Peg Insertion with Policy Search

Caris Moses

6.231 Final Project

Motivation

- Policy search provide a **generalizable method** to learning parameters for polices.

Motivation

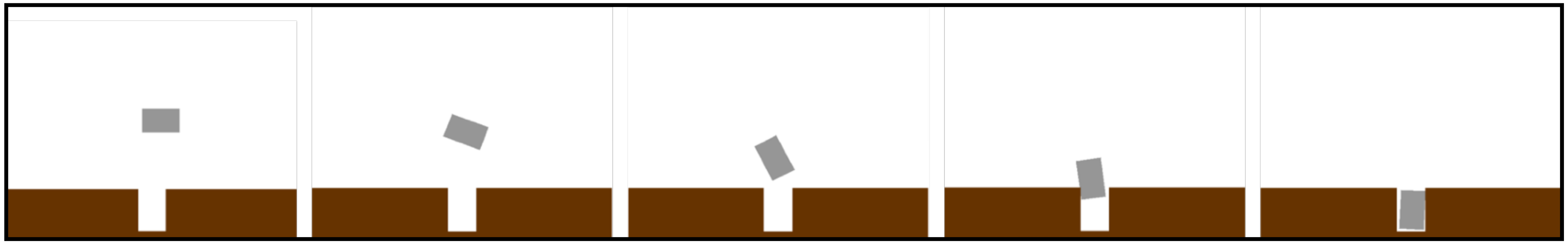
- Policy search provide a **generalizable method** to learning parameters for polices.
- The methods considered in this talk are **model-free** and therefore require no knowledge of system dynamics.

Motivation

- Policy search provide a **generalizable method** to learning parameters for polices.

- The methods considered in this talk are **model-free** and therefore require no knowledge of system dynamics.

- Peg insertion is a know **difficult task** for stochastic robot processes.



CSAIL

Policy Parameterizations

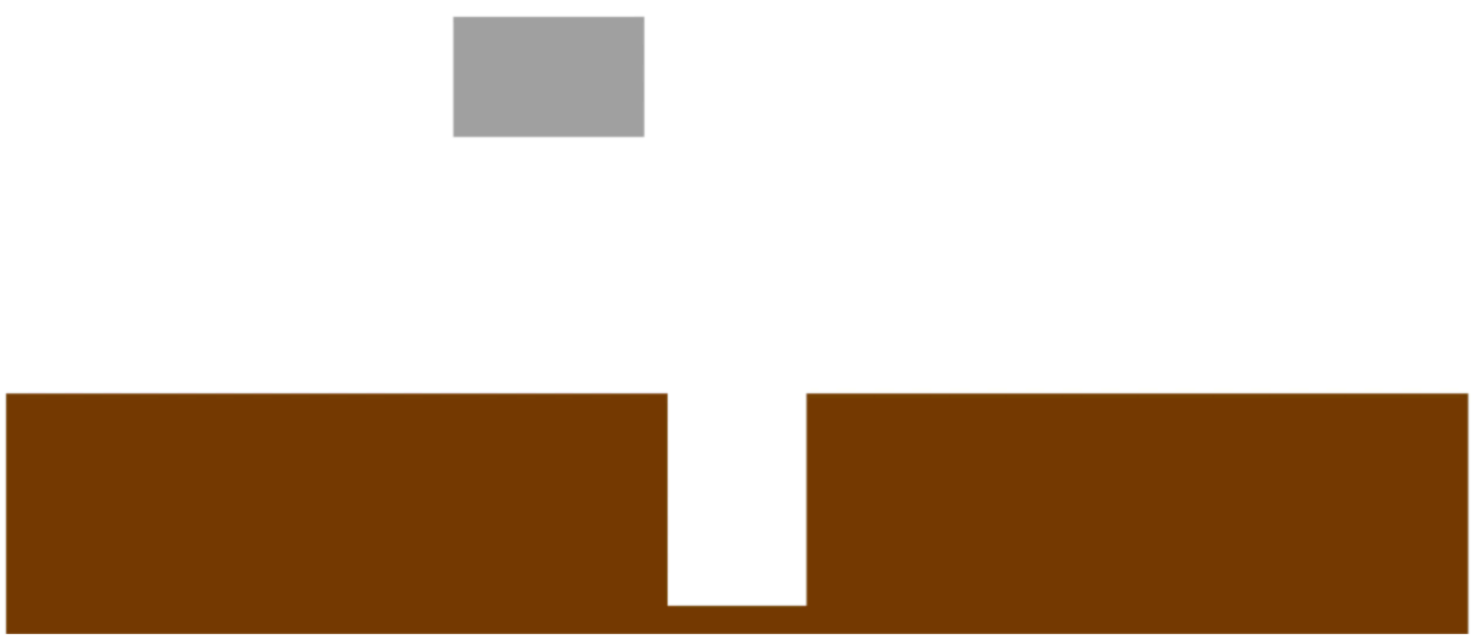
$$\pi(a|s;\theta) = ?$$

Policy Parameterizations
Dynamic Movement Primitives

- Attractor Dynamics (PD controller) for 1D state

$$\boxed{\pi(a|s, v_s) = -K(s - g_s) - Dv_s}$$

$K = 7$
$D = .5$



CSAIL

Policy Parameterizations

Dynamic Movement Primitives[2]

- Attractor Dynamics (PD controller with parameterized forcing term) for 1D state

$$\boxed{\pi(a|s, v_s) = -K(s - g_s) - Dv_s + \mathbf{b}^T \boldsymbol{\theta}}$$

CSAIL

Policy Parameterizations

Dynamic Movement Primitives[2]

- Attractor Dynamics (PD controller with parameterized forcing term) for 1D state

$$\boxed{\pi(a|s, v_s) = -K(s - g_s) - Dv_s + \mathbf{b}^T \boldsymbol{\theta}}$$

- Canonical System Dynamics

$$\boxed{\dot{z}_t = -\alpha z_t} \qquad \boxed{z_0 = 1}$$

CSAIL

Policy Parameterizations
Dynamic Movement Primitives[2]

- Attractor Dynamics (PD controller with parameterized forcing term) for 1D state

$$\boxed{\pi(a|s, v_s) = -K(s - g_s) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}}$$

- Canonical System Dynamics

$$\boxed{\dot{z}_t = -\alpha z_t} \qquad \boxed{z_0 = 1}$$

- *p* Radial Basis Function Kernels

$$\boxed{\begin{aligned}[\mathbf{b}_t]_j &= \frac{\psi_j}{\sum_{i=1}^{p} \psi_i} z_t |s_0 - g_s| \\ \psi_j &= \exp(-h_j(z_t - c_j)^2)\end{aligned}}$$

Hyperparameters:

$K > 0$

$D > 0$

$\alpha > 0$

g_s

h_j

$c_j \in [0,1]$

CSAIL

Policy Parameterizations

Dynamic Movement Primitives

- Example DMP

$$K = 0$$
$$D = 0$$
$$c_0 = .95 \quad h_0 = 100$$
$$c_1 = .6 \quad h_1 = 100$$
$$\theta_x = [3, \; 0]$$
$$\theta_y = [0, -5]$$
$$\theta_\theta = [2, \; 0]$$

CSAIL

Policy Parameterizations
Dynamic Movement Primitives

- Deterministic Policy

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = -K(s - g_s) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}}$$

CSAIL

Policy Parameterizations
Stochastic DMPs

- Deterministic Policy

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = -K(s - g_s) - Dv_s + \mathbf{b}^T \boldsymbol{\theta}}$$

- Stochastic Policy $\boxed{\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})}$

Policy Parameterizations
Stochastic DMPs

- Deterministic Policy

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = -K(s - g_s) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}}$$

- Stochastic Policy $\boxed{\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})}$

    - Parameter Space Exploration

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = \mathcal{N}(-K(s - g_s) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}, \mathbf{b}^T\boldsymbol{\Sigma}\mathbf{b})}$$

CSAIL

Policy Parameterizations
Stochastic DMPs

- Deterministic Policy

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = -K(s - g_s) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}}$$

- Stochastic Policy $\boxed{\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})}$

    - Parameter Space Exploration

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = \mathcal{N}(-K(s - g_s) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}, \mathbf{b}^T\boldsymbol{\Sigma}\mathbf{b})}$$

    - Action Space Exploration

$$\boxed{\pi(a|s, v_s; \boldsymbol{\theta}) = \mathcal{N}(-K(s - g) - Dv_s + \mathbf{b}^T\boldsymbol{\theta}, \boldsymbol{\Sigma})}$$

CSAIL

Policy Parameterizations

Linear Stochastic Policy

- Action Space Exploration

$$\boxed{\pi(\mathbf{F}|\mathbf{s};\boldsymbol{\theta}) = \boldsymbol{\theta}\,\mathbf{s} + \epsilon}$$

$$\boxed{\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})}$$

CSAIL

Policy Search Methods

How do we learn θ?

Policy Search Methods
Gradient Methods

- Value Function

$$\boxed{J(\boldsymbol{\theta}) = \mathbf{E}_{\tau}[C(\tau)]}$$

- Gradient Method Parameter Update

$$\boxed{\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \eta \nabla_{\theta} J(\boldsymbol{\theta})}$$

CSAIL

Policy Search Methods

Gradient Methods: Finite Differences[1]

- Value Function

$$\boxed{J(\boldsymbol{\theta}) = \mathbf{E}_\tau[C(\tau)]}$$

- Gradient Method Parameter Update

$$\boxed{\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \eta \nabla_\theta J(\boldsymbol{\theta})}$$

- Finite Differences Gradient Estimate

$$\boxed{\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \approx \frac{J(\boldsymbol{\theta} + \boldsymbol{\delta}) - J(\boldsymbol{\theta} - \boldsymbol{\delta})}{2\delta}}$$

CSAIL

Policy Search Methods

Gradient Methods: REINFORCE[1]

- Gradient Estimate

$$\boxed{\begin{aligned} J(\boldsymbol{\theta}) &= \mathbf{E}_\tau[C(\tau)] \\ &= \int_\tau C(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \end{aligned}}$$

Policy Search Methods

Gradient Methods: REINFORCE[1]

- Gradient Estimate

$$\boxed{\begin{aligned} J(\boldsymbol{\theta}) &= \mathbf{E}_{\tau}[C(\tau)] \\ &= \int_{\tau} C(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \end{aligned}}$$

$$\boxed{\begin{aligned} \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) &= \int_{\tau} C(\tau) \nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(\tau) d\tau \\ &= \int_{\tau} C(\tau) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \\ &= \mathbf{E}_{\tau}[C(\tau) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\tau)] \end{aligned}}$$

CSAIL

Policy Search Methods

Gradient Methods: REINFORCE[1]

- Gradient Estimate

$$\boxed{\begin{aligned} J(\boldsymbol{\theta}) &= \mathbf{E}_\tau[C(\tau)] \\ &= \int_\tau C(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \end{aligned}}$$

$$\boxed{\begin{aligned} \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) &= \int_\tau C(\tau) \nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(\tau) d\tau \\ &= \int_\tau C(\tau) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \\ &= \mathbf{E}_\tau[C(\tau) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\tau)] \end{aligned}}$$

$$\boxed{= \mathbf{E}_\tau[C(\tau) \nabla_{\boldsymbol{\theta}} \log \pi(a|s, v_s; \boldsymbol{\theta})]}$$

CSAIL

Policy Search Methods
Gradient Methods: REINFORCE[1]

- Gradient Estimate

$$\boxed{\begin{aligned} J(\boldsymbol{\theta}) &= \mathbf{E}_\tau[C(\tau)] \\ &= \int_\tau C(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \end{aligned}}$$

$$\boxed{\begin{aligned} \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) &= \int_\tau C(\tau) \nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(\tau) d\tau \\ &= \int_\tau C(\tau) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\tau) p_{\boldsymbol{\theta}}(\tau) d\tau \\ &= \mathbf{E}_\tau[C(\tau) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\tau)] \end{aligned}}$$

$$\boxed{= \mathbf{E}_\tau[C(\tau) \nabla_{\boldsymbol{\theta}} \log \pi(a|s, v_s; \boldsymbol{\theta})]}$$

- Only works with **stochastic policies**
- High variance can be mitigated by using **optimal baseline**[3]

CSAIL

Policy Search Methods

Policy Improvement with Path Integrals[2]

- (Discrete) Path Integral — path cost with penalty term for unlikely trajectories

$$\boxed{S(\tau_t) = \sum_{t'=t}^{T} q_{t'}(s_{t'}, v_{s,t'}) + a_{t'}^T R a_{t'} - \log p(\tau_t | s_{t'}, v_{s,t'})}$$

Constant: *R* (weight on the control cost)

CSAIL

Policy Search Methods

Policy Improvement with Path Integrals[2]

- Softmax representing the probability of a trajectory given a set of trajectories and their path integrals

$$\boxed{P(\tau_t) = \frac{\exp(-\frac{1}{\lambda}S(\tau_t))}{\sum_\tau \exp(-\frac{1}{\lambda}S(\tau_t))}}$$

Constant: λ

CSAIL

Policy Search Methods

Policy Improvement with Path Integrals[2]

- Softmax representing the probability of a trajectory given a set of trajectories and their path integrals

$$\boxed{P(\tau_t) = \frac{\exp(-\frac{1}{\lambda}S(\tau_t))}{\sum_\tau \exp(-\frac{1}{\lambda}S(\tau_t))}}$$

Constant: λ

- Parameter Update

$$\boxed{\delta\boldsymbol{\theta}_t = \sum_\tau P(\tau_t)\mathbf{M}_t\epsilon_t}$$

$$\boxed{\mathbf{M}_t = \frac{R^{-1}\mathbf{b}_t\mathbf{b}_t^T}{\mathbf{b}_t^T R^{-1}\mathbf{b}_t}}$$

$$\boxed{[\delta\boldsymbol{\theta}]_j = \frac{\sum_{t=0}^{T}(T-t)\psi_{j,t}[\delta\boldsymbol{\theta}_t]_j}{(T-t)\psi_{j,t}}}$$

$$\boxed{\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \delta\boldsymbol{\theta}}$$

CSAIL

Policy Search Methods

Policy Improvement with Path Integrals[2]

- Softmax representing the probability of a trajectory given a set of trajectories and their path integrals

$$P(\tau_t) = \frac{\exp(-\frac{1}{\lambda}S(\tau_t))}{\sum_\tau \exp(-\frac{1}{\lambda}S(\tau_t))}$$

- Referred to as **probability weighted averaging**

- Parameter Update

$$\delta\boldsymbol{\theta}_t = \sum_\tau P(\tau_t)\mathbf{M}_t\epsilon_t$$

$$[\delta\boldsymbol{\theta}]_j = \frac{\sum_{t=0}^{T}(T-t)\psi_{j,t}[\delta\boldsymbol{\theta}_t]_j}{(T-t)\psi_{j,t}}$$

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \delta\boldsymbol{\theta}$$

- Paths with high cost have low weight/probability and vice versa

CSAIL

Results

Problem Description
Stochastic DMP with 2D θ

- Stochastic DMP - each dimension of the state has its own controller

$$\boxed{\pi(F_x|x, v_x) = -K(x - g_x) - Dv_x}$$

$$\boxed{\pi(F_y|y, v_y; \boldsymbol{\theta}_y) = \mathcal{N}(-K(y - g_y) - Dv_y + \mathbf{b}^T\boldsymbol{\theta}_y, \mathbf{b}^T\boldsymbol{\Sigma}\mathbf{b})}$$

$$\boxed{\pi(F_\theta|\theta, v_\theta; \boldsymbol{\theta}_\theta) = \mathcal{N}(-K(\theta - g_\theta) - Dv_\theta + \mathbf{b}^T\boldsymbol{\theta}_\theta, \mathbf{b}^T\boldsymbol{\Sigma}\mathbf{b})}$$

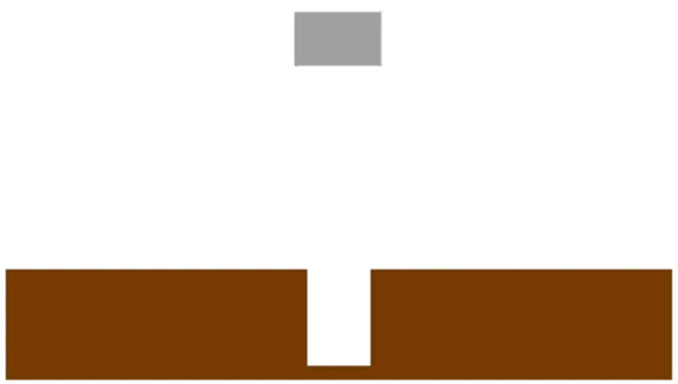
- p = 1 RBF Kernel

$$\boxed{c_0 = .95 \; h_0 = 1.}$$

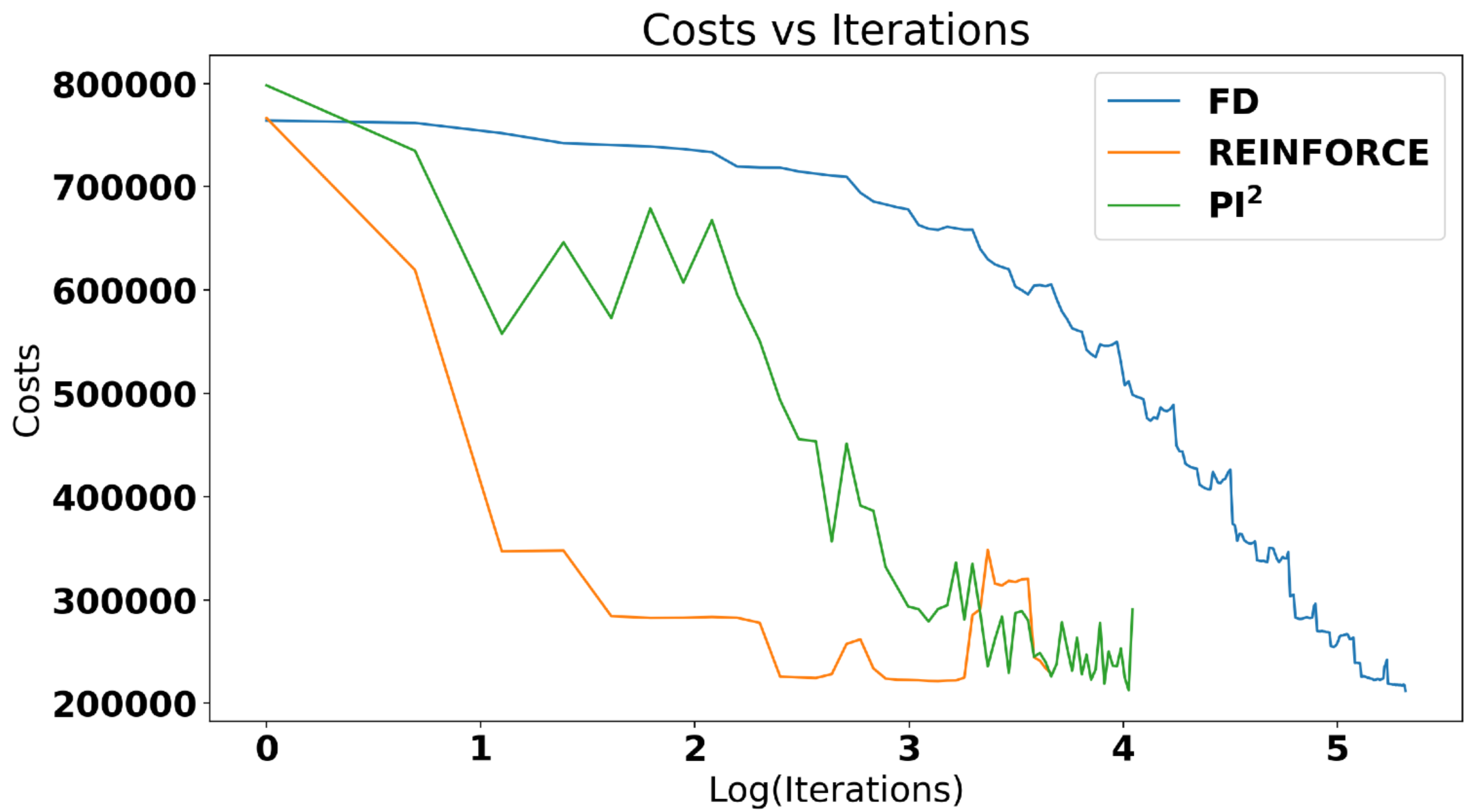
CSAIL

Results

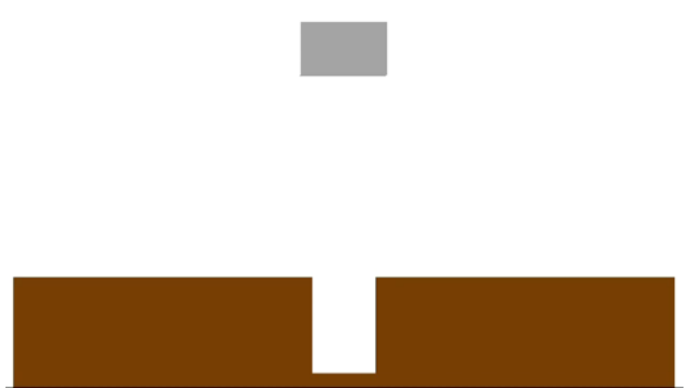
Stochastic DMP with 2D θ



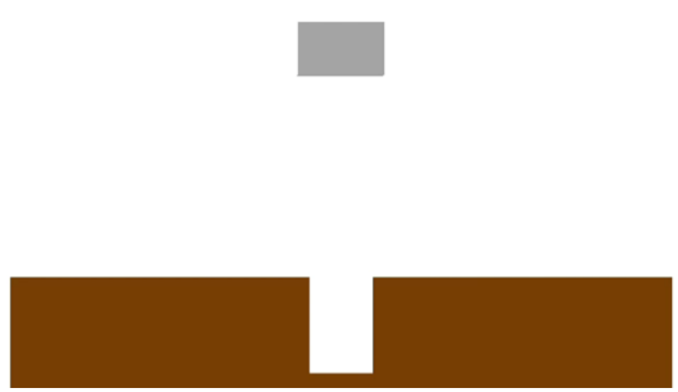
Initial θ



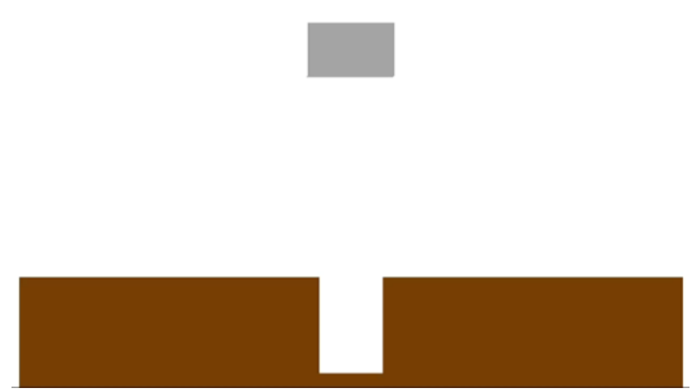
Finite Differences



REINFORCE



PI^2



CSAIL

Problem Description
Linear Stochastic Policy

- Stochastic linear policy - the parameter determines all dimensions of the action

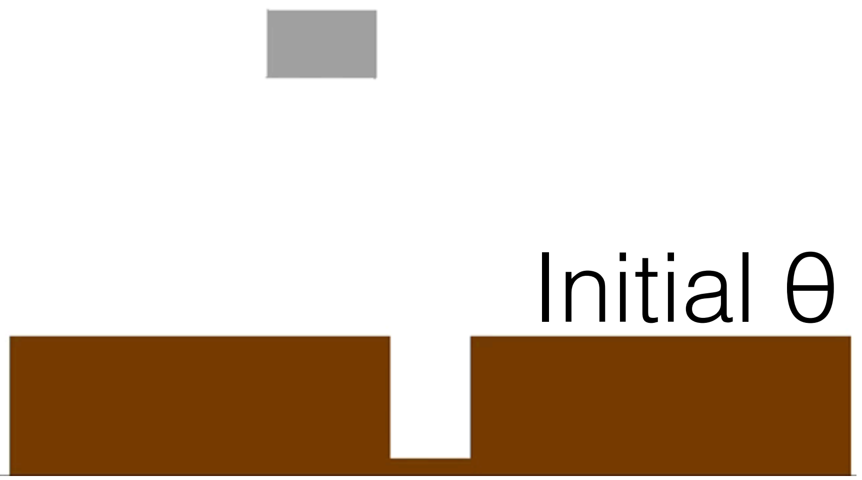
$$\boxed{\pi(\mathbf{F}|\mathbf{s};\boldsymbol{\theta}) = \boldsymbol{\theta}\,\mathbf{s} + \epsilon}$$

$$\mathbf{F} = \begin{bmatrix} F_x \\ F_y \\ F_\theta \end{bmatrix} \mathbf{s} = \begin{bmatrix} x \\ y \\ \theta \\ v_x \\ v_y \\ v_\theta \end{bmatrix} \mathbf{g} = \begin{bmatrix} g_x \\ g_y \\ g_\theta \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

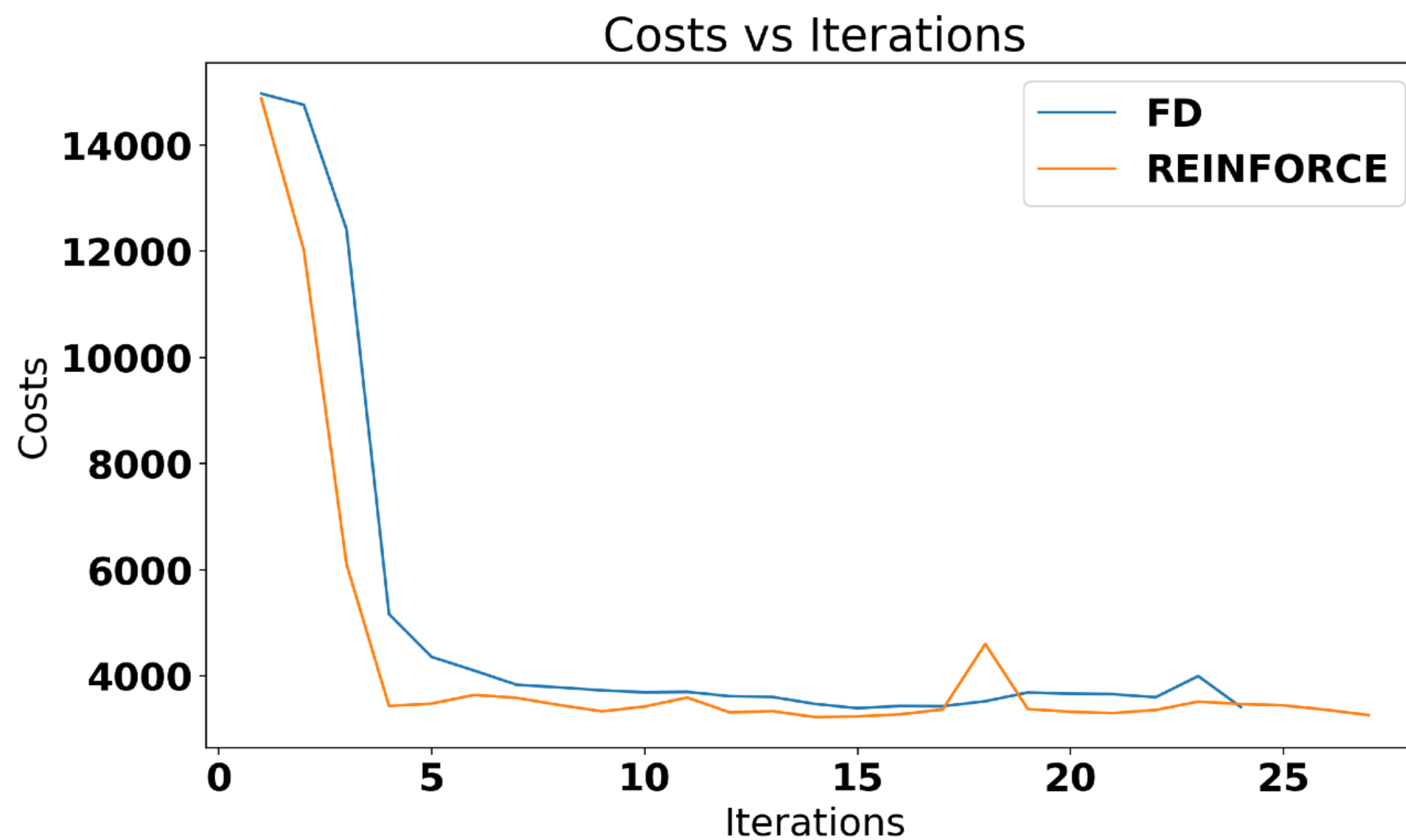
CSAIL

Results

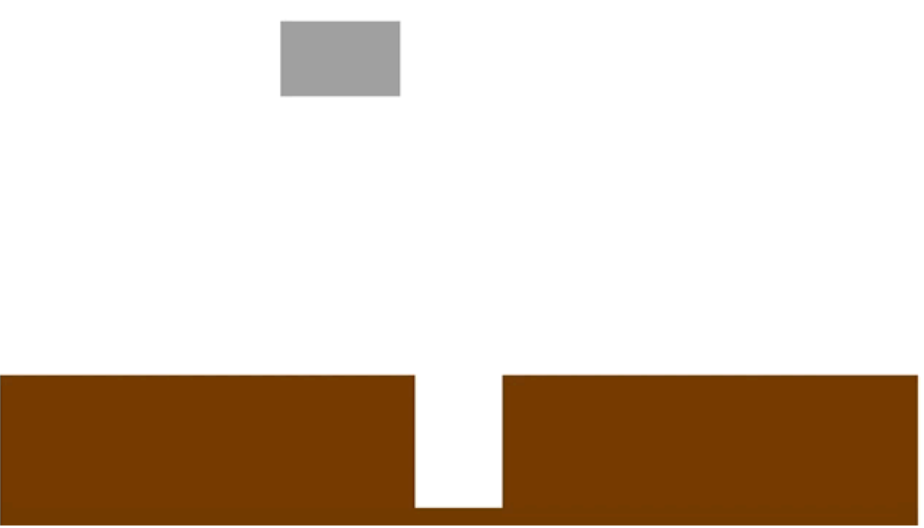
Linear Stochastic Policy



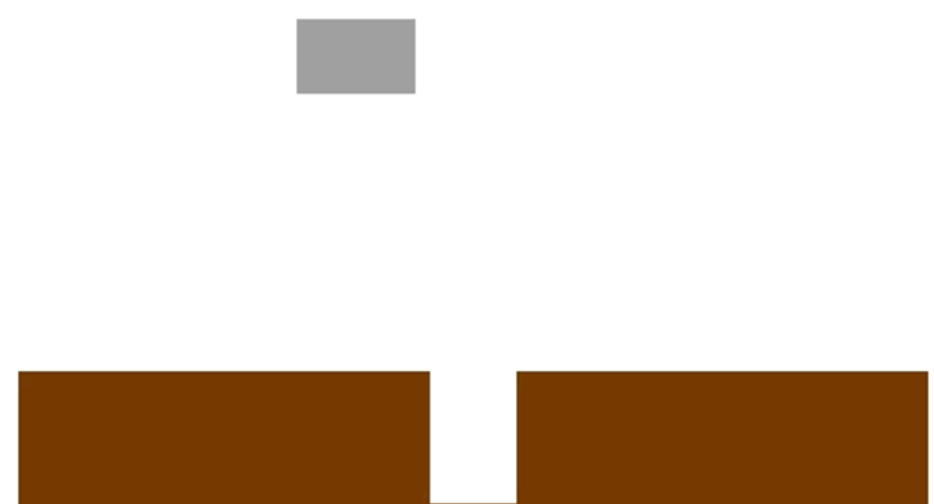
Initial θ



Finite Differences



REINFORCE



Analysis

- **Difficult** to manipulate a peg into a hole using simply **one-dimensional RBF** forcing term.
- **Linear stochastic policies are better** suited for the insertion task.
- **Non-smooth cost function** makes peg insertion task difficult to learn in general.

Novelty and Contribution

- Compared three methods of policy search: REINFORCE, PI^2, and Finite Differences on a peg insertion task.

- Found that **REINFORCE outperforms both PI^2 and Finite Differences** for a 2D parameter with a stochastic **DMP** policy.

- **REINFORCE** and Finite Differences both perform **better** with a **linear** stochastic policy than with a **DMP** stochastic policy.

CSAIL

Thank you!

- Questions?

- Contact Information: carism@mit.edu

- References

[1] Peters, Jan (2010) Policy gradient methods. Scholarpedia, 5(11):3698.
[2] Theodorou, Evangelos, Jonas Buchli, and Stefan Schaal. "A generalized path integral control approach to reinforcement learning." Journal of Machine Learning Research 11.Nov (2010): 3137-3181.
[3] Zhao, Tingting, et al. "Analysis and improvement of policy gradient estimation." Advances in Neural Information Processing Systems. 2011.

CSAIL