Reinforcement Learning for Manipulation
Motivation

Many robotic behaviors are hand tuned by engineers.

- Designing spline nodes...
- Finding appropriate timings...
- Tuning thresholds...
- ...

Reinforcement Learning for Manipulation
Peter Pastor
Learning from Demonstration

\[ u = \pi(x, t, w) \]

- imitation learning
- policy
- motor command
- state
- time
- adjustable parameters

Reinforcement Learning for Manipulation
Peter Pastor
Learning from Demonstration

\[ \dot{x} = f(x, t, w) \]

change of state, state, time, learned parameters

imitation learning \rightarrow \text{policy}

Dynamic Movement Primitives

Reinforcement Learning for Manipulation
Peter Pastor
Learning from Demonstration

imitation learning → policy

\[ \dot{x} = f(x, t, w) \]

change of state
state
learned parameters
time

Dynamic Movement Primitives

Reinforcement Learning for Manipulation
Peter Pastor
Problem Statement

- From demonstration only kinematics are observable
- Dynamic tasks may be hard/impossible to plan/demonstrate
- Demonstrated behavior may be specific to particular robot

imitation learning \rightarrow \text{policy}
Problem Statement

- From demonstration only kinematics are observable
- Dynamic tasks may be hard/impossible to plan/demonstrate
- Demonstrated behavior may be specific to particular robot

imitation learning $\rightarrow$ initial policy
Problem Statement

- From demonstration only kinematics are observable
- Dynamic tasks may be hard/impossible to plan/demonstrate
- Demonstrated behavior may be specific to particular robot

![Diagram showing imitation learning leading to an initial policy, which is then improved through a cost function.](image)
Problem Statement

- From demonstration only kinematics are observable
- Dynamic tasks may be hard/impossible to plan/demonstrate
- Demonstrated behavior may be specific to particular robot
The $\text{PI}^2$ Algorithm

demonstration $\rightarrow$ initial parameters $\rightarrow$ policy
The $\text{PI}^2$ Algorithm

demonstration $\rightarrow$ initial parameters $\rightarrow$ policy

$n$ noisy rollouts + cost of each rollout
The PI$^2$ Algorithm

demonstration $\rightarrow$ initial parameters $\rightarrow$ policy

\[ n \text{ noisy rollouts} + \text{cost of each rollout} \]

Policy Improvement using Path Integrals (PI$^2$)

[Theodorou, Buchli, Schaal]
The $\text{PI}^2$ Algorithm

- Demonstration
- Initial parameters
- New parameters
- Policy
- $n$ noisy rollouts
- Cost of each rollout

Policy Improvement using Path Integrals ($\text{PI}^2$)

[Theodorou, Buchli, Schaal]
The $\text{PI}^2$ Algorithm

1. Demonstration $\rightarrow$ Initial parameters
2. New parameters $\rightarrow$ Policy
3. Policy $\rightarrow$ $n$ noisy rollouts + Cost of each rollout
4. Policy Improvement using Path Integrals ($\text{PI}^2$)
5. Final policy $\leftarrow$ Final parameters

[Theodorou, Buchli, Schaal]
The PI² Algorithm

- Demonstration
- Initial parameters
- Policy
- New parameters
- Policy Improvement using Path Integrals (PI²)

- Task execution
- Final policy
- Final parameters
- $n$ noisy rollouts
- Cost of each rollout

[Theodorou, Buchli, Schaal]
Example: Pool Hackathon

- Fully integrated pool application:
  - table localization
  - shot planing
  - shot execution
  ...

- Specialized pool cue

- Pool stroke motion has limited controllability and could be more powerful
Example: Pool Hackathon

Goals:

- Avoid using the special cue stick
- Learn a powerful and precise pool stroke
Software Architecture

- dmp_motion_generation
- task_manager
- task_recorder
- policy_library
- pool_task.cpp
- dmp_motion_learner
- dmp_motion_controller
- pool_task_transform.cpp
- policy_improvement
- policy_improvement_loop

Reinforcement Learning for Manipulation
Peter Pastor
The Approach

Cost function

1) Minimize ball travel time
2) Minimize offset to center
The Approach

Cost function

1) Minimize ball travel time
2) Minimize offset to center
The Approach

- Cost function
  1) Minimize ball travel time
  2) Minimize offset to center

- Coordinate transformation
  Mapping from end-effector space to lower dimensional task constraints satisfying space
Learning a Powerful and Precise Pool Stroke
(Near) Future Work

- Apply the framework to more challenging tasks
  Opening a bottle with two hands
- Associate sensory information to learned policies to enable predictive model

+ Cost of each trial
(Near) Future Work

- Apply the framework to more challenging tasks
  - Opening a bottle with two hands
- Associate sensory information to learned policies to enable predictive model
  - Resample recorded data traces such that data points from different sensors have the same time stamp
  - Compute statistics over all trials for each data point
  - Look for low variance features
  - Use these features to predict the outcome (cost) and/or detect failure
    - Example: Humans walking and predicting foot contact forces
    - Liquid transfer using a pipette

Reinforcement Learning for Manipulation
Peter Pastor
THANKS !!!

Questions ??