \right] \in \mathbb{R}^{d \times d}$$

$$= \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h | s_h) \left(\nabla_\theta \ln \pi_\theta(a_h | s_h) \right)^\top \right]$$

There is a closed form update:

Linear objective and quadratic convex constraint, we can solve it optimally!

Indeed this gives us:

$$\theta^{k+1} = \theta^k + \eta F_{\theta^k}^{-1} \nabla_\theta J(\pi_{\theta^k})$$

Where $\eta = \sqrt{\dfrac{\delta}{\nabla_\theta J(\pi_{\theta^k})^\top F_{\theta^k}^{-1} \nabla_\theta J(\pi_{\theta^k})}}$

An Implementation: Sample Based NPG

1. Init π_0
2. For $k = 0, \ldots K$:
   - Estimate PG $\widehat{\nabla}_\theta J(\pi_{\theta^k})$
   - Estimate Fisher info-matrix: $\widehat{F}_{\theta^k} = \mathbb{E}_{\tau \sim \rho_{\theta^k}} \left[\sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) \left(\nabla \ln \pi_{\theta^k}(a_h | s_h) \right)^\top \right]$
   - Natural Gradient Ascent: $\theta^{k+1} = \theta^k + \eta \left(\widehat{F_{\theta^k}} + \lambda I \right)^{-1} \widehat{\nabla_\theta J(\pi_{\theta^k})}$
3. Return π_K

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1 + \exp(\theta)}, \frac{1}{1 + \exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

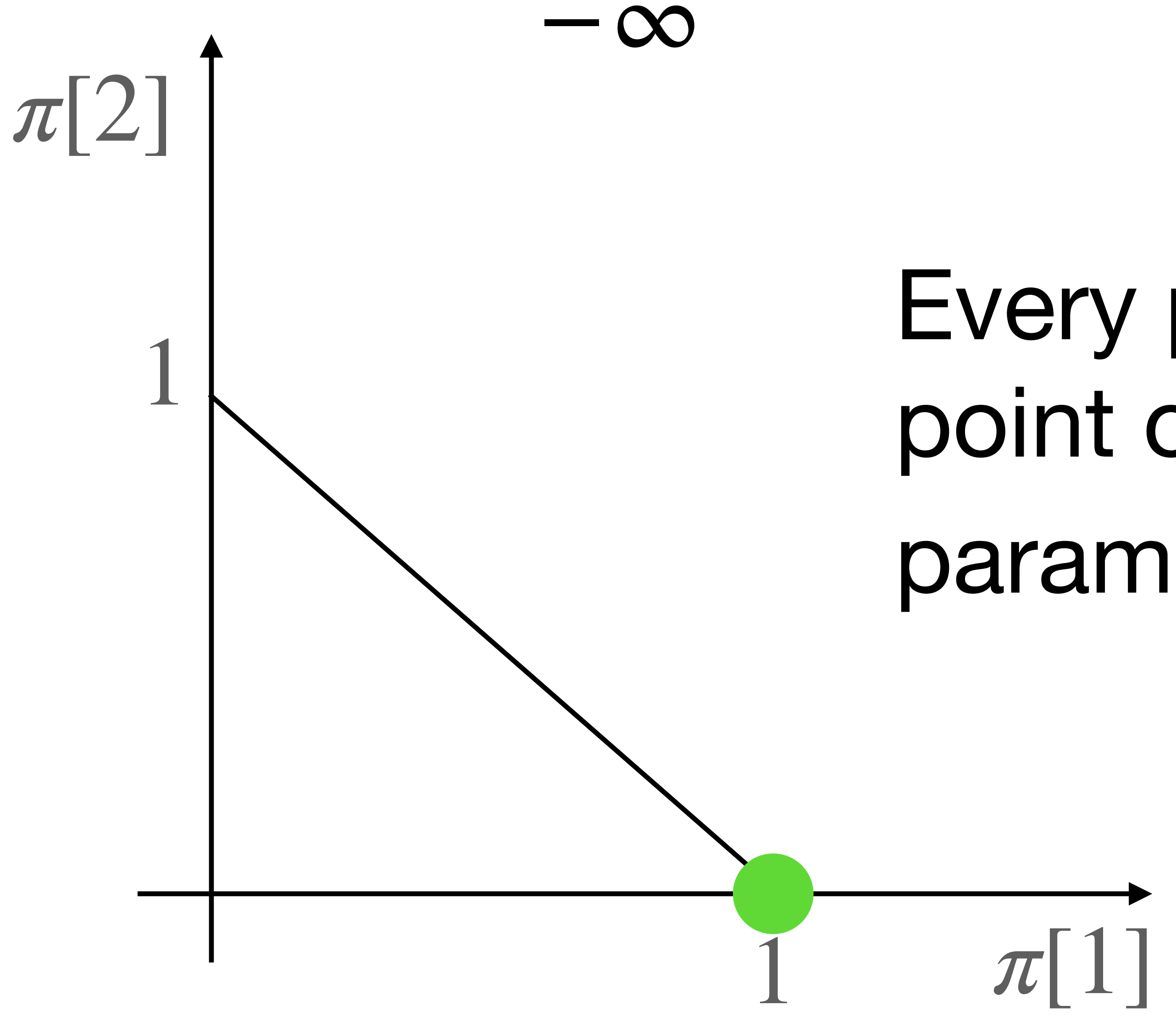
$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1 + \exp(\theta)}, \frac{1}{1 + \exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Every possible policy is a point on the line segment, parameterized by θ.

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

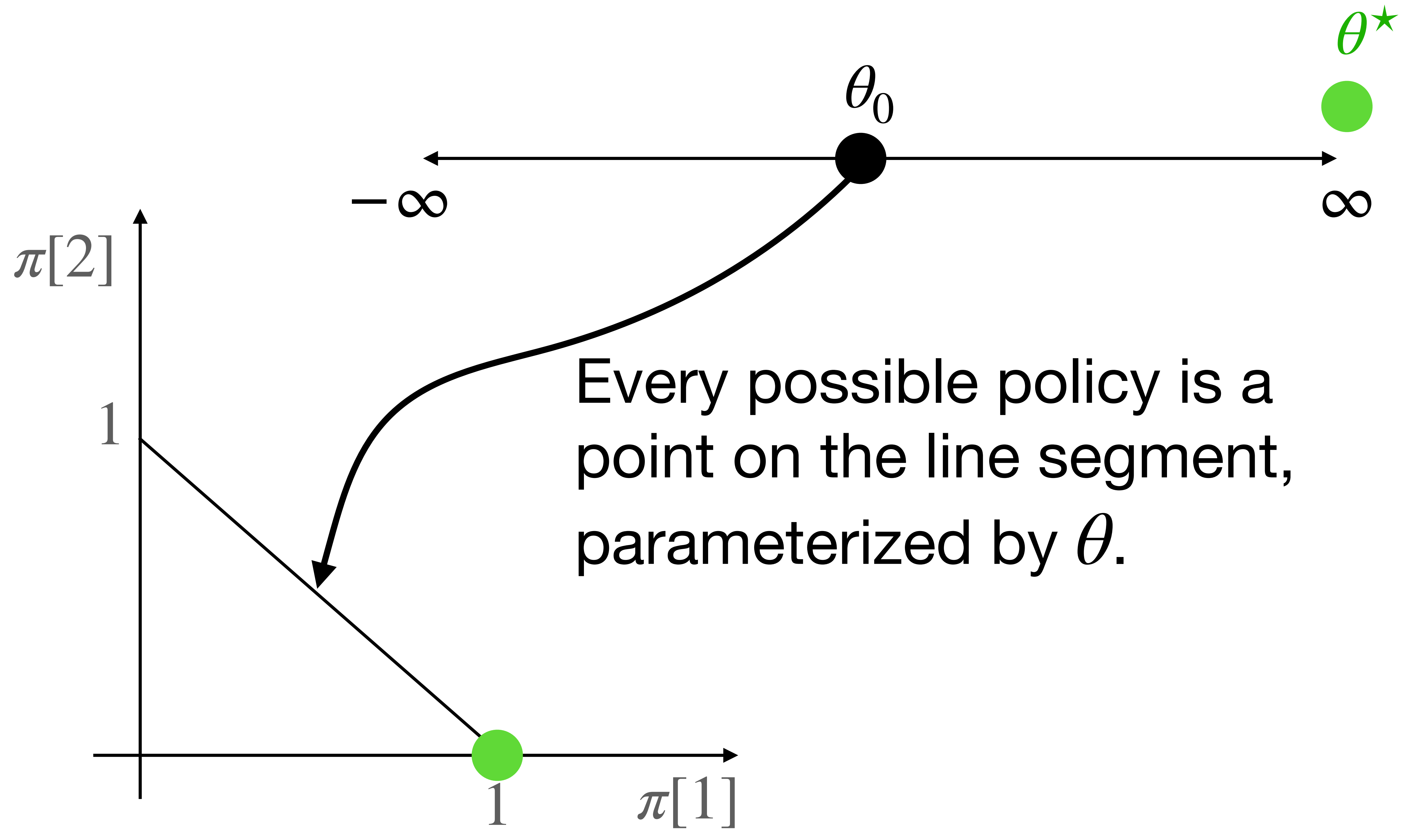
$$(\pi_\theta[1], \pi_\theta[2]) := \left(\overbrace{\frac{\exp(\theta)}{1+\exp(\theta)}}^{p(\theta)}, \overbrace{\frac{1}{1+\exp(\theta)}}^{1-p(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$

$$p'(\theta) = p(1-p)$$

$$= \frac{e^\theta}{(1+e^\theta)^2}$$

$$J'(\theta) = 99\, p'$$



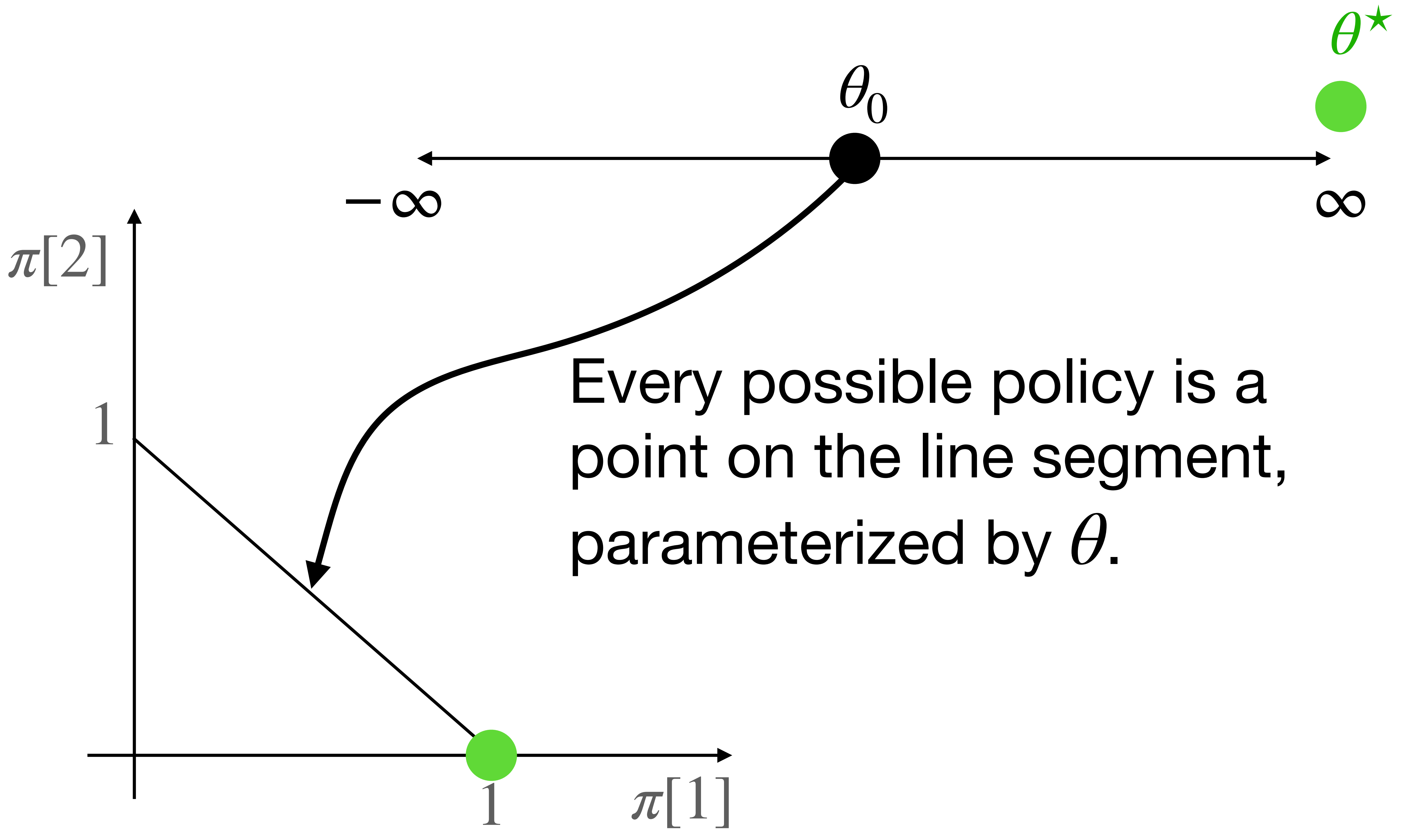
Every possible policy is a point on the line segment, parameterized by θ.

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)} \right)$$

Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Every possible policy is a point on the line segment, parameterized by θ.

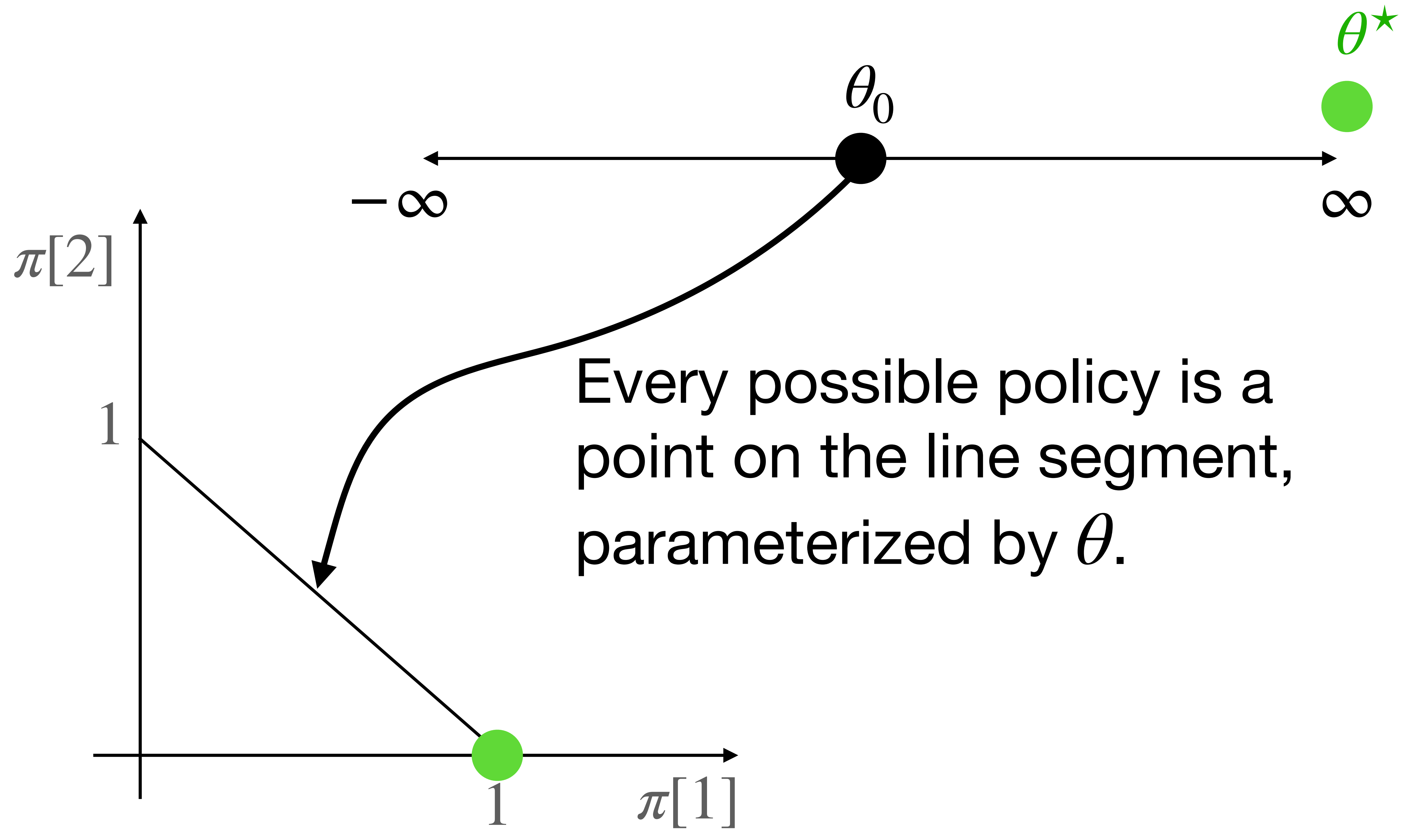
Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)}\right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$

Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

Exact PG: $\theta^{k+1} = \theta^k + \eta \dfrac{99\exp(\theta^k)}{(1+\exp(\theta^k))^2}$

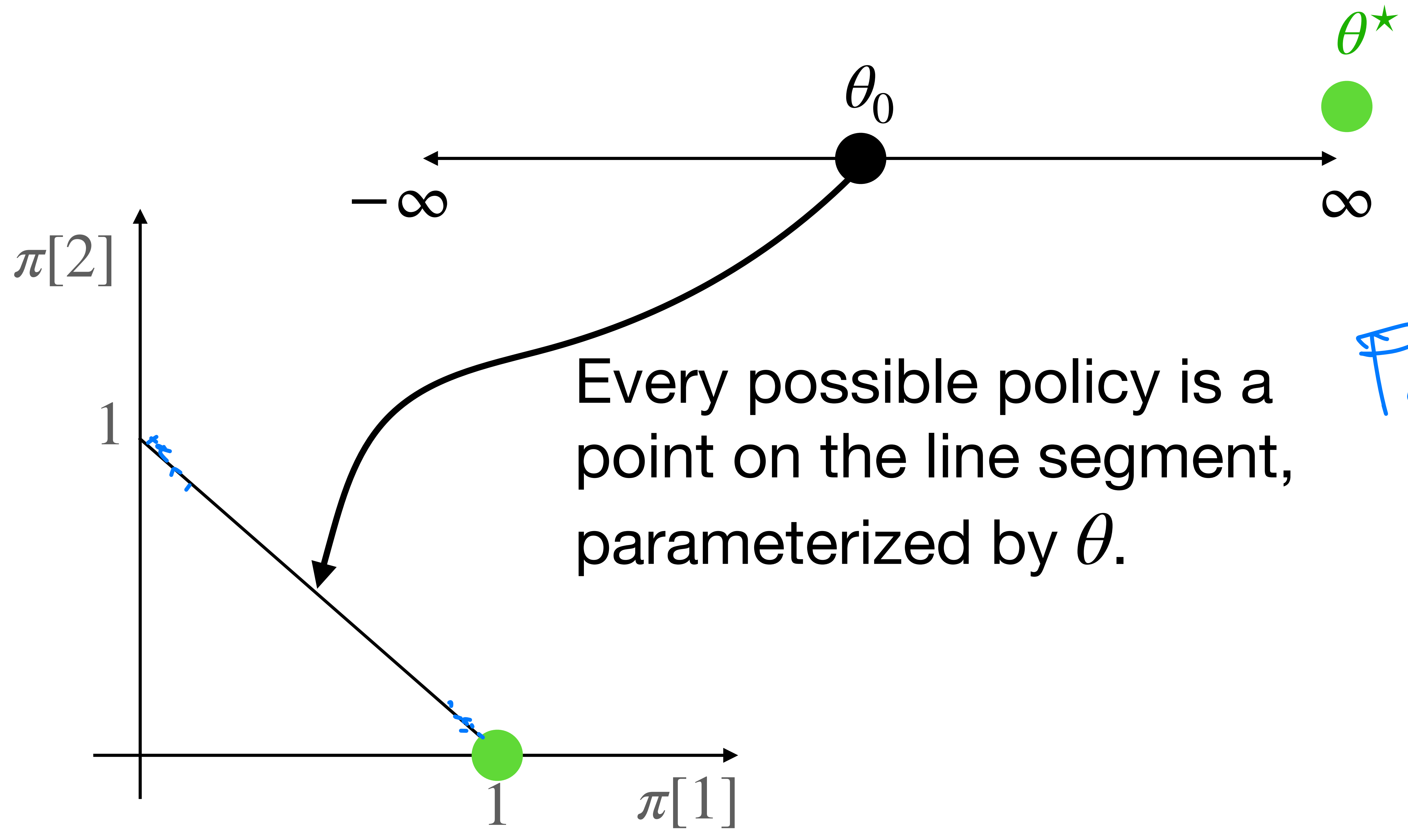


Every possible policy is a point on the line segment, parameterized by θ.

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Every possible policy is a point on the line segment, parameterized by θ.

Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

Exact PG: $\theta^{k+1} = \theta^k + \eta \dfrac{99\exp(\theta^k)}{(1+\exp(\theta^k))^2}$

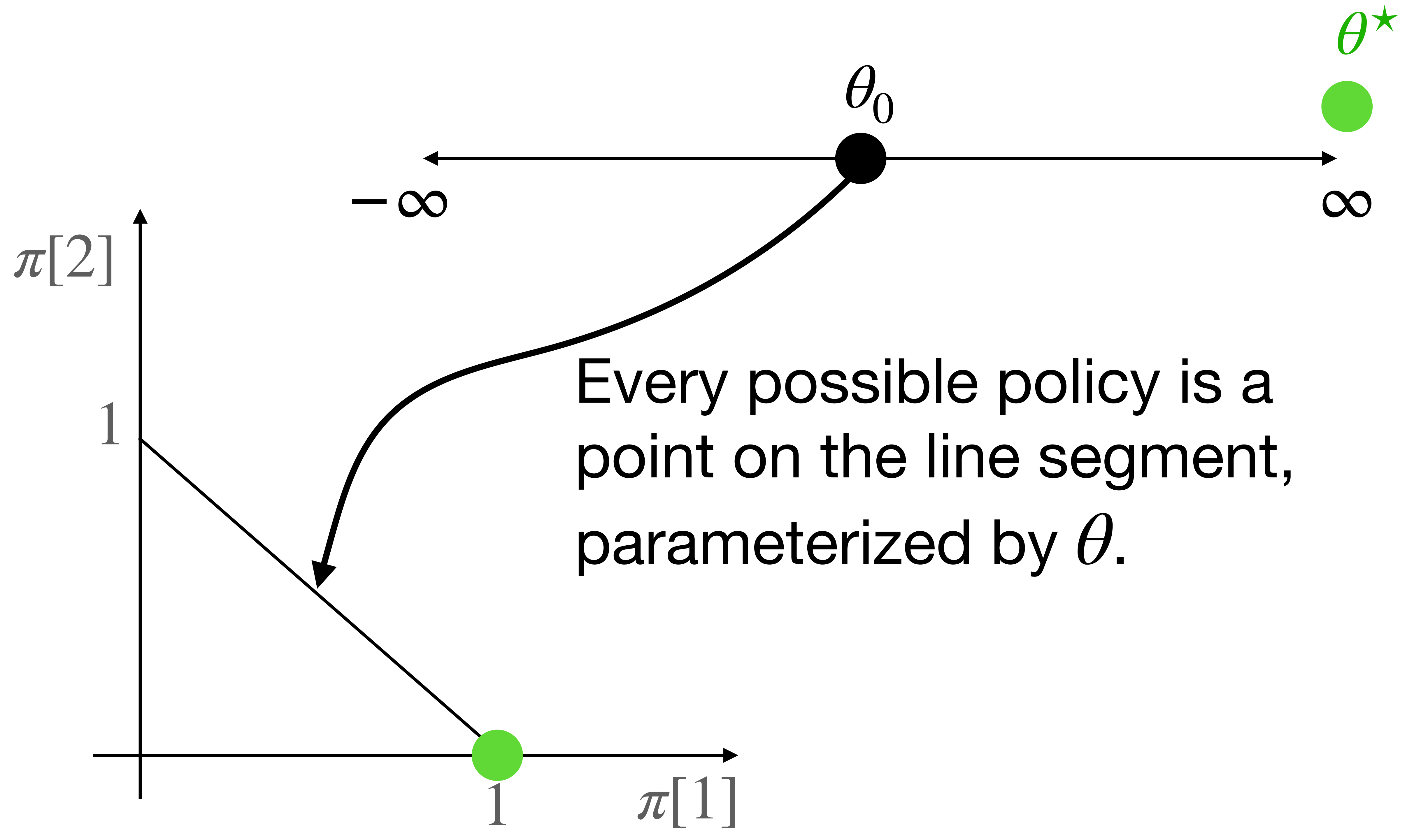
i.e., vanilla GA moves to $\theta = \infty$ with smaller and smaller steps, since $J'(\theta) \to 0$ as $\theta \to \infty$

$F(\theta) = \pi_\theta[1] \nabla \log \pi_\theta(1) + \pi_2(\theta) \nabla \log \pi_\theta(2)$

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

Exact PG: $\theta^{k+1} = \theta^k + \eta \dfrac{99\exp(\theta^k)}{(1+\exp(\theta^k))^2}$

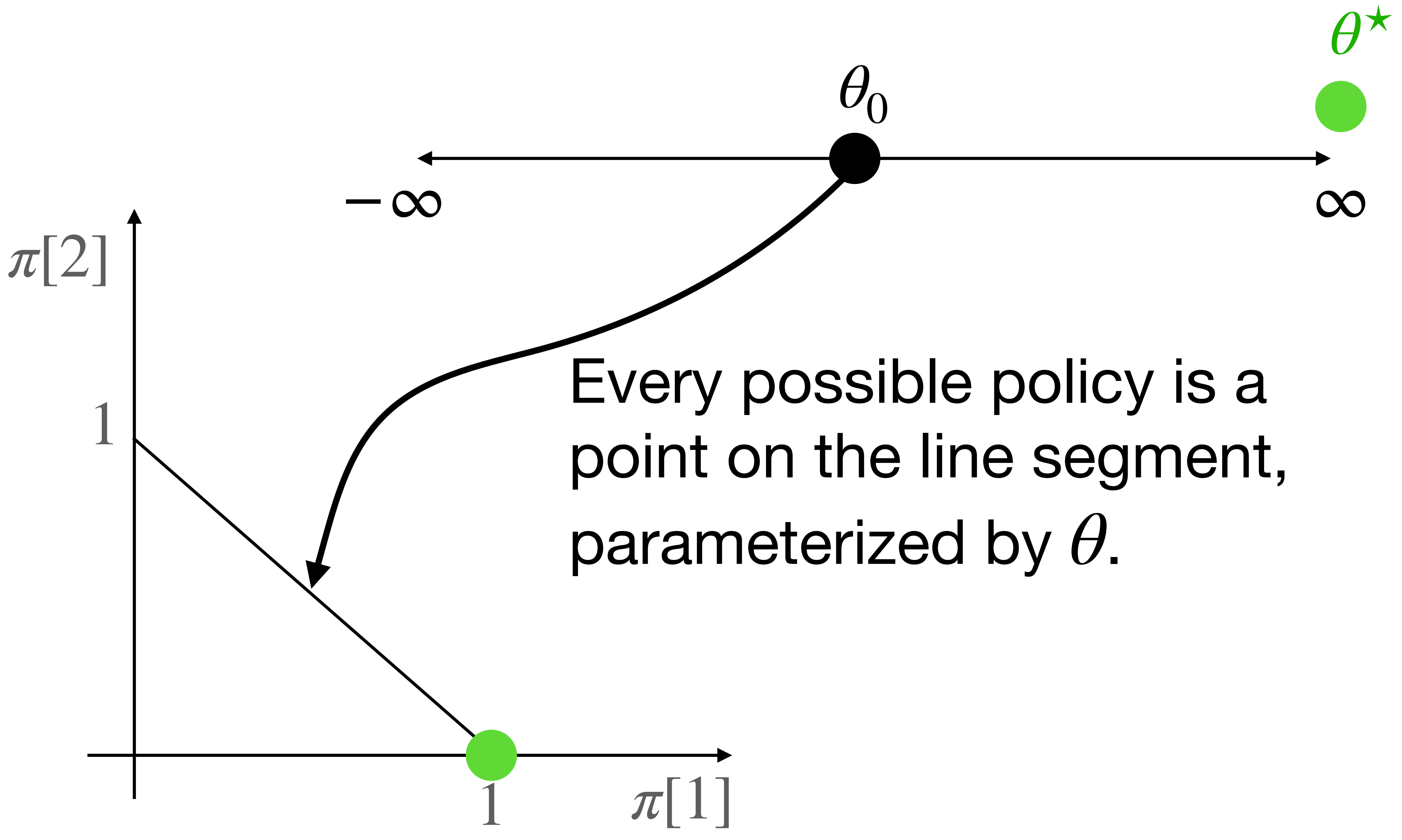
i.e., vanilla GA moves to $\theta = \infty$ with smaller and smaller steps, since $J'(\theta) \to 0$ as $\theta \to \infty$

Fisher information scalar: $F_\theta = \dfrac{\exp(\theta)}{(1+\exp(\theta))^2}$

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

Exact PG: $\theta^{k+1} = \theta^k + \eta \dfrac{99\exp(\theta^k)}{(1+\exp(\theta^k))^2}$

i.e., vanilla GA moves to $\theta = \infty$ with smaller and smaller steps, since $J'(\theta) \to 0$ as $\theta \to \infty$

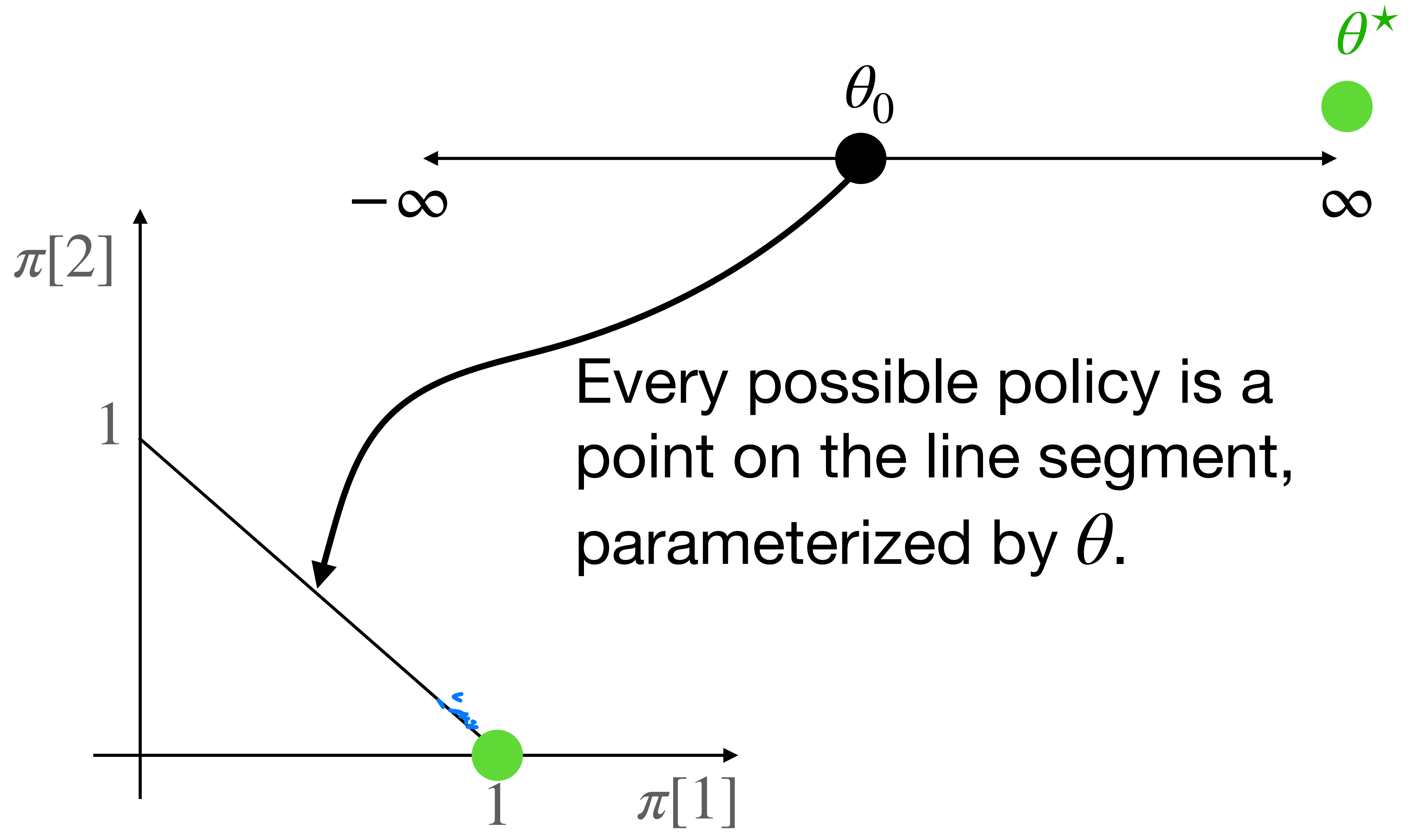
Fisher information scalar: $F_\theta = \dfrac{\exp(\theta)}{(1+\exp(\theta))^2}$

NPG: $\theta^{k+1} = \theta^k + \eta \dfrac{J'(\theta^k)}{F_{\theta^k}}$

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)} \right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

Exact PG: $\theta^{k+1} = \theta^k + \eta \dfrac{99\exp(\theta^k)}{(1+\exp(\theta^k))^2}$

i.e., vanilla GA moves to $\theta = \infty$ with smaller and smaller steps, since $J'(\theta) \to 0$ as $\theta \to \infty$

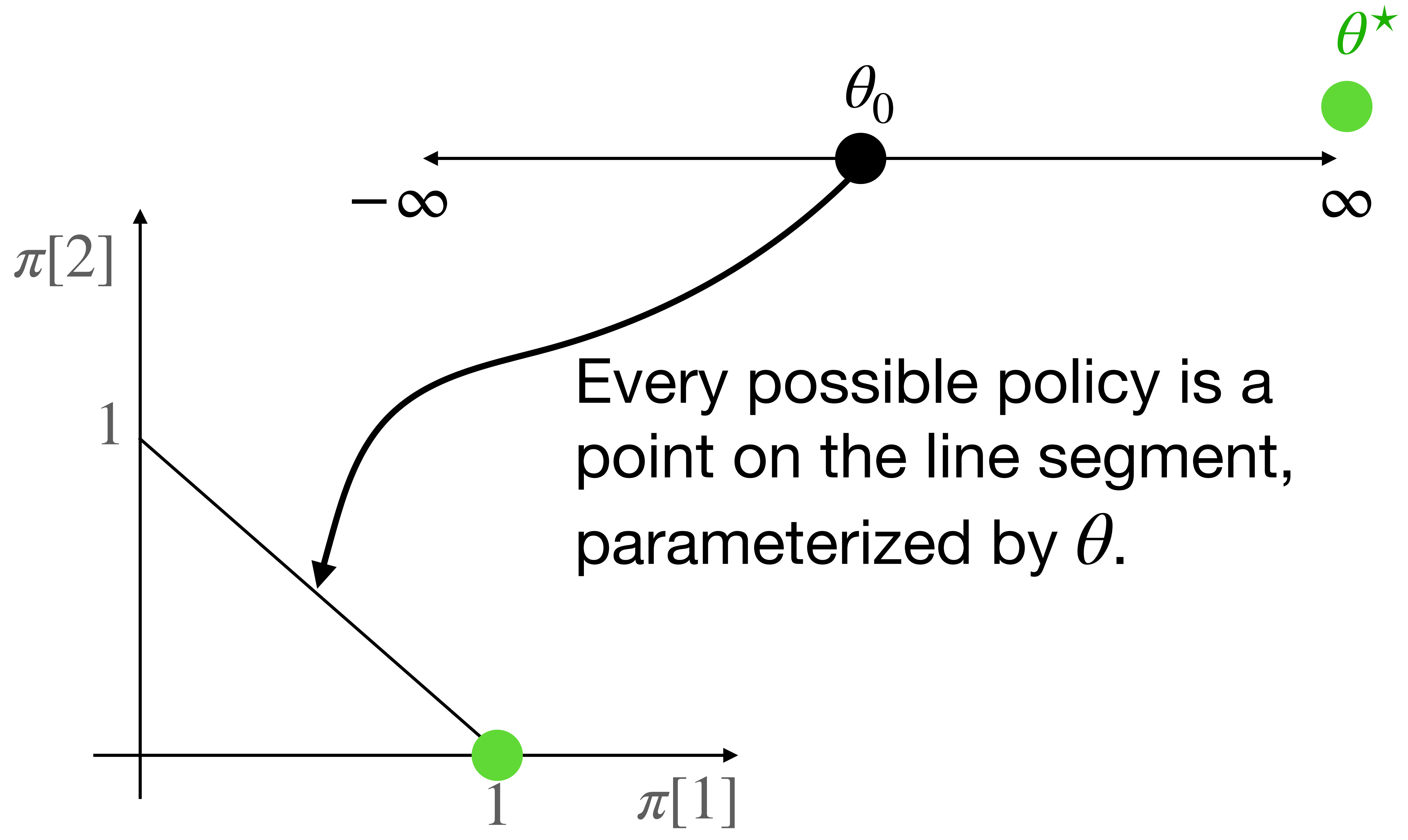
Fisher information scalar: $F_\theta = \dfrac{\exp(\theta)}{(1+\exp(\theta))^2}$

NPG: $\theta^{k+1} = \theta^k + \eta \dfrac{J'(\theta^k)}{F_{\theta^k}} = \theta^k + \eta \cdot 99$

Example of Natural Gradient on 1-d problem: 2 actions, 1 state

$$(\pi_\theta[1], \pi_\theta[2]) := \left(\frac{\exp(\theta)}{1+\exp(\theta)}, \frac{1}{1+\exp(\theta)}\right)$$

$$J(\theta) = 100 \cdot \pi_\theta[1] + 1 \cdot \pi_\theta[2]$$



Gradient: $J'(\theta) = \dfrac{99\exp(\theta)}{(1+\exp(\theta))^2}$

Exact PG: $\theta^{k+1} = \theta^k + \eta \dfrac{99\exp(\theta^k)}{(1+\exp(\theta^k))^2}$

i.e., vanilla GA moves to $\theta = \infty$ with smaller and smaller steps, since $J'(\theta) \to 0$ as $\theta \to \infty$

Fisher information scalar: $F_\theta = \dfrac{\exp(\theta)}{(1+\exp(\theta))^2}$

NPG: $\theta^{k+1} = \theta^k + \eta \dfrac{J'(\theta^k)}{F_{\theta^k}} = \theta_t + \eta \cdot 99$

NPG moves to $\theta = \infty$ much more quickly (for a fixed η)

Today:

Today

- Recap++
- Proximal Policy Optimization (PPO)
  - Importance Sampling
- Exploration?
- PG review

Back to TRPO/NPG

1. Init π_0
2. For $k = 0, \ldots K$:

$$\theta^{k+1} = \arg\max_{\theta} \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi_\theta(s_h)} A^{\pi^k}(s_h, a_h) \right]$$

$$\text{s.t. } KL\left(\rho_{\pi^k} | \rho_{\pi_\theta}\right) \leq \delta$$

3. Return π_K

- The difficulty with TRPO and NPG is that they could be computationally costly. Need to solve constrained optimization or matrix inversion ("second order") problems.
- Can we use a method which only uses gradients?

Let's try to use a "Lagrangian relaxation" of TRPO

Proximal Policy Optimization (PPO)

1. Init π_0, choose λ
2. For $k = 0, \ldots K$:

$$\theta^{k+1} = \arg\max_{\theta} \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi_\theta(s_h)} A^{\pi^k}(s_h, a_h) \right] \underbrace{-\lambda KL\left(\rho_{\pi^k} | \rho_{\pi_\theta}\right)}_{\text{regularization}}$$

3. Return π_K

The regularization term is:

$$KL\left(\rho_{\pi_{\theta^k}} | \rho_{\pi_\theta}\right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\ln \frac{\rho_{\pi_{\theta^k}}(\tau)}{\rho_{\pi_\theta}(\tau)}\right]$$

The regularization term is:

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

$$KL\left(\rho_{\pi_{\theta^k}} \mid \rho_{\pi_\theta}\right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\ln \frac{\rho_{\pi_{\theta^k}}(\tau)}{\rho_{\pi_\theta}(\tau)}\right]$$

The regularization term is:

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

$$KL\left(\rho_{\pi_{\theta^k}} \mid \rho_{\pi_\theta}\right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\ln \frac{\rho_{\pi_{\theta^k}}(\tau)}{\rho_{\pi_\theta}(\tau)}\right]$$

$$= \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\sum_{h=0}^{H-1} \ln \frac{\pi_{\theta^k}(a_h \mid s_h)}{\pi_\theta(a_h \mid s_h)}\right]$$

The regularization term is:

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

$$KL\left(\rho_{\pi_{\theta^k}} \mid \rho_{\pi_\theta}\right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\ln \frac{\rho_{\pi_{\theta^k}}(\tau)}{\rho_{\pi_\theta}(\tau)}\right]$$

$$= \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\sum_{h=0}^{H-1} \ln \frac{\pi_{\theta^k}(a_h \mid s_h)}{\pi_\theta(a_h \mid s_h)}\right]$$

$$= \mathbb{E}_{\tau \sim \rho_{\pi_{\theta^k}}}\left[\sum_{h=0}^{H-1} \ln \frac{1}{\pi_\theta(a_h \mid s_h)}\right] + \left[\text{term not a function of } \theta\right]$$

Proximal Policy Optimization (PPO)

1. Init π_0, choose λ
2. For $k = 0, \ldots K$:
   use SGD to optimize:
   $\theta^{k+1} \approx \arg\max_{\theta} \ell^k(\theta)$
   where:
   $$\ell^k(\theta) := \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\mathbb{E}_{a_h\sim\pi_\theta(s_h)}A^{\pi^k}(s_h, a_h)\right] - \lambda\mathbb{E}_{\tau\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\ln\frac{1}{\pi_\theta(a_h \mid s_h)}\right]$$
3. Return π_K

How do we estimate this objective?

Today

- Recap++
- Proximal Policy Optimization (PPO)
    - Importance Sampling
- Exploration?
- PG review

Importance Sampling

Importance Sampling

- Suppose we seek to estimate $E_{x \sim \widetilde{p}}[f(x)]$.

Importance Sampling

$? \; f, \tilde{p}$

easy $\quad \hat{E}_{x \sim p}[f(x)]$

- Suppose we seek to estimate $E_{x \sim \tilde{p}}[f(x)]$.
- Assume: we have an (i.i.d.) dataset $x_1, \ldots x_N$, where $x_i \sim p$, where p is known, and
  - f and $\tilde{p}$ are known.
  - we are not able to collect values of $f(x)$ for $x \sim \tilde{p}$.
    (e.g. we have already collected our data from some costly experiment).

Importance Sampling

- Suppose we seek to estimate $E_{x \sim \tilde{p}}[f(x)]$.
- Assume: we have an (i.i.d.) dataset $x_1, \ldots x_N$, where $x_i \sim p$, where p is known, and
  - f and $\tilde{p}$ are known.
  - we are not able to collect values of $f(x)$ for $x \sim \tilde{p}$.
    (e.g. we have already collected our data from some costly experiment).
- Note: $E_{x \sim \tilde{p}}\left[f(x)\right] = \sum_x \tilde{p}(x) f(x) = \sum_x p(x) \frac{\tilde{p}(x)}{p(x)} f(x) = E_{x \sim p}\left[\frac{\tilde{p}(x)}{p(x)} f(x)\right]$

Importance Sampling

- Suppose we seek to estimate $E_{x\sim\tilde{p}}[f(x)]$.
- Assume: we have an (i.i.d.) dataset $x_1, \ldots x_N$, where $x_i \sim p$, where p is known, and
  - f and $\tilde{p}$ are known.
  - we are not able to collect values of $f(x)$ for $x \sim \tilde{p}$.
    (e.g. we have already collected our data from some costly experiment).
- Note: $E_{x\sim\tilde{p}}\left[f(x)\right] = \underset{x\sim p}{E_{\cancel{x\sim\tilde{p}}}}\left[\frac{\tilde{p}(x)}{p(x)} f(x)\right]$

Importance Sampling

- Suppose we seek to estimate $E_{x\sim\widetilde{p}}[f(x)]$.
- Assume: we have an (i.i.d.) dataset $x_1, \ldots x_N$, where $x_i \sim p$, where p is known, and
  - f and $\widetilde{p}$ are known.
  - we are not able to collect values of $f(x)$ for $x \sim \widetilde{p}$.
    (e.g. we have already collected our data from some costly experiment).
- Note: $E_{x\sim\widetilde{p}}\left[f(x)\right] = E_{x\sim D}\left[\frac{\widetilde{p}(x)}{p(x)}f(x)\right]$
- An unbiased estimate of $E_{x\sim\widetilde{p}}[f(x)]$ is given by $\frac{1}{N}\sum_i \frac{\widetilde{p}(x_i)}{p(x_i)}f(x_i)$

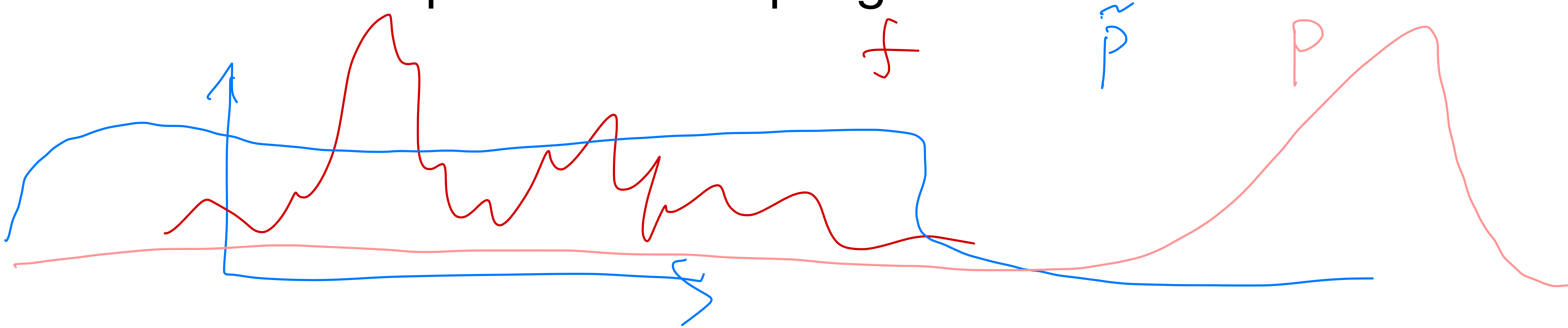
Importance Sampling

- Suppose we seek to estimate $E_{x\sim\tilde{p}}[f(x)]$.
- Assume: we have an (i.i.d.) dataset $x_1, \ldots x_N$, where $x_i \sim p$, where p is known, and
  - f and $\tilde{p}$ are known.
  - we are not able to collect values of $f(x)$ for $x \sim \tilde{p}$.
    (e.g. we have already collected our data from some costly experiment).
- Note: $E_{x\sim\tilde{p}}\left[f(x)\right] = E_{x\sim D}\left[\frac{\tilde{p}(x)}{p(x)}f(x)\right]$
- An unbiased estimate of $E_{x\sim\tilde{p}}[f(x)]$ is given by $\frac{1}{N}\sum_i \frac{\tilde{p}(x_i)}{p(x_i)}f(x_i)$
- Terminology:
  $\tilde{p}(x)$ is the target distribution; $p(x)$ is the proposal distribution;
  $\tilde{p}(x)/p(x)$ is the likelihood ratio.

Importance Sampling

- Suppose we seek to estimate $E_{x\sim\tilde{p}}[f(x)]$.
- Assume: we have an (i.i.d.) dataset $x_1, \ldots x_N$, where $x_i \sim p$, where p is known, and
  - f and $\tilde{p}$ are known.
  - we are not able to collect values of $f(x)$ for $x \sim \tilde{p}$.
    (e.g. we have already collected our data from some costly experiment).
- Note: $E_{x\sim\tilde{p}}\left[f(x)\right] = E_{x\sim D}\left[\frac{\tilde{p}(x)}{p(x)}f(x)\right]$
- An unbiased estimate of $E_{x\sim\tilde{p}}[f(x)]$ is given by $\frac{1}{N}\sum_i \frac{\tilde{p}(x_i)}{p(x_i)}f(x_i)$
- Terminology:
  $\tilde{p}(x)$ is the target distribution; $p(x)$ is the proposal distribution;
  $\tilde{p}(x)/p(x)$ is the likelihood ratio.
- What about the variance of this estimator?

Importance Sampling & Variance



Back to Estimating $\ell^k(\theta)$

Back to Estimating $\ell^k(\theta)$

- To estimate,

$$\ell^k(\theta) := \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\mathbb{E}_{a_h\sim\pi_{\theta}(s_h)}A^{\pi^k}(s_h,a_h)\right] - \lambda\mathbb{E}_{\tau\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\ln\frac{1}{\pi_{\theta}(a_h\,|\,s_h)}\right]$$

Back to Estimating $\ell^k(\theta)$

- To estimate,

$$\ell^k(\theta) := \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\mathbb{E}_{a_h\sim\pi_\theta(s_h)}A^{\pi^k}(s_h,a_h)\right] - \lambda\mathbb{E}_{\tau\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\ln\frac{1}{\pi_\theta(a_h\,|\,s_h)}\right]$$

- we will use importance sampling:

$$\ell^k(\theta) := \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\mathbb{E}_{a_h\sim\pi^k(s_h)}\left[\frac{\pi_\theta(s_h)}{\pi^k(s_h)}A^{\pi^k}(s_h,a_h)\right]\right] - \lambda\mathbb{E}_{\tau\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1}\ln\frac{1}{\pi_\theta(a_h\,|\,s_h)}\right]$$

Back to Estimating $\ell^k(\theta)$

$E_{a \sim \pi_\theta}\left[\nabla \log \pi_\theta A^k\right]$

- To estimate,

$$\ell^k(\theta) := \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi_\theta(s_h)} A^{\pi^k}(s_h, a_h) \right] - \lambda \mathbb{E}_{\tau \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \ln \frac{1}{\pi_\theta(a_h \mid s_h)} \right]$$

- we will use importance sampling:

$$\ell^k(\theta) := \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi^k(s_h)} \left[\frac{\pi_\theta(s_h)}{\pi^k(s_h)} A^{\pi^k}(s_h, a_h) \right] \right] - \lambda \mathbb{E}_{\tau \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \ln \frac{1}{\pi_\theta(a_h \mid s_h)} \right]$$

$$= \mathbb{E}_{\tau \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \left(\frac{\pi_\theta(s_h)}{\pi^k(s_h)} A^{\pi^k}(s_h, a_h) - \lambda \ln \frac{1}{\pi_\theta(a_h \mid s_h)} \right) \right]$$

Estimating $\ell^k(\theta)$

Estimating $\ell^k(\theta)$

1. Using N trajectories sampled under ρ^k to learn a $\widetilde{b}_h$
   $\widetilde{b}(s, h) \approx V_h^{\pi^k}(s)$

Estimating $\ell^k(\theta)$

1. Using N trajectories sampled under ρ^k to learn a $\widetilde{b}_h$
   $\widetilde{b}(s,h) \approx V_h^{\pi^k}(s)$
2. Obtain M NEW trajectories $\tau_1, \ldots \tau_M \sim \rho^k$
   Set $\widehat{\ell}^k(\theta) = \frac{1}{M}\sum_{m=1}^{M}\sum_{h=0}^{H-1}\left(\frac{\pi_\theta(s_h^m)}{\pi^k(s_h^m)}\left(R_h(\tau^m) - \widetilde{b}(s_h^m, h)\right) - \lambda \ln \frac{1}{\pi_\theta(a_h^m \mid s_h^m)}\right)$

do some # of grad. steps on $\hat{\ell}^k(\theta)$

$\hat{\theta}^{k+1} \approx \arg\max_\theta \ell^k(\theta)$

The meta-approach:

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

1. Init π_0
2. For $k = 0, \ldots K$:

$$\pi^{k+1} \approx \arg\max_{\theta} \Delta_k(\pi^\theta),$$

such that ρ_θ is "close" to ρ_{θ^k}

where $\Delta_k(\pi) = \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h\sim\pi(s_h)} A^{\pi^k}(s_h, a_h)\right]$

GP.

$$\Delta_k(\pi^\theta) = \nabla J(\theta^k) \cdot (\theta - \theta^k)$$

s.t. $\|\theta - \theta^k\| \le \delta$

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

1. Init π_0
2. For $k = 0, \ldots K$:

$\pi^{k+1} \approx \arg\max_{\theta} \Delta_k(\pi^\theta)$, where $\Delta_k(\pi) = \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi(s_h)} A^{\pi^k}(s_h, a_h) \right]$

such that ρ_θ is "close" to ρ_{θ^k}

- CPI: conservative policy iteration
  uses unconstrained optimization: $\widetilde{\pi} \approx \arg\max_{\theta} \Delta_k(\pi^\theta)$,
  enforces closeness with "mixing": $\pi^{k+1} = (1-\alpha) \cdot \pi^k + \alpha \cdot \widetilde{\pi}^{\cancel{k+1}}$

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

1. Init π_0
2. For $k = 0, \ldots K$:

$\pi^{k+1} \approx \arg\max_{\theta} \Delta_k(\pi^{\theta})$, where $\Delta_k(\pi) = \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi(s_h)} A^{\pi^k}(s_h, a_h) \right]$

such that ρ_θ is "close" to ρ_{θ^k}

- CPI: conservative policy iteration
  uses unconstrained optimization: $\widetilde{\pi} \approx \arg\max_{\theta} \Delta_k(\pi^{\theta})$,
  enforces closeness with "mixing": $\pi^{k+1} = (1-\alpha) \cdot \pi^k + \alpha \cdot \widetilde{\pi}^{k+1}$
- TRPO: use KL to enforce closeness.

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

1. Init π_0
2. For $k = 0,\ldots K$:

$$\pi^{k+1} \approx \arg\max_{\theta} \Delta_k(\pi^\theta), \qquad \text{where } \Delta_k(\pi) = \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h\sim\pi(s_h)} A^{\pi^k}(s_h, a_h)\right]$$

such that ρ_θ is "close" to ρ_{θ^k}

- CPI: conservative policy iteration
  uses unconstrained optimization: $\widetilde{\pi} \approx \arg\max_{\theta} \Delta_k(\pi^\theta)$,
  enforces closeness with "mixing": $\pi^{k+1} = (1-\alpha)\cdot\pi^k + \alpha\cdot\widetilde{\pi}^{k+1}$
- TRPO: use KL to enforce closeness.
- NPG: is TRPO up to "leading order" (via Taylor's theorem).

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

1. Init π_0
2. For $k = 0,\ldots K$:

$$\pi^{k+1} \approx \arg\max_{\theta} \Delta_k(\pi^{\theta}),$$

where $\Delta_k(\pi) = \mathbb{E}_{s_0,\ldots s_{H-1}\sim\rho_{\pi^k}}\left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h\sim\pi(s_h)} A^{\pi^k}(s_h, a_h)\right]$

such that ρ_θ is "close" to ρ_{θ^k}

- CPI: conservative policy iteration
  uses unconstrained optimization: $\widetilde{\pi} \approx \arg\max_{\theta} \Delta_k(\pi^{\theta}),$
  enforces closeness with "mixing": $\pi^{k+1} = (1-\alpha)\cdot\pi^k + \alpha\cdot\widetilde{\pi}^{k+1}$
- TRPO: use KL to enforce closeness.
- NPG: is TRPO up to "leading order" (via Taylor's theorem).
- PPO: uses a Lagrangian relaxation (i.e. regularization)

Meta-Approach: CPI/TRPO/NPG/PPO are all pretty similar.

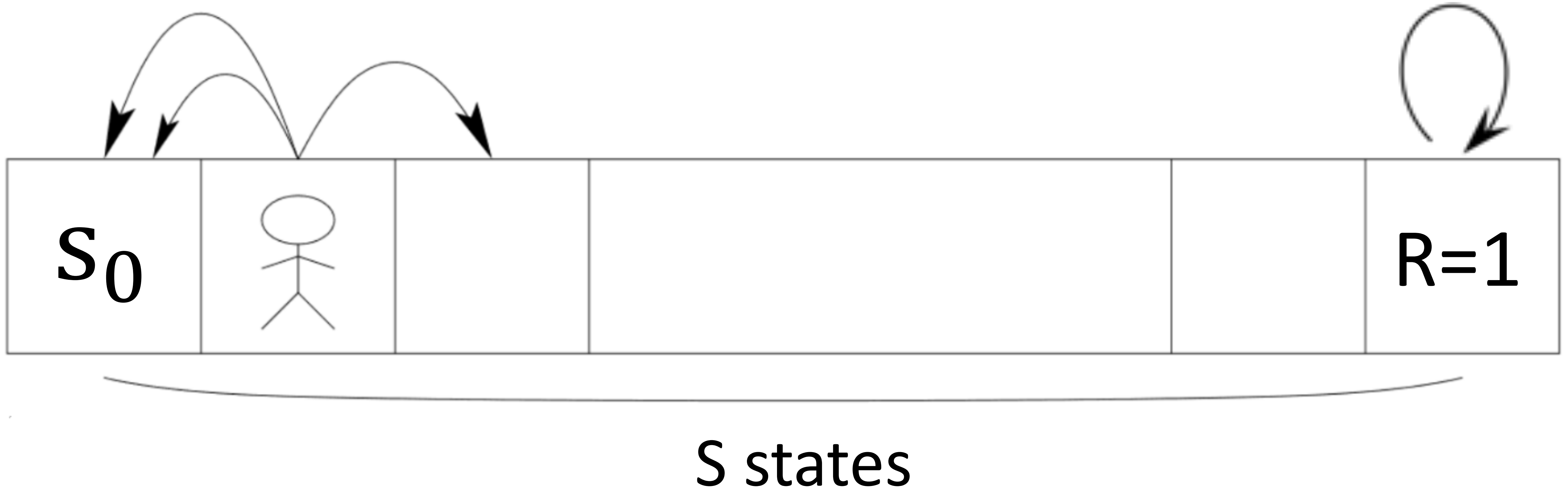
1. Init π_0
2. For $k = 0, \ldots K$:

   $\pi^{k+1} \approx \arg\max_{\theta} \Delta_k(\pi^{\theta})$, where $\Delta_k(\pi) = \mathbb{E}_{s_0,\ldots s_{H-1} \sim \rho_{\pi^k}} \left[\sum_{h=0}^{H-1} \mathbb{E}_{a_h \sim \pi(s_h)} A^{\pi^k}(s_h, a_h) \right]$

   such that ρ_θ is "close" to ρ_{θ^k}

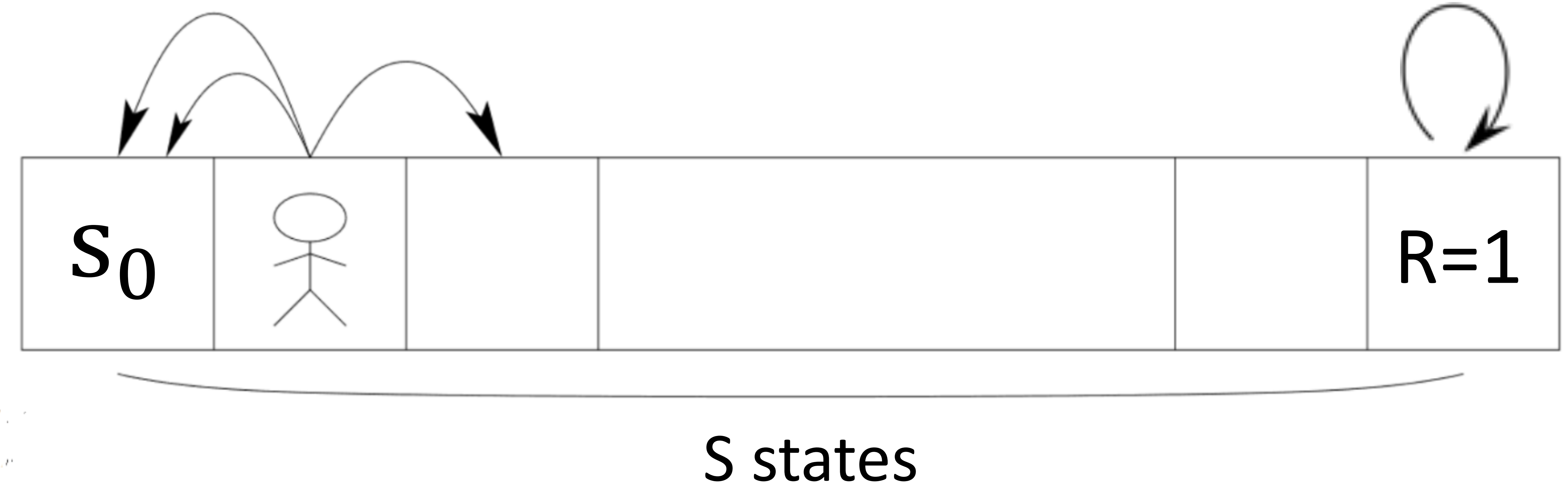
   - CPI: conservative policy iteration
     uses unconstrained optimization: $\widetilde{\pi} \approx \arg\max_{\theta} \Delta_k(\pi^{\theta})$,
     enforces closeness with "mixing": $\pi^{k+1} = (1-\alpha) \cdot \pi^k + \alpha \cdot \widetilde{\pi}^{k+1}$
   - TRPO: use KL to enforce closeness.
   - NPG: is TRPO up to "leading order" (via Taylor's theorem).
   - PPO: uses a Lagrangian relaxation (i.e. regularization)
3. Return π_K

“Lack of Exploration” leads to Optimization and Statistical Challenges



Thrun ‘92

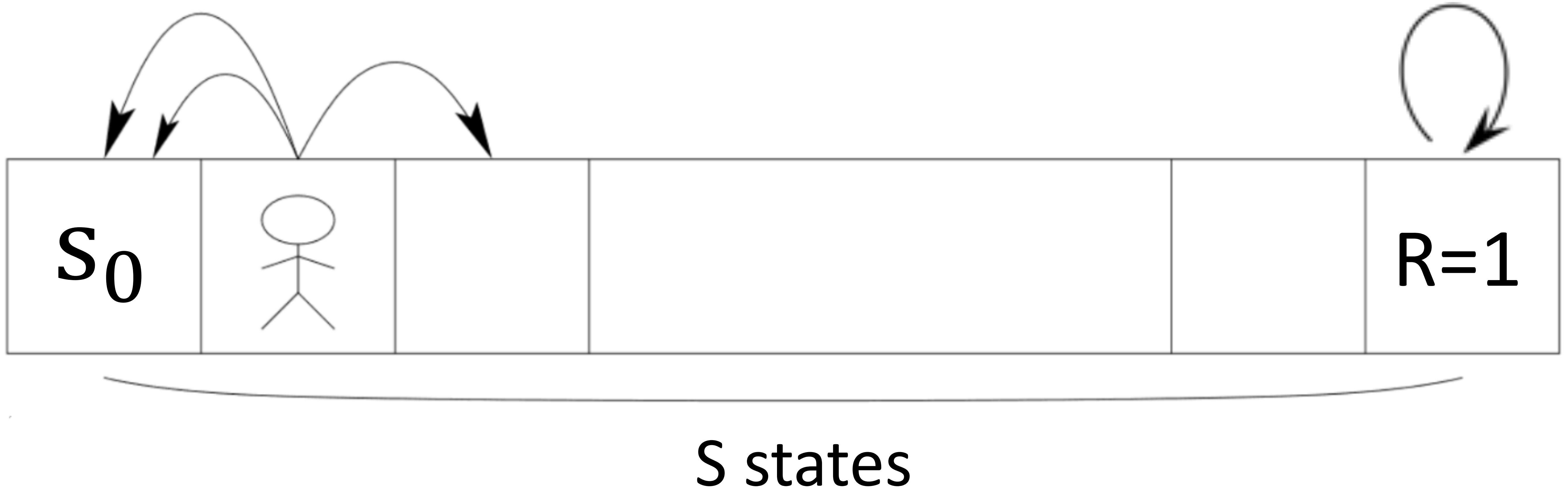
“Lack of Exploration” leads to Optimization and Statistical Challenges



Thrun ‘92

- Suppose $|S| \approx H$ & $\mu(s_0) = 1$ (i.e. we start at s_0).

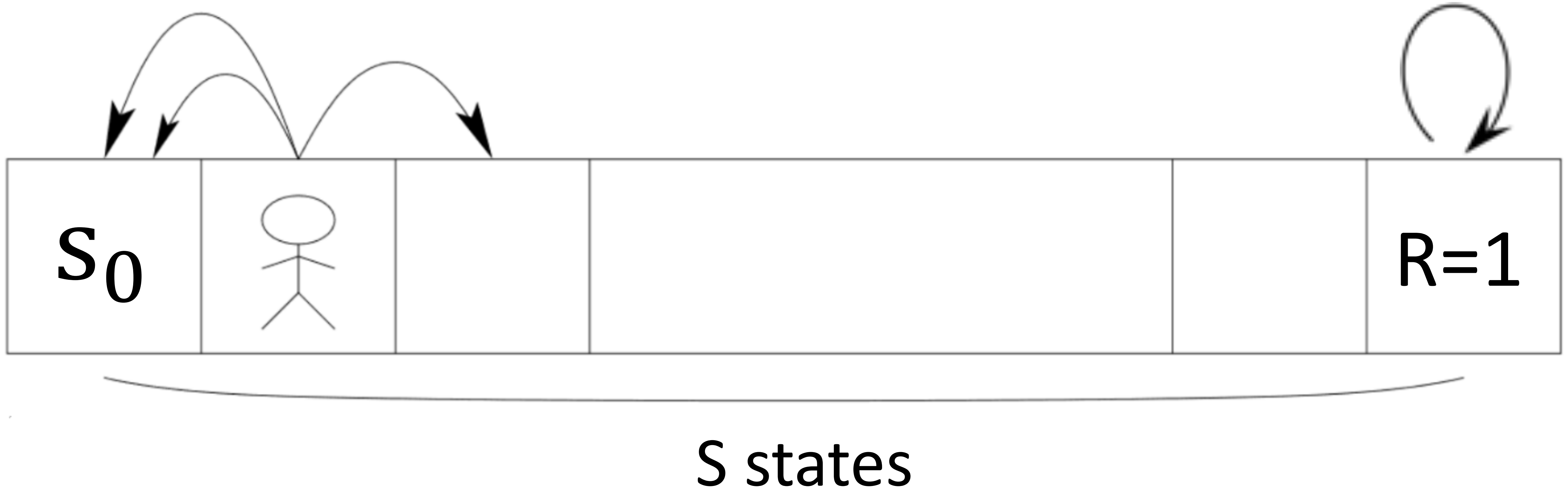
"Lack of Exploration" leads to Optimization and Statistical Challenges



Thrun '92

- Suppose $|S| \approx H$ & $\mu(s_0) = 1$ (i.e. we start at s_0).
- A randomly initialized policy π^0 has prob. $O(1/3^H)$ of hitting the goal state in a trajectory.

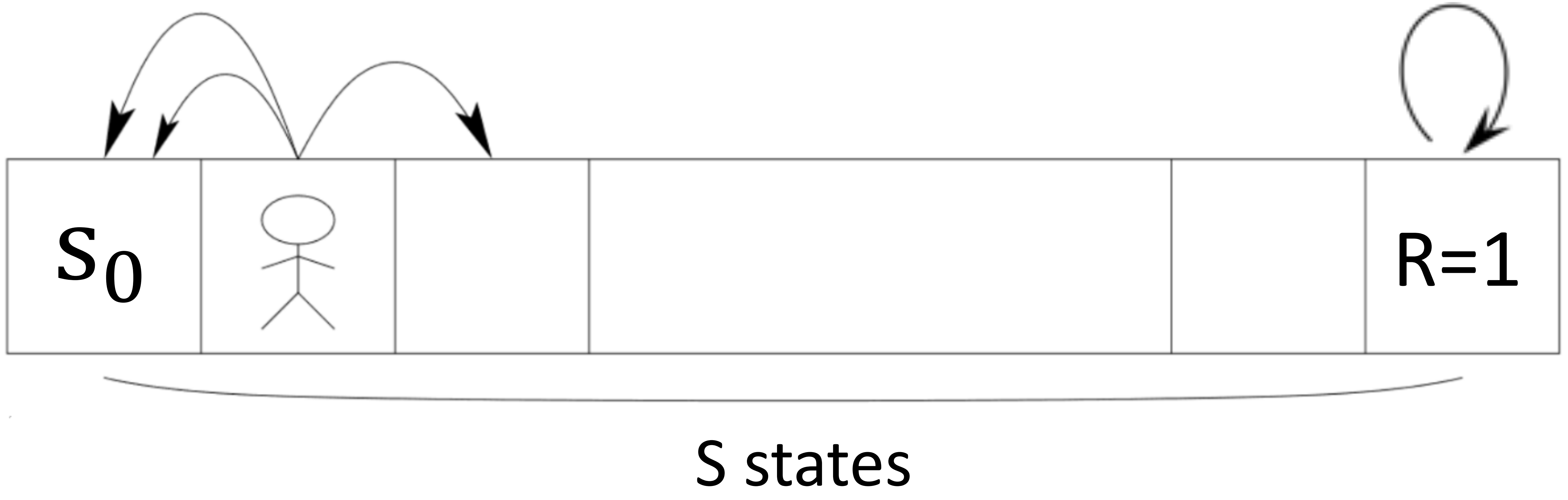
“Lack of Exploration” leads to Optimization and Statistical Challenges



Thrun ‘92

- Suppose $|S| \approx H$ & $\mu(s_0) = 1$ (i.e. we start at s_0).
- A randomly initialized policy π^0 has prob. $O(1/3^H)$ of hitting the goal state in a trajectory.
- Implications:
  - The following sample based approach, with $\mu(s_0) = 1$, require $O(3^H)$ trajectories.
    - Holds for (sample based) Fitted DP
    - Holds for (sample based) PG/CPI/TRPO/NPG/PPO

"Lack of Exploration" leads to Optimization and Statistical Challenges

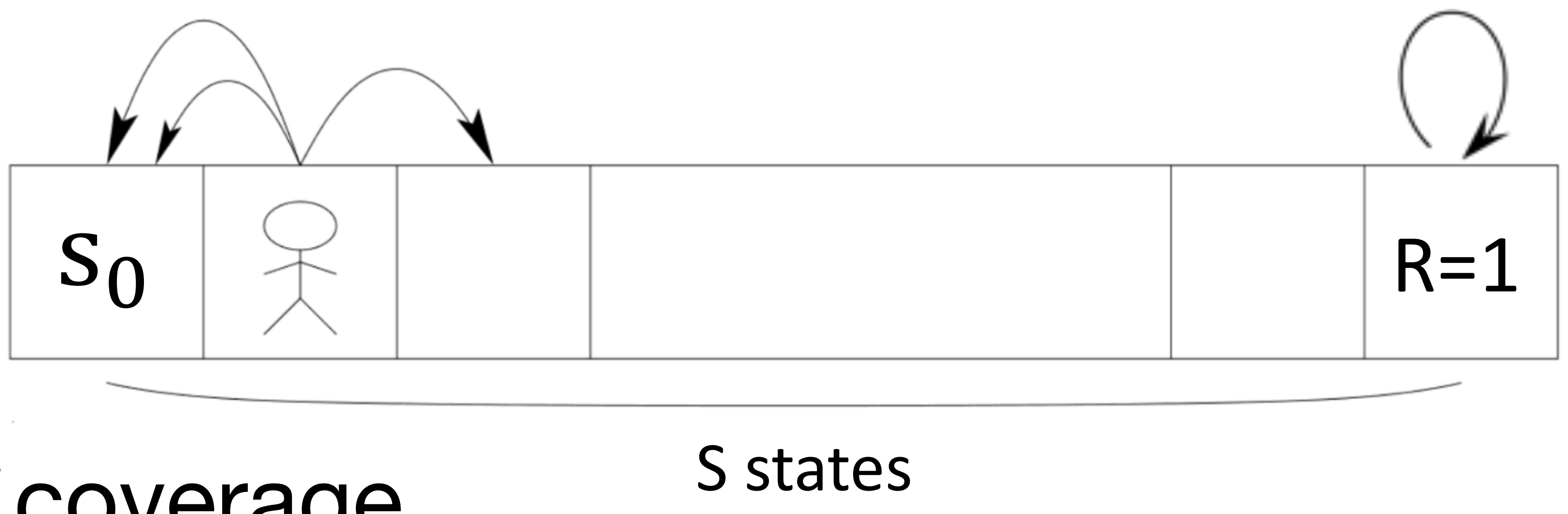


Thrun '92

- Suppose $|S| \approx H$ & $\mu(s_0) = 1$ (i.e. we start at s_0).
- A randomly initialized policy π^0 has prob. $O(1/3^H)$ of hitting the goal state in a trajectory.
- Implications:
  - The following sample based approach, with $\mu(s_0) = 1$, require $O(3^H)$ trajectories.
    - Holds for (sample based) Fitted DP
    - Holds for (sample based) PG/CPI/TRPO/NPG/PPO
- Basically, for these approaches, we are stuck without exploration, if $\mu(s_0) = 1$.

Let's examine the role of μ

Let's examine the role of μ



Thrun '92

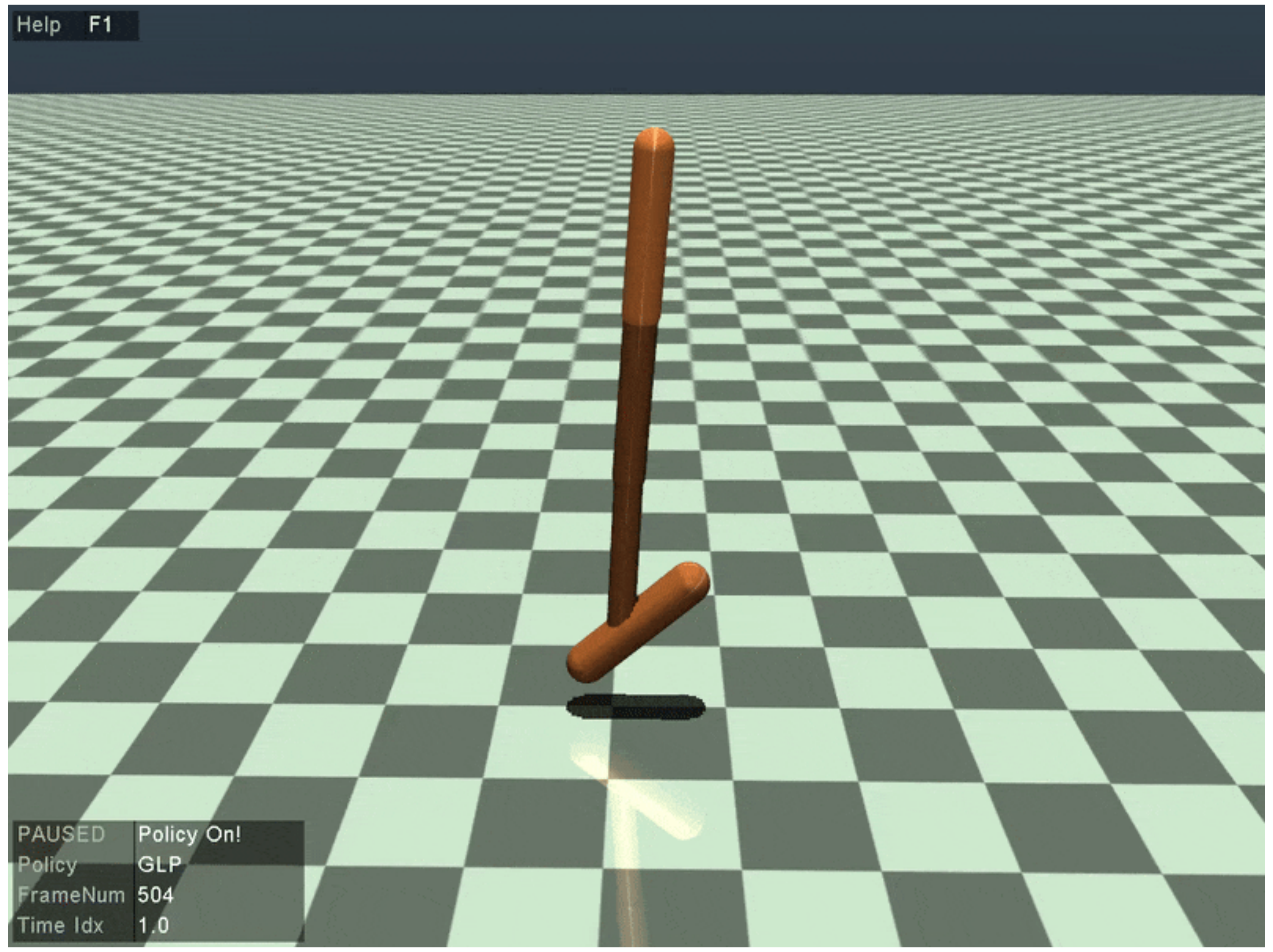
- Suppose that somehow the distribution μ had better coverage.
  - e.g, μ was uniform over the all states in our toy problem, then all approaches we covered would work (with mild assumptions)
  - Theory: CPI/TRPO/NPG/PPO have better guarantees than fitted DP methods (assuming some "coverage")

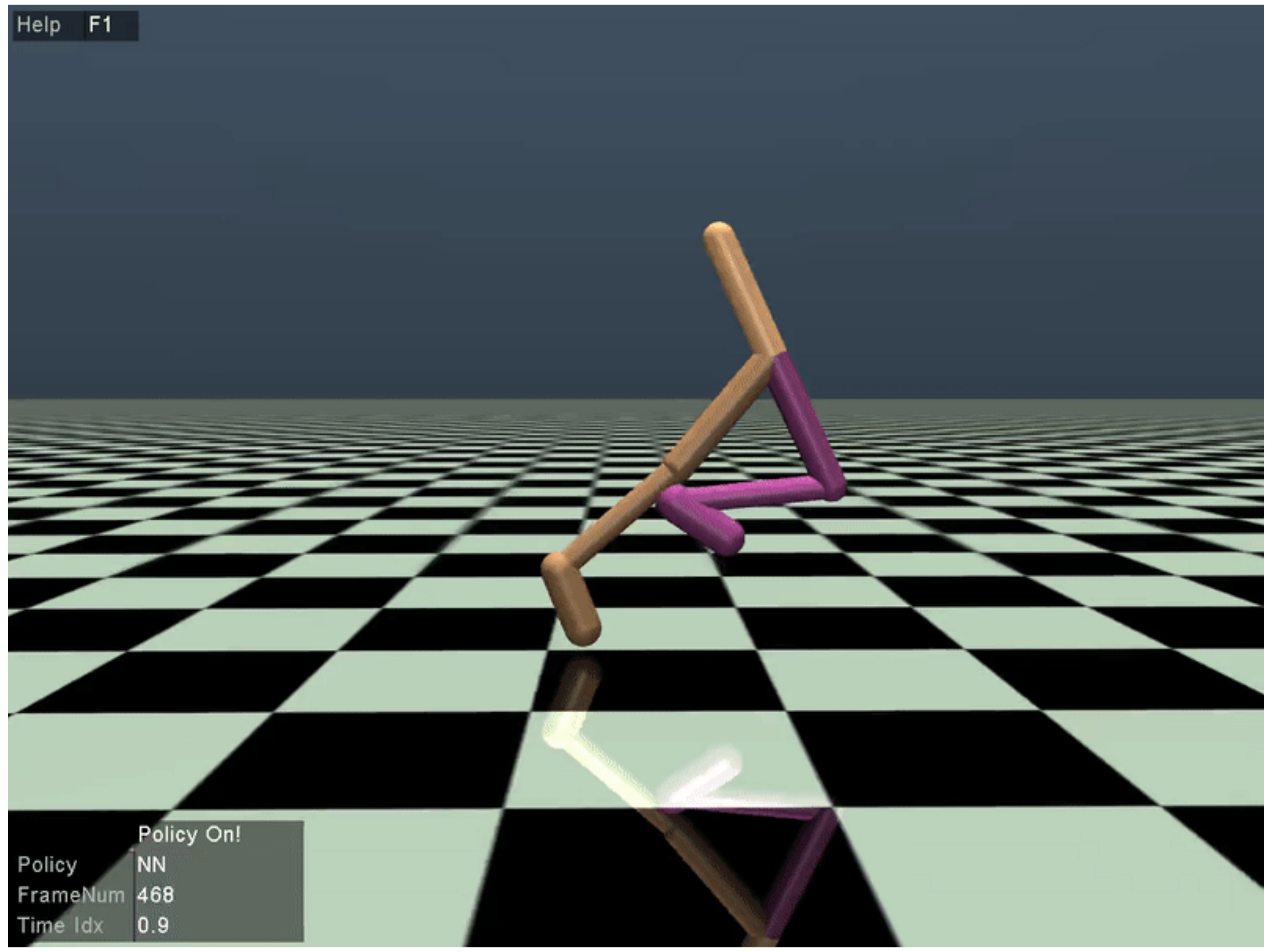
Let's examine the role of μ

Thrun '92

- Suppose that somehow the distribution μ had better coverage.
  - e.g, μ was uniform over the all states in our toy problem, then all approaches we covered would work (with mild assumptions)
  - Theory: CPI/TRPO/NPG/PPO have better guarantees than fitted DP methods (assuming some "coverage")
- Strategies without coverage:
  - If we have a simulator, sometimes we can design μ to have better coverage.
    - this is helpful for robustness as well.
  - Imitation learning (next time).
    - An expert gives us samples from a "good" μ.
  - Explicit exploration:
    - UCB-VI: we'll merge two good ideas!
    - Encourage exploration in PG methods.
  - Try with reward shaping

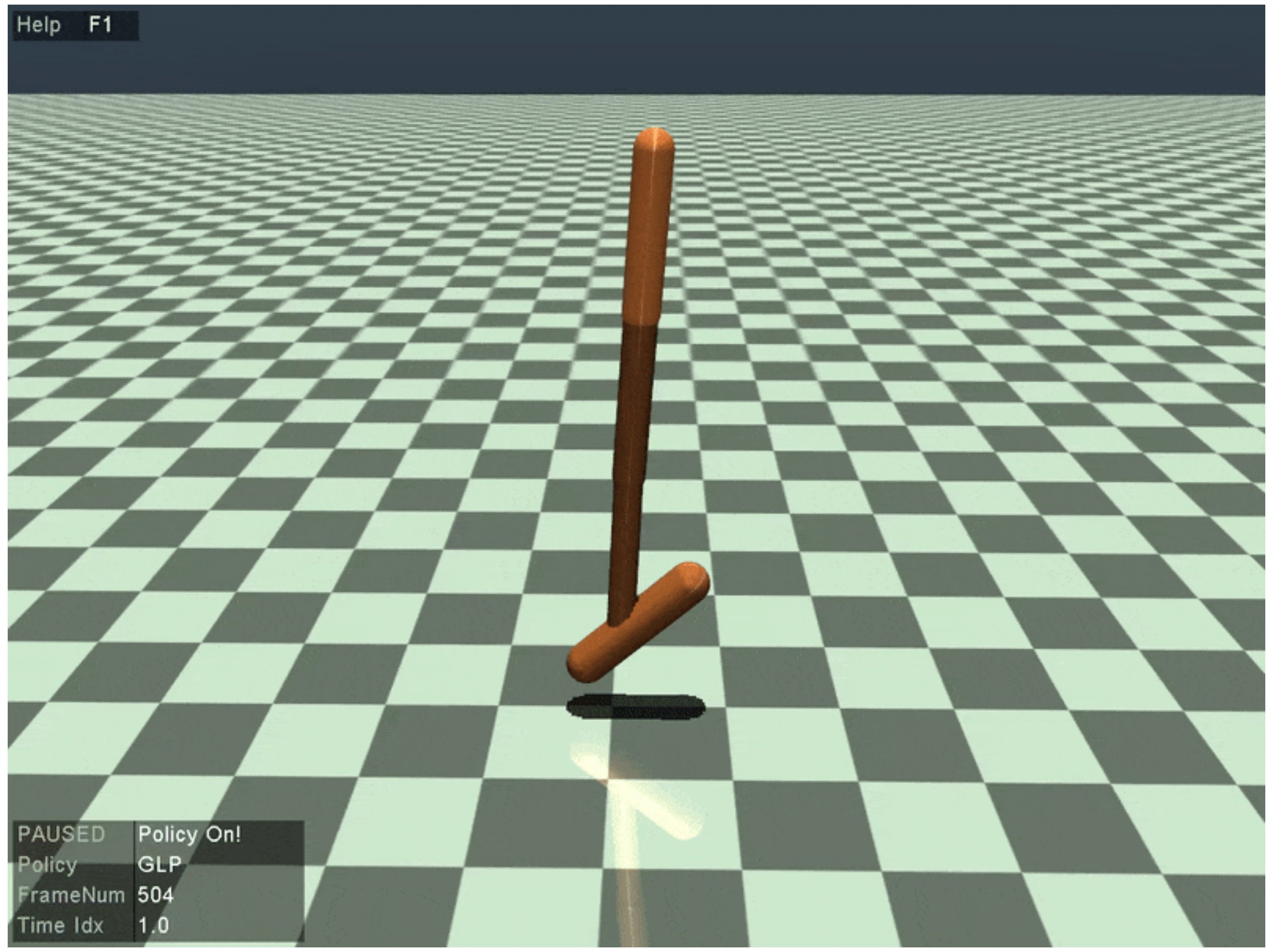
Aside: Brittle policies if we train starting from only from one configuration!

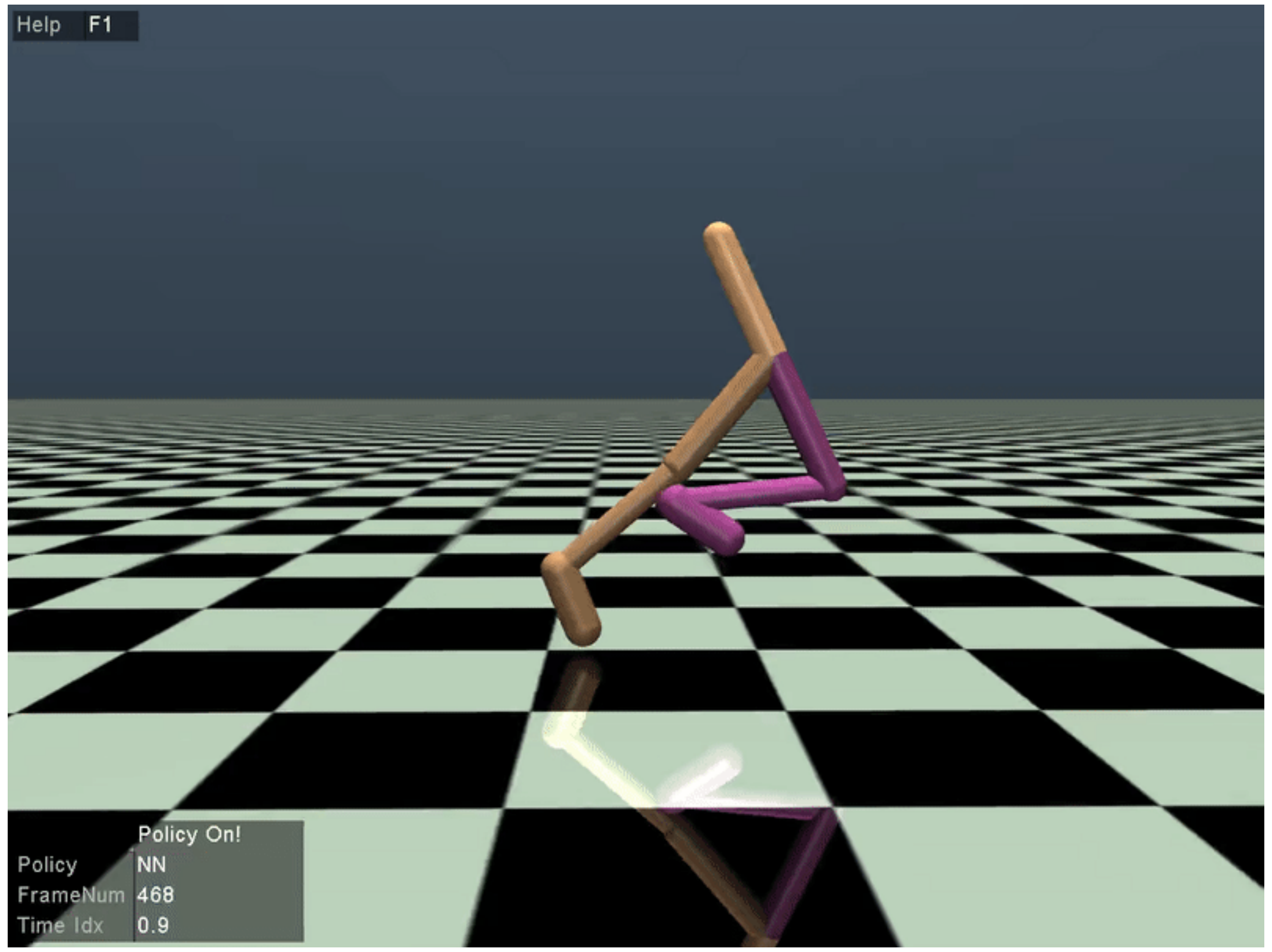




- [Rajeswaran, Lowrey, Todorov, K. 2017]: showed policies optimized for a single starting configuration s_0 are not robust!

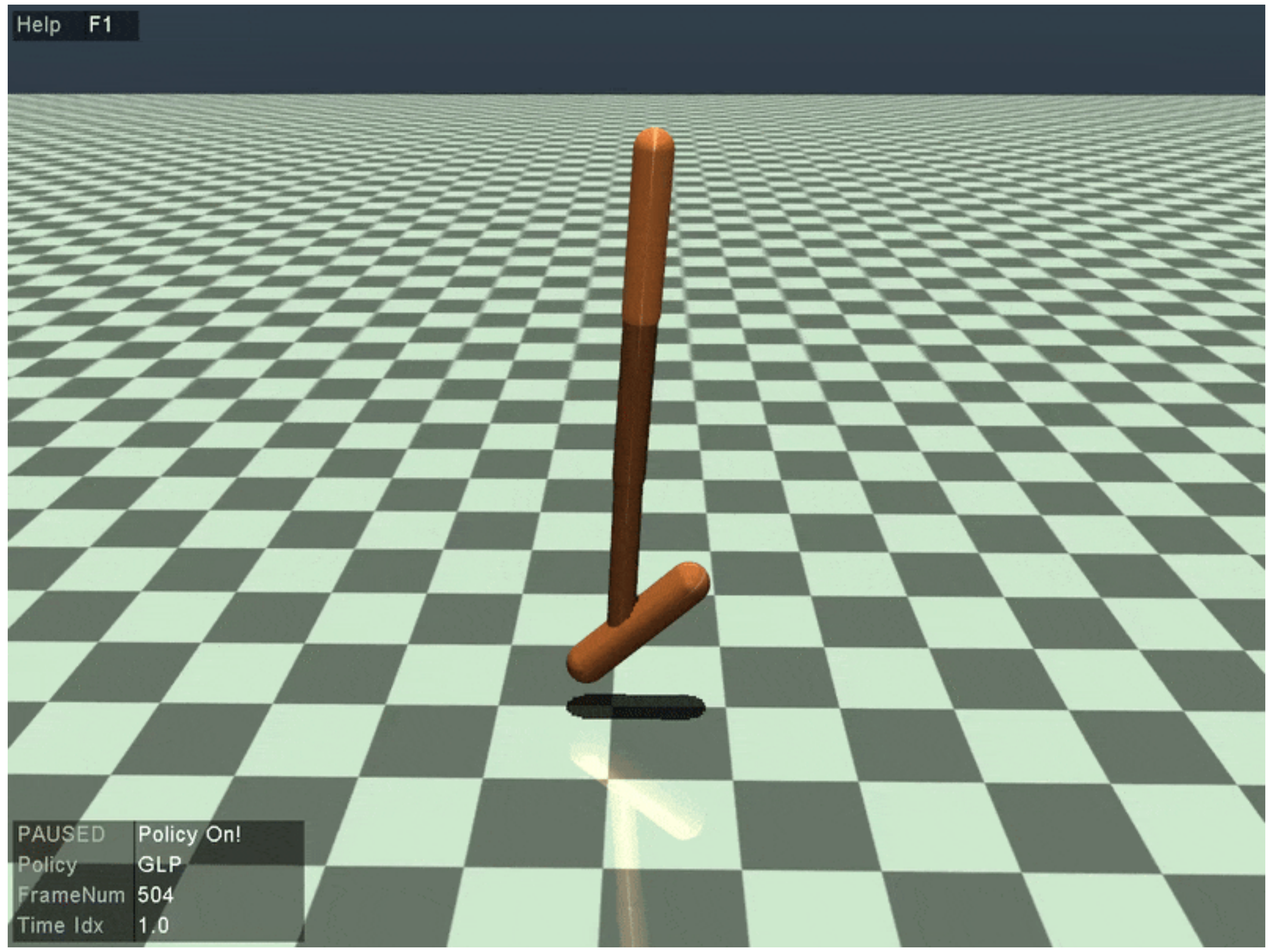
Aside: Brittle policies if we train starting from only from one configuration!

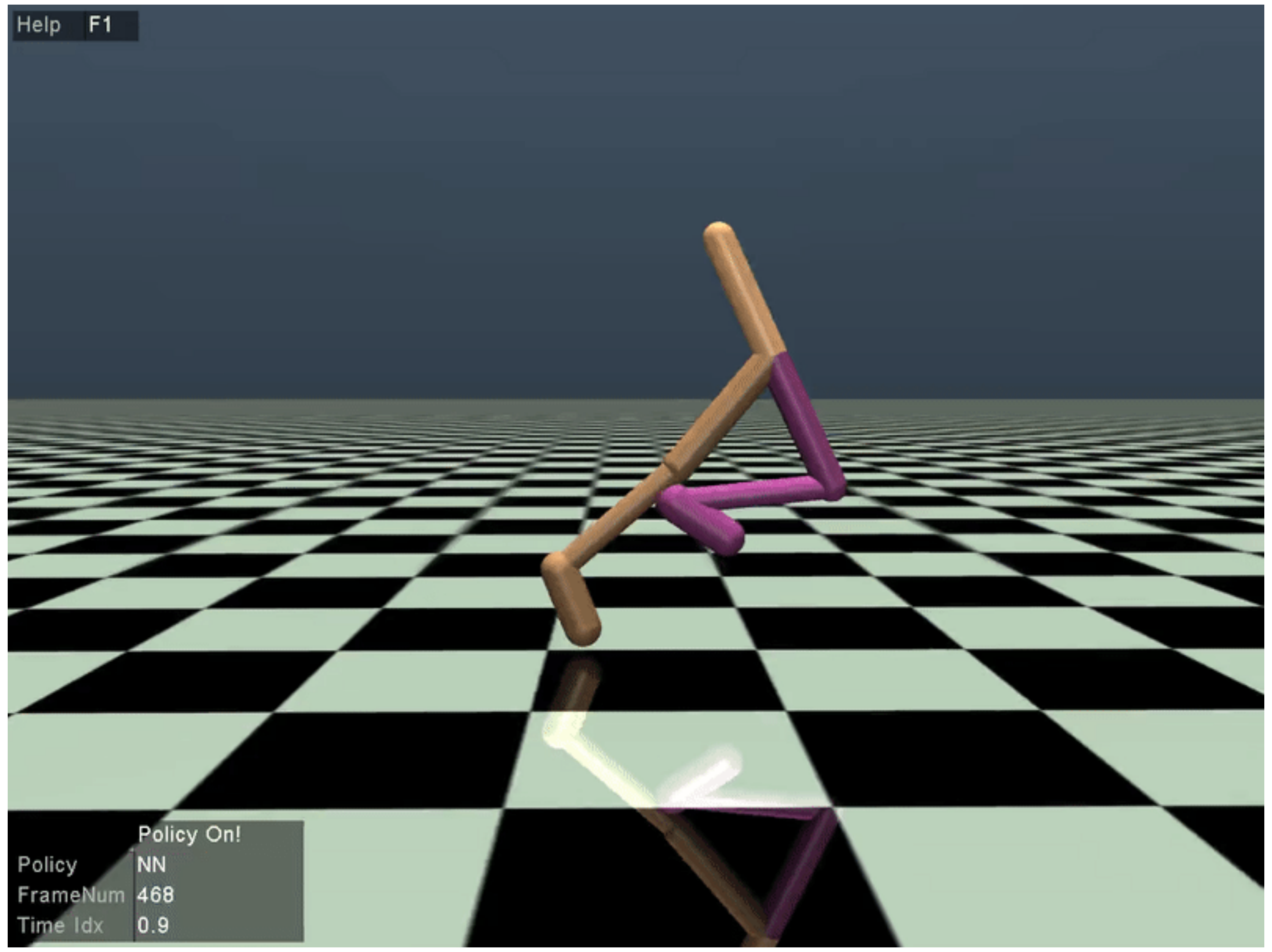




- [Rajeswaran, Lowrey, Todorov, K. 2017]: showed policies optimized for a single starting configuration s_0 are not robust!

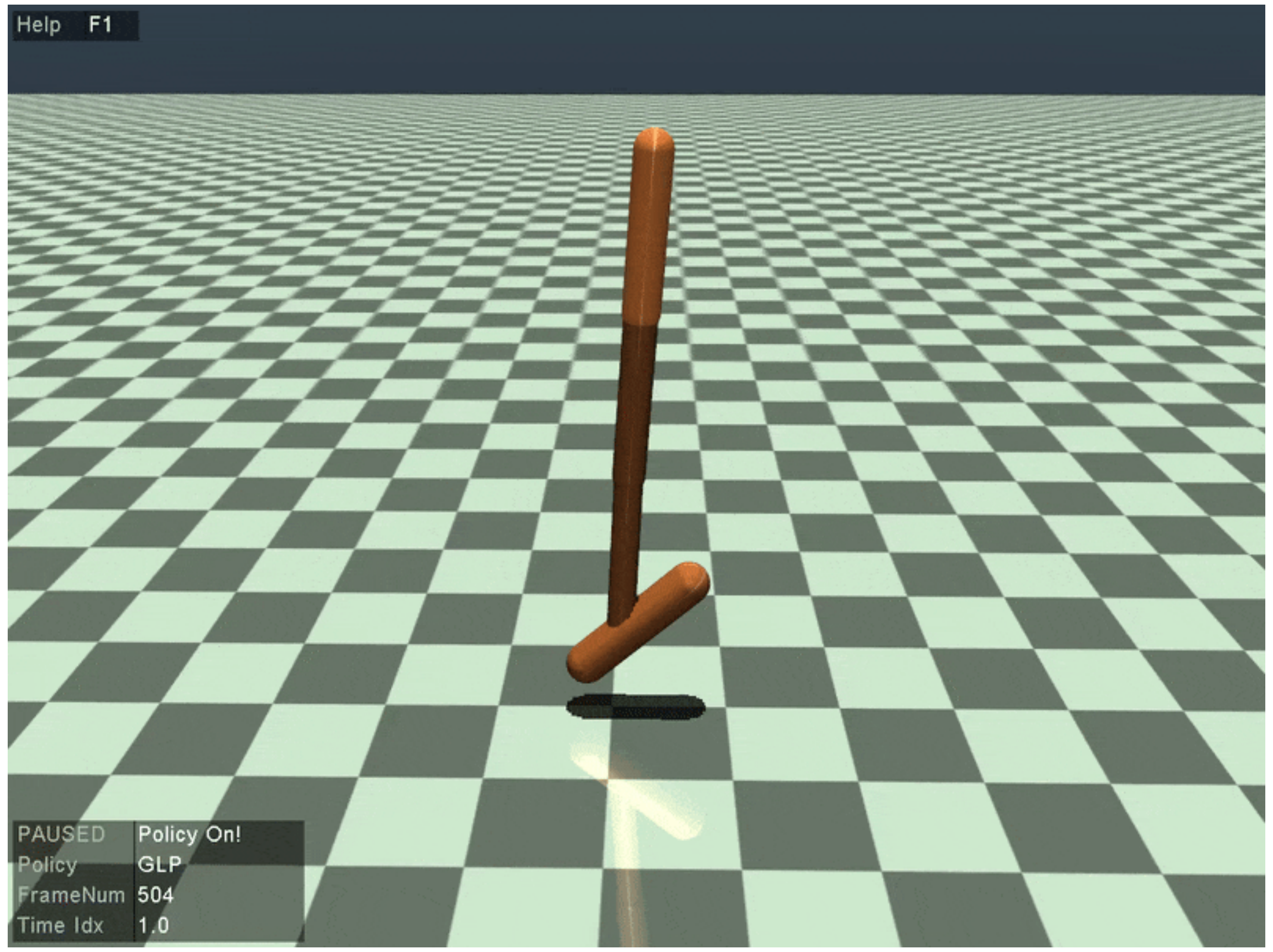
Aside: Brittle policies if we train starting from only from one configuration!

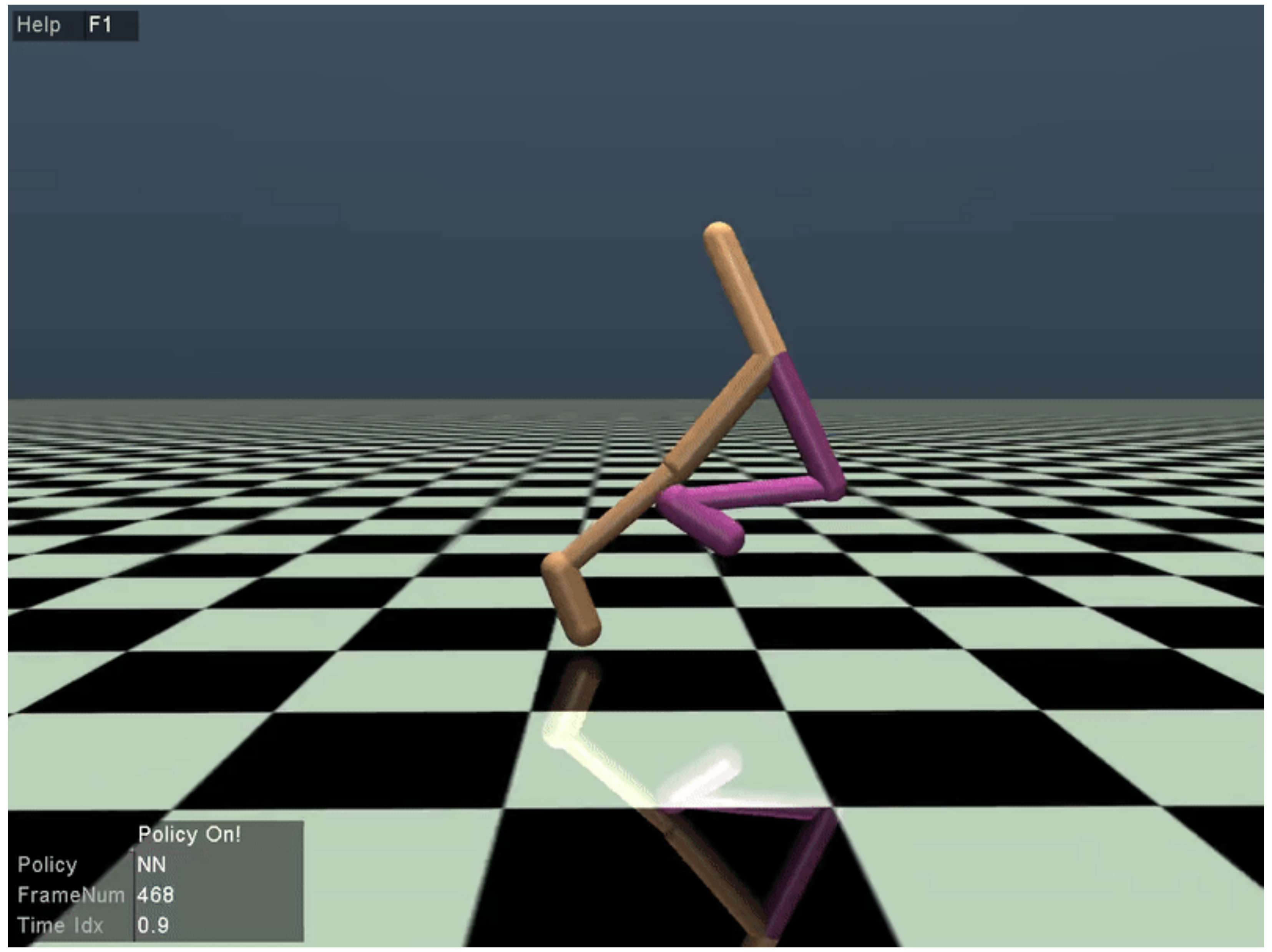




- [Rajeswaran, Lowrey, Todorov, K. 2017]: showed policies optimized for a single starting configuration s_0 are not robust!

Aside: Brittle policies if we train starting from only from one configuration!





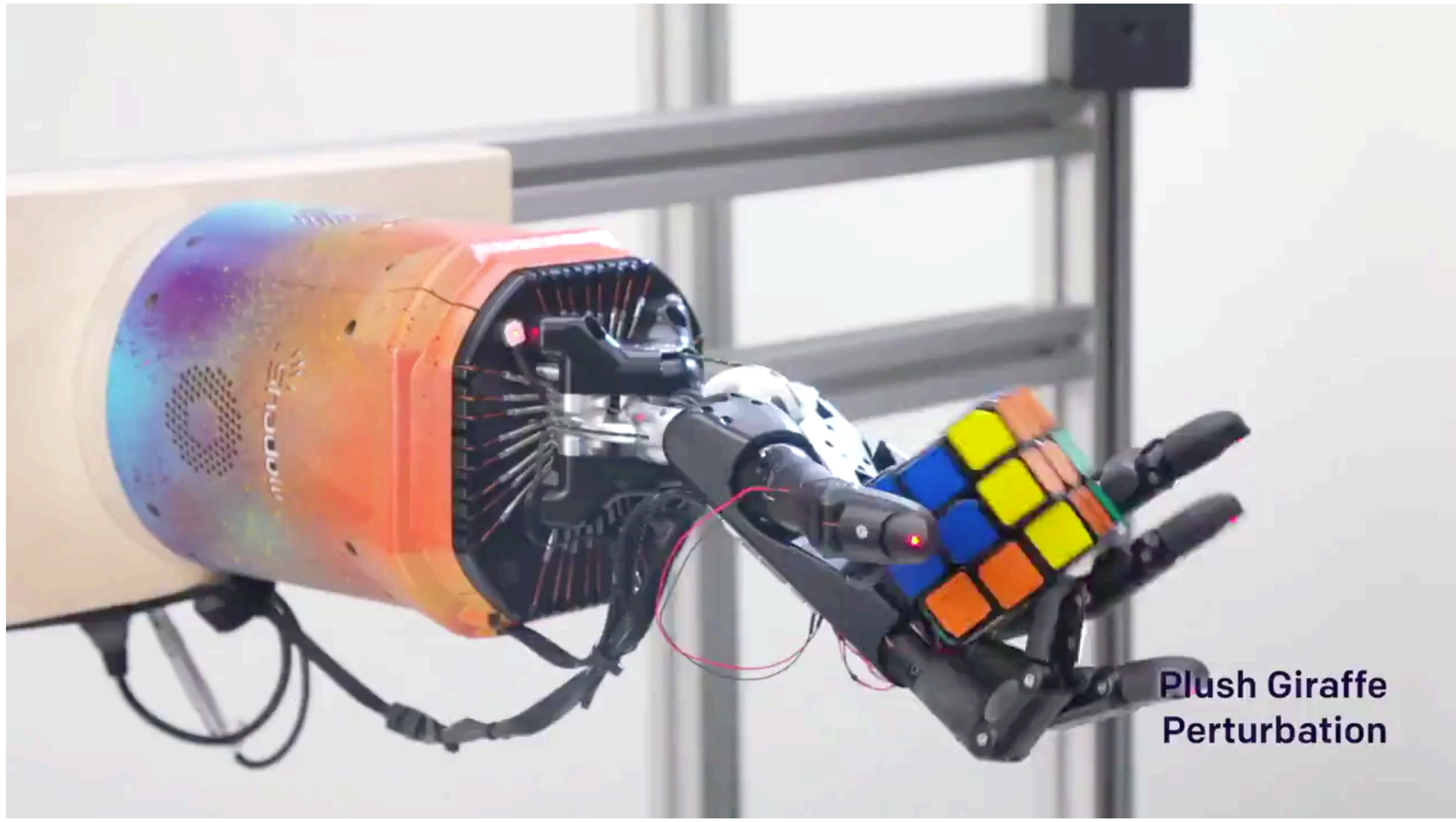
- [Rajeswaran, Lowrey, Todorov, K. 2017]: showed policies optimized for a single starting configuration s_0 are not robust!
- How to fix this?
  - Training from different starting configurations sampled from $s_0 \sim \mu$ fixes this.

$$\max_{\theta} E_{s_0 \sim \mu}[V^{\theta}(s_0)]$$

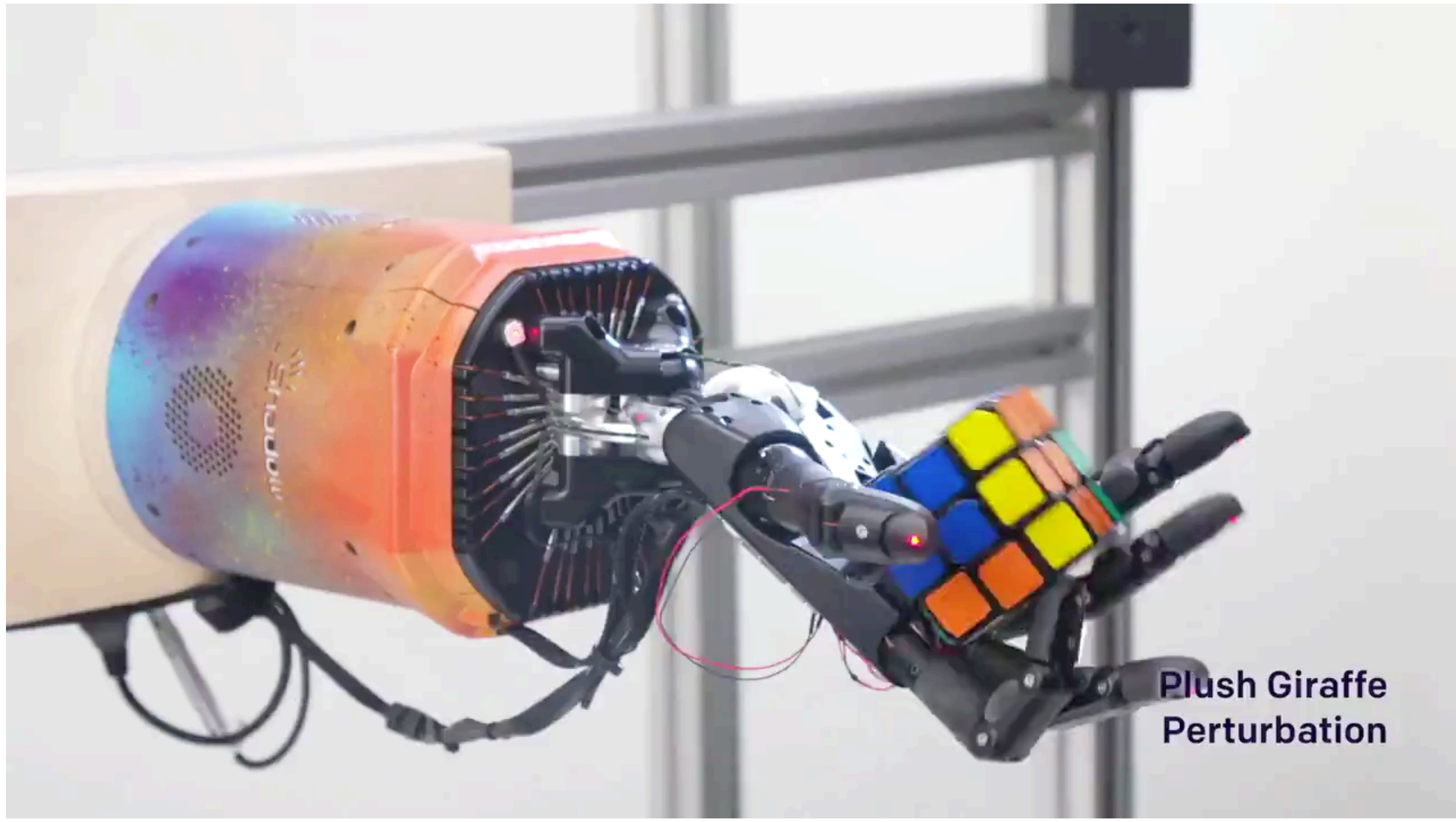
  - The measure μ is also relevant for robustness.

OpenAI: progress on dexterous hand manipulation

OpenAI: progress on dexterous hand manipulation



OpenAI: progress on dexterous hand manipulation



Trained with “domain randomization”

Basically, the measure $s_0 \sim \mu$ was diverse.

What about guarantees?

What about guarantees?

- The hope is that (average case) “supervised learning” works, then RL would also work.

What about guarantees?

- The hope is that (average case) “supervised learning” works, then RL would also work.

[Theorem:] (informal, see AJKS Ch 4+13) for log linear policies,

What about guarantees?

- The hope is that (average case) “supervised learning” works, then RL would also work.

[Theorem:] (informal, see AJKS Ch 4+13) for log linear policies,

- Approximation error: For all policies, suppose that for all π,

$$\min_{\theta} \mathbb{E}_{\tau \sim \rho_{\pi_\pi}} \left[\frac{1}{H} \sum_{h=0}^{H-1} \left(Q^\pi(s_h, a_h) - \theta^\top \phi(s_h, a_h) \right)^2 \right] \leq \delta$$

What about guarantees?

- The hope is that (average case) “supervised learning” works, then RL would also work.

[Theorem:] (informal, see AJKS Ch 4+13) for log linear policies,

- Approximation error: For all policies, suppose that for all π,

$$\min_{\theta} \mathbb{E}_{\tau \sim \rho_{\pi_\pi}} \left[\frac{1}{H} \sum_{h=0}^{H-1} \left(Q^\pi(s_h, a_h) - \theta^\top \phi(s_h, a_h) \right)^2 \right] \le \delta$$

- Sample size: Suppose that we use a # samples that is poly in d & $1/\epsilon_{stat}$ for both fittedPI and NPG.

What about guarantees?

- The hope is that (average case) “supervised learning” works, then RL would also work.

[Theorem:] (informal, see AJKS Ch 4+13) for log linear policies,

- Approximation error: For all policies, suppose that for all π,

$$\min_{\theta} \mathbb{E}_{\tau \sim \rho_{\pi_\pi}} \left[\frac{1}{H} \sum_{h=0}^{H-1} \left(Q^\pi(s_h, a_h) - \theta^\top \phi(s_h, a_h) \right)^2 \right] \leq \delta$$

- Sample size: Suppose that we use a # samples that is poly in d & $1/\epsilon_{stat}$ for both fittedPI and NPG.
- Coverage: suppose that μ has coverage over the marginal distribution $\rho_{\pi^\star}(s \mid h)$:

$$\max_{s,h} \left(\frac{\rho_{\pi^\star}(s \mid h)}{\mu(s)} \right) \leq C$$

What about guarantees?

- The hope is that (average case) "supervised learning" works, then RL would also work.

[Theorem:] (informal, see AJKS Ch 4+13) for log linear policies,

- Approximation error: For all policies, suppose that for all π,

$$\min_{\theta} \mathbb{E}_{\tau \sim \rho_{\pi_\pi}} \left[\frac{1}{H} \sum_{h=0}^{H-1} \left(Q^{\pi}(s_h, a_h) - \theta^{\top} \phi(s_h, a_h) \right)^2 \right] \leq \delta$$

- Sample size: Suppose that we use a # samples that is poly in d & $1/\epsilon_{stat}$ for both fittedPI and NPG.
- Coverage: suppose that μ has coverage over the marginal distribution $\rho_{\pi^\star}(s \mid h)$:

$$\max_{s,h} \left(\frac{\rho_{\pi^\star}(s \mid h)}{\mu(s)} \right) \leq C$$

- Computation: suppose we run NPG with poly in $1/\epsilon_{stat}$, H iterations.

What about guarantees?

- The hope is that (average case) "supervised learning" works, then RL would also work.

[Theorem:] (informal, see AJKS Ch 4+13) for log linear policies,

- Approximation error: For all policies, suppose that for all π,

$$\min_{\theta} \mathbb{E}_{\tau \sim \rho_{\pi_\pi}} \left[\frac{1}{H} \sum_{h=0}^{H-1} \left(Q^\pi(s_h, a_h) - \theta^\top \phi(s_h, a_h) \right)^2 \right] \leq \delta$$

- Sample size: Suppose that we use a # samples that is poly in d & $1/\epsilon_{stat}$ for both fittedPI and NPG.
- Coverage: suppose that μ has coverage over the marginal distribution $\rho_{\pi^\star}(s \mid h)$:

$$\max_{s,h} \left(\frac{\rho_{\pi^\star}(s \mid h)}{\mu(s)} \right) \leq C$$

- Computation: suppose we run NPG with poly in $1/\epsilon_{stat}$, H iterations.
- **Theorem**: NPG will return a policy with sub-optimality determined by C and the average case error δ:

$$J(\hat{\pi}) \geq J(\pi^\star) - \epsilon_{stat} - 2H^2 C\delta$$

Summary:

1. NPG: a simpler way to do TRPO, a “pre-conditioned” gradient method.
2. PPO: “first order” approx to TRPO

Attendance:

bit.ly/3RcTC9T

Feedback:

bit.ly/3RHtlxy