COM SCI 188

Intro to Robotics

Lecture 12

Yuchen Cui

Winter 2026

Agenda

- Announcements
- Recap: Reinforcement Learning

- TODAY: Imitation Learning
  - Dynamic Movement Primitives
  - Behavioral Cloning
  - Inverse Reinforcement Learning

Announcements

- Coding Assignment 3 is out, due 2/23
- Midterm next Thursday 2/19 (in-class)


Reinforcement Learning
First trial...

Deep Reinforcement Learning for Robotics: A Survey of Real-World Successes

Chen Tang[1,*], Ben Abbatematteo[1,*], Jiaheng Hu[1,*], Rohan Chandra[2], Roberto Martín-Martín[1], Peter Stone[1,3]

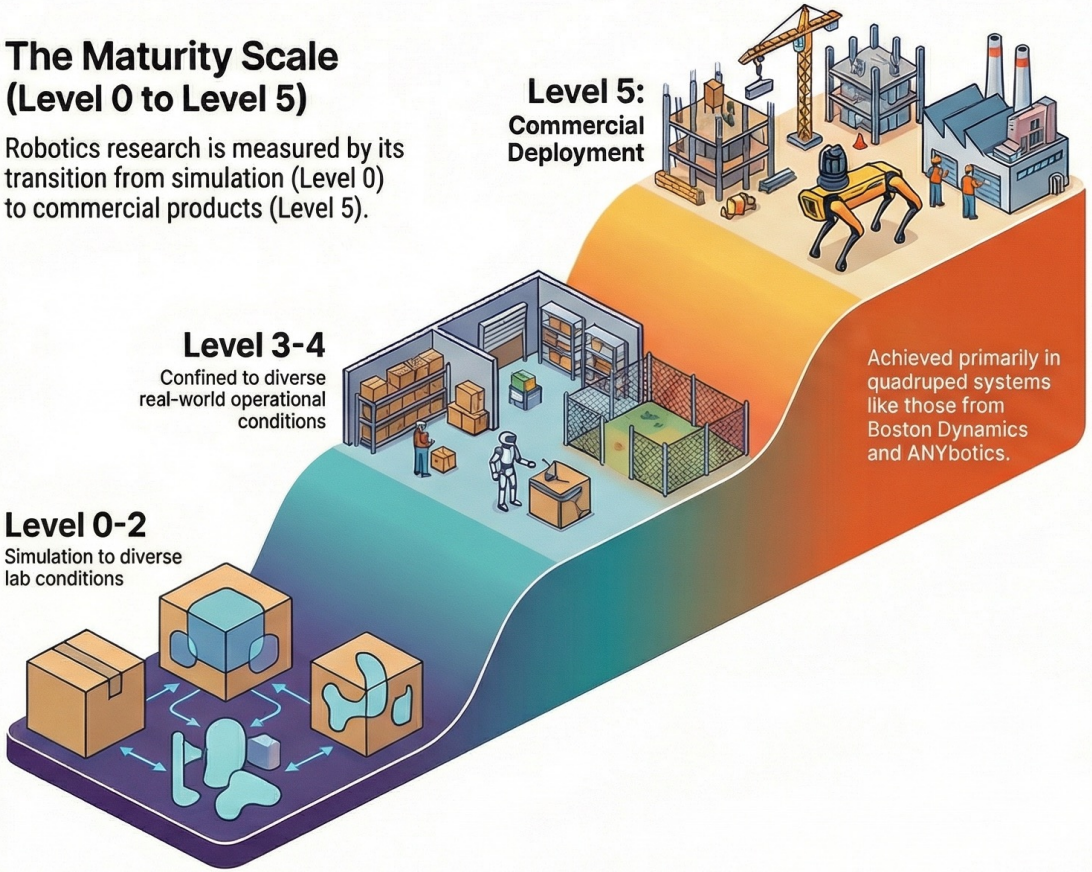
https://arxiv.org/pdf/2408.03539

The State of Deep Reinforcement Learning (DRL) in Real-World Robotics

A survey of DRL successes in robotics, contrasting high-success areas like quadruped locomotion with emerging fields like human-robot interaction, and outlining the maturity from simulation to commercial deployment.

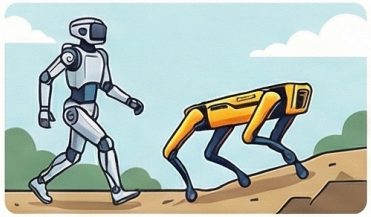
The Maturity Scale (Level 0 to Level 5)

Robotics research is measured by its transition from simulation (Level 0) to commercial products (Level 5).



Competency Maturity & Future Hurdles

Locomotion vs. Interaction Gaps

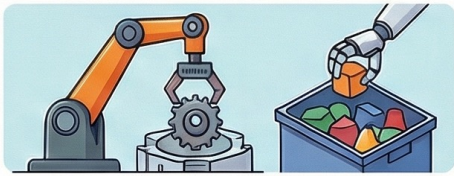


Mature Competency: Locomotion
Locomotion is highly mature.

Early Stage: Interaction & Coordination
Human-Robot Interaction and Multi-Robot coordination remain in early stages.

Success in Restricted Domains

DRL excels when tasks are enumerable a priori, like grasping and assembly.



Three Major Open Challenges

Improving Sample Efficiency
Learning faster from fewer examples.

Enabling Safe Real-World Learning
Preventing damage during training.

Mastering Long-Horizon Tasks
Planning and executing multi-step sequences.

Practical Challenges of RL for Robotics

- Sample Efficiency
- Safety
- Reset
- Reward Specification

Specifying reward for RL is hard…

Reward hacking: RL agent learns to exploit loopholes or unintended behaviors in its reward function to achieve high rewards without actually accomplishing the intended task

Imitation Learning

Why learn from demonstrations?

- Natural and expressive
- No expert knowledge required
- Valuable human intuition
- Program new tasks as-needed

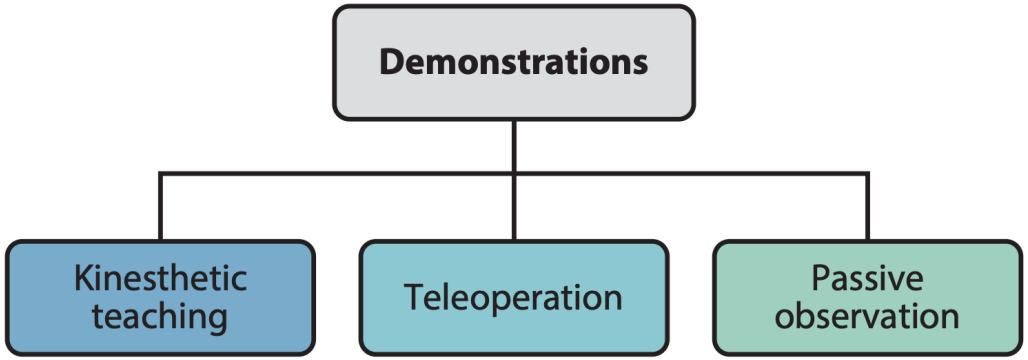
Human babies imitate

What to imitate?

Demonstration(s) -> Autonomous Behavior

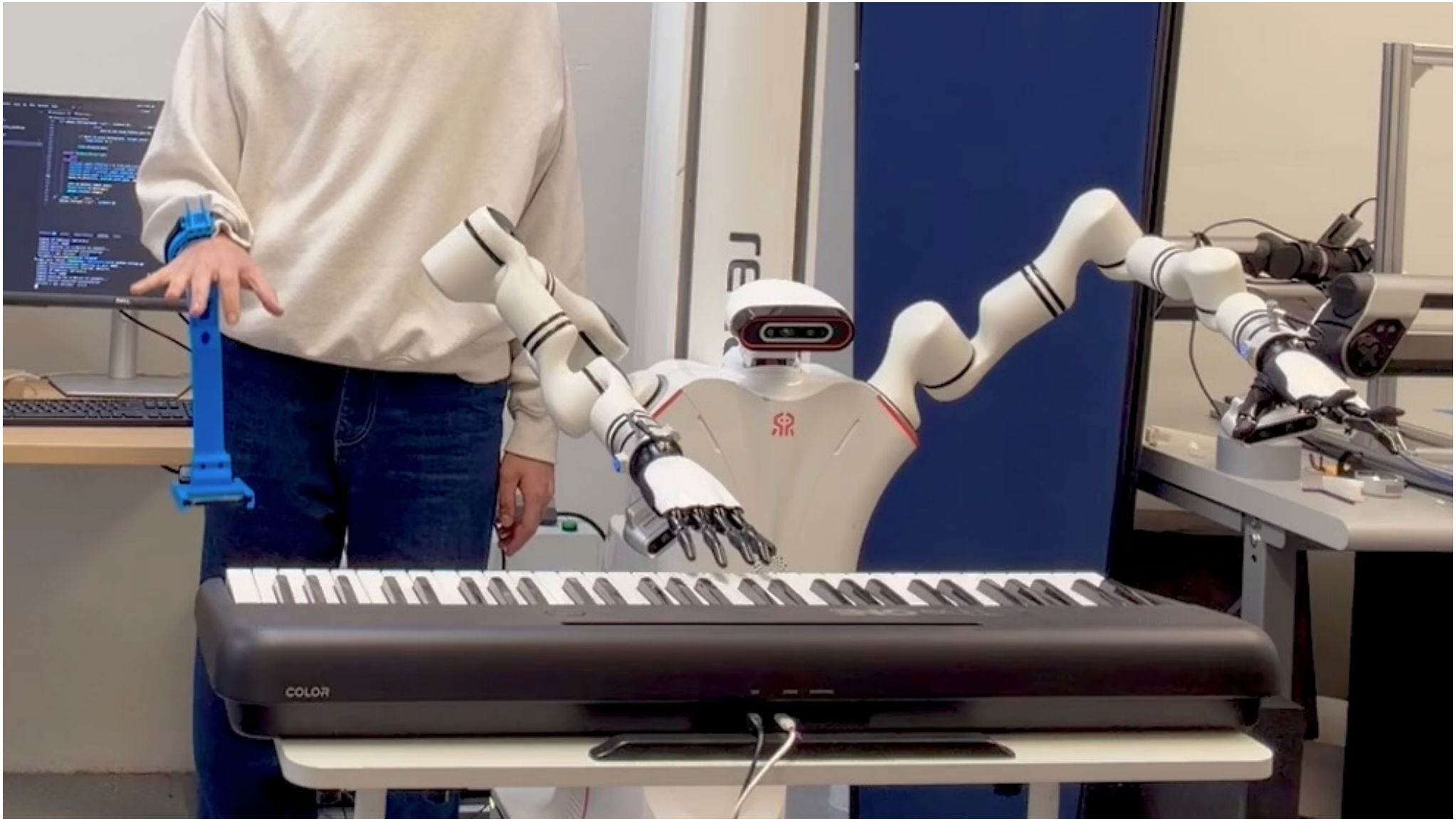
- ***Dynamic Movement Primitives (DMP)***: replay the **motion**
- ***Behavior Cloning (BC):*** supervised learning of expert **policy**
- ***Inverse Reinforcement Learning (IRL)***: inferring the underlying **intent**

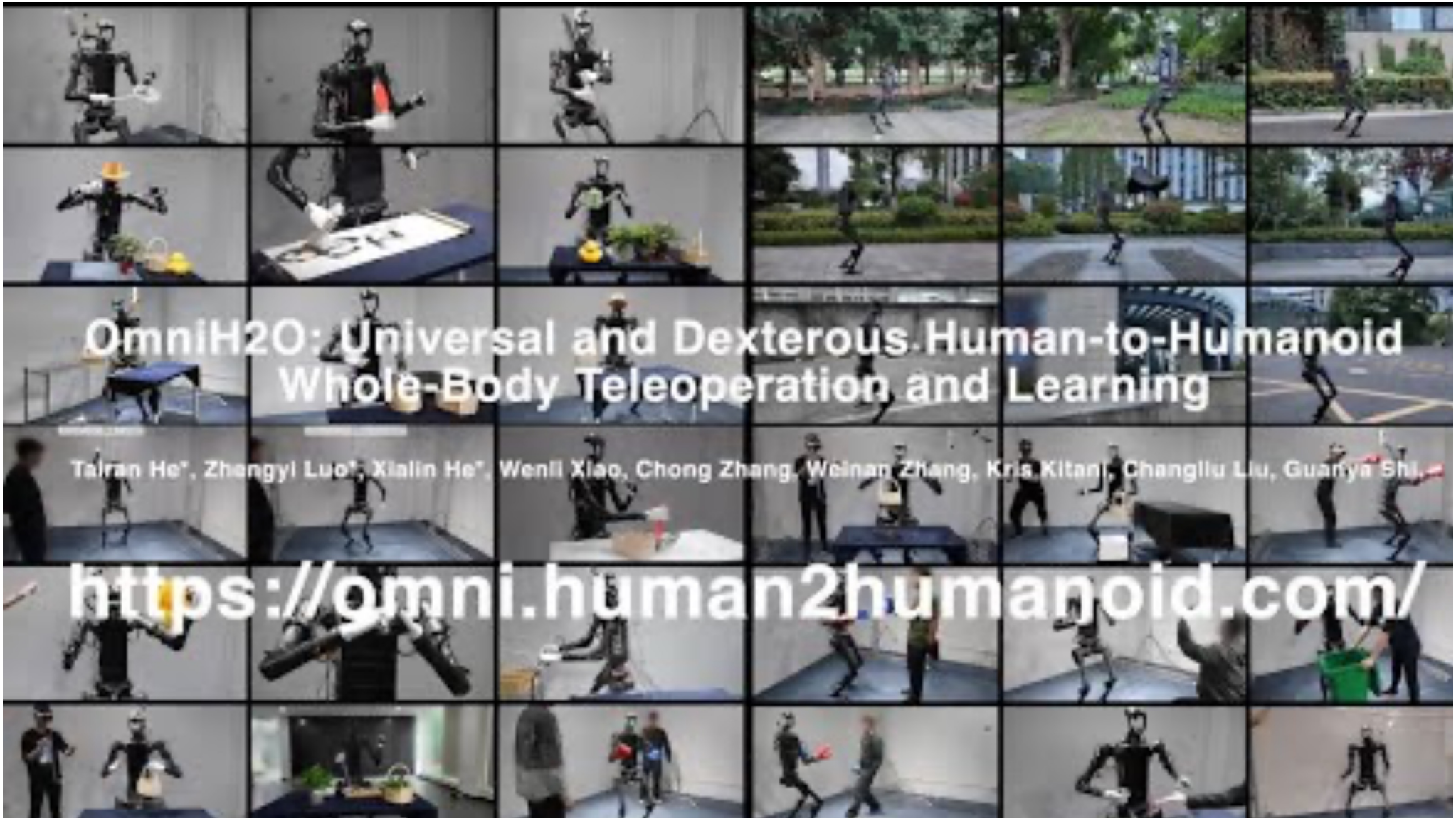
Types of Demonstrations



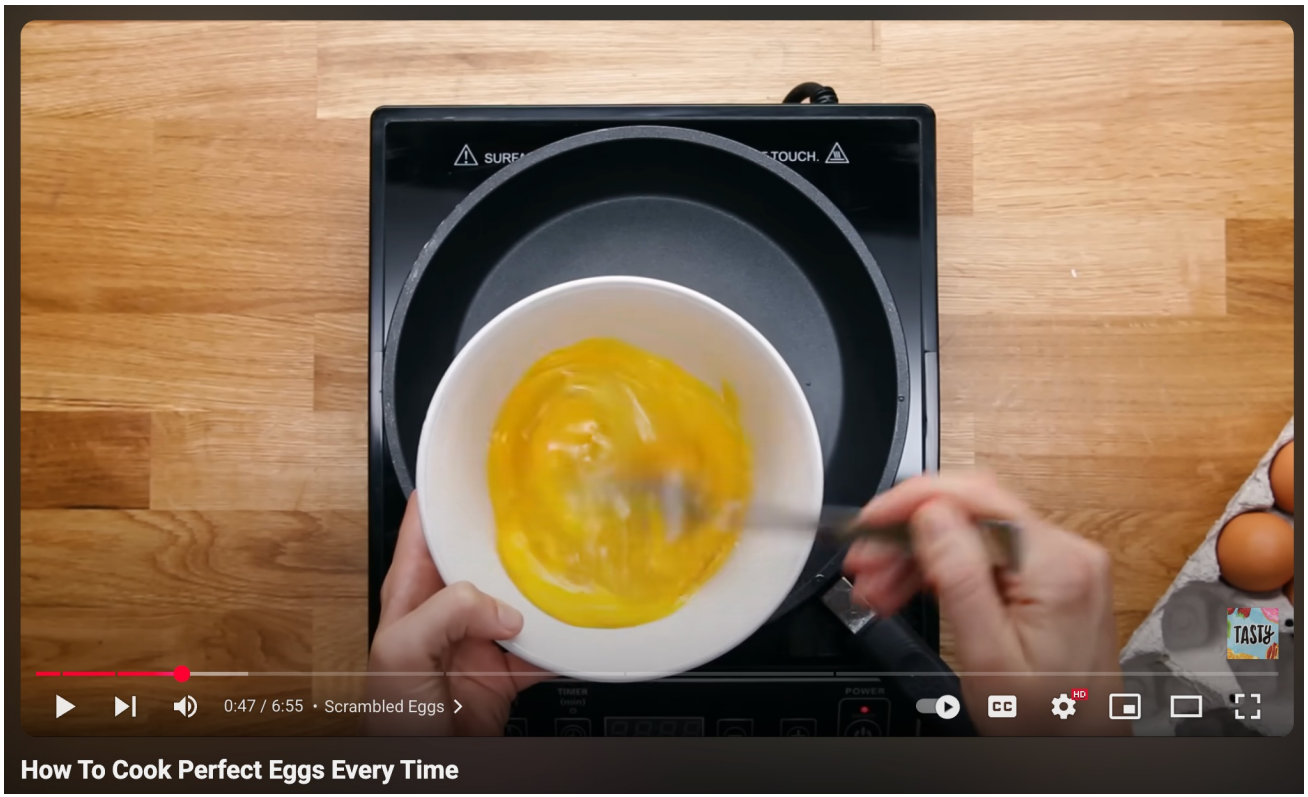
Ravichandar, Harish, et al. "Recent advances in robot learning from demonstration."
Annual review of control, robotics, and autonomous systems 3.1 (2020): 297-330.


tional Center

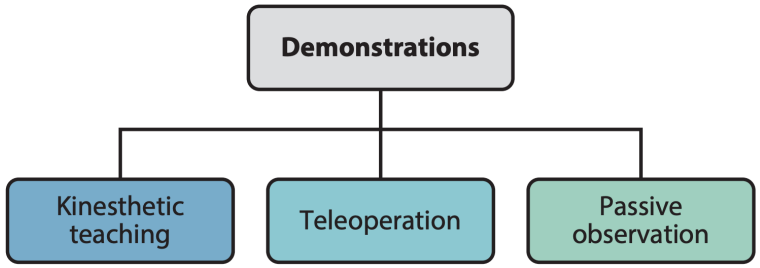

COLOR


OmniH2O: Universal and Dexterous Human-to-Humanoid
Whole-Body Teleoperation and Learning
Tairan He*, Zhengyi Luo*, Xialin He*, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, Guanya Shi
https://omni.human2humanoid.com/

Learning from Observations



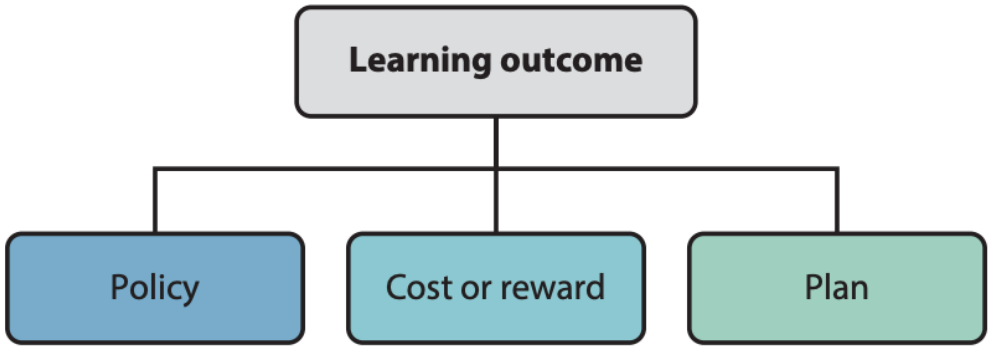
Types of Demonstrations



Demonstration	Ease of demonstration	High DOFs	Ease of mapping
Kinesthetic teaching	✓		✓
Teleoperation		✓	✓
Passive observation	✓	✓	

Ravichandar, Harish, et al. "Recent advances in robot learning from demonstration."
Annual review of control, robotics, and autonomous systems 3.1 (2020): 297-330.

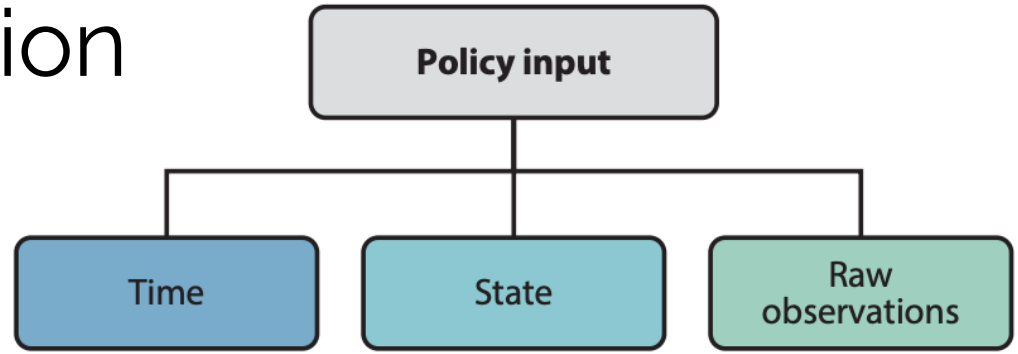
Learning Outcomes



Learning outcome	Low-level control	Action space continuity	Compact representation	Long-horizon planning	Multistep tasks
Policy	✓	✓	✓		
Cost or reward	✓	✓		✓	
Plan			✓	✓	✓

Ravichandar, Harish, et al. "Recent advances in robot learning from demonstration."
Annual review of control, robotics, and autonomous systems 3.1 (2020): 297-330.

Policy Parameterization



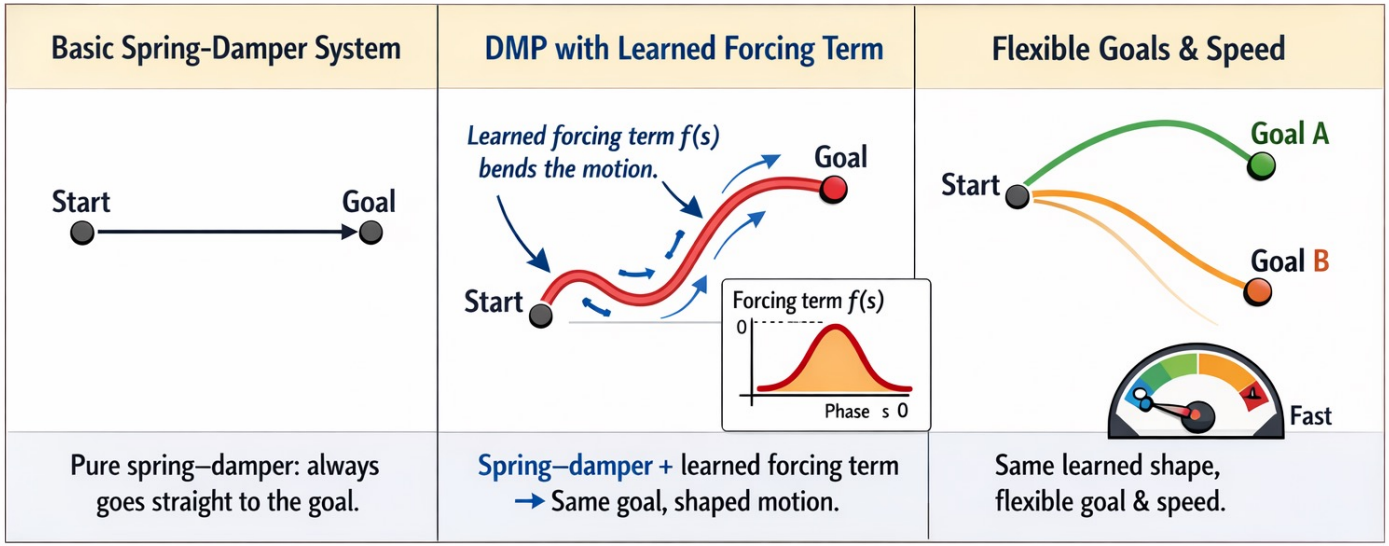
Policy input	Ease of design	Performance guarantees	Robustness to perturbations	Task variety	Algorithmic efficiency
Time	✓	✓			✓
State		✓	✓		✓
Raw observations	✓		✓	✓	

Ravichandar, Harish, et al. "Recent advances in robot learning from demonstration."
Annual review of control, robotics, and autonomous systems 3.1 (2020): 297-330.

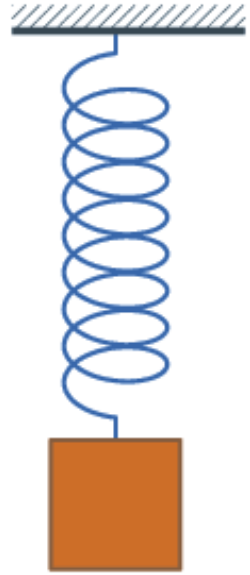
Dynamic Movement Primitives

Dynamic Movement Primitives

learnable dynamical systems for generating movements
stable, spring-like motion models augmented with a learned shaping term that lets you reproduce and flexibly adapt demonstrated movements to new goals and speeds



Dynamic Movement Primitives



goal goal initial state

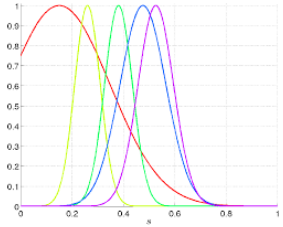
$$\tau \dot{v} = K(g - x) - Dv + (g - x_0) f$$
$$\tau \dot{x} = v$$

K: spring constant
D: damping term

Non-linear force function:

$$f(s) = \frac{\sum_i w_i \psi_i(s) s}{\sum_i \psi_i(s)} \qquad \psi_i(s) = \exp(-h_i(s - c_i)^2)$$

Gaussian basis functions

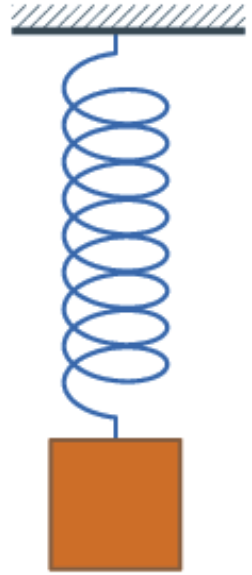


canonical system: $\tau \dot{s} = -\alpha s$

s: phase variable

Pastor, Peter, et al. "Learning and generalization of motor skills by learning from demonstration." *2009 IEEE international conference on robotics and automation*. IEEE, 2009.

Dynamic Movement Primitives



Learning:

$$f_{\text{target}}(s) = \frac{-K(g-x) + Dv + \tau\dot{v}}{g - x_0} \qquad f(s) = \frac{\sum_i w_i \psi_i(s) s}{\sum_i \psi_i(s)}$$

$$J = \sum_s \left(f_{\text{target}}(s) - f(s)\right)^2$$

Linear regression

Pastor, Peter, et al. "Learning and generalization of motor skills by learning from demonstration." *2009 IEEE international conference on robotics and automation*. IEEE, 2009.

Characteristics of DMPs

- Convergence to the goal g is guaranteed (for bounded weights) since f(s) vanishes at the end of a movement.
- The weights wi can be learned to generate any desired *smooth* trajectory.
- The equations are spatial and temporal invariant, i.e., movements are self-similar for a change in goal, start point, and temporal scaling without a need to change the weights wi .
- The formulation generates movements which are robust against perturbation due to the inherent attractor dynamics of the equations.

Pastor, Peter, et al. "Learning and generalization of motor skills by learning from demonstration." *2009 IEEE international conference on robotics and automation*. IEEE, 2009.

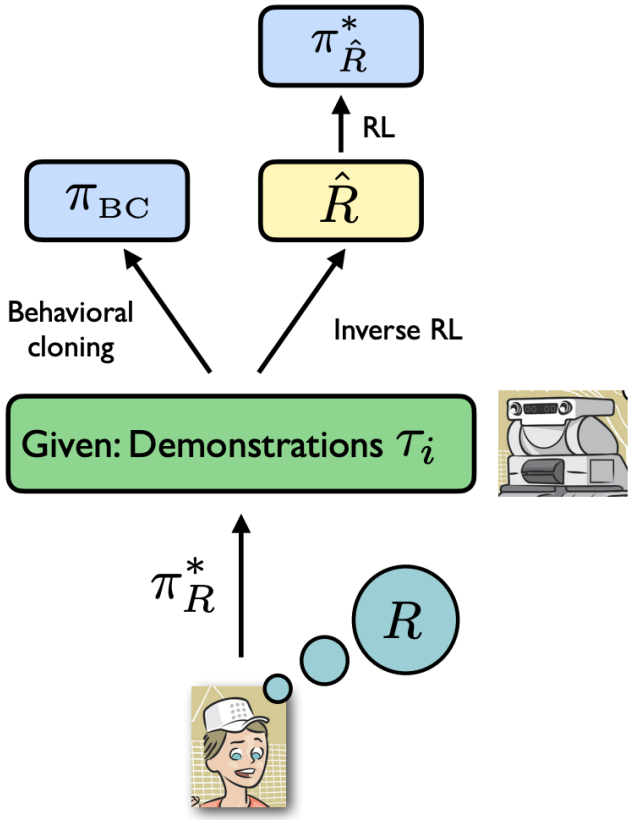
Summary

- DMP enable learning “movement styles” while enabling generalization to new movement targets
- DMP is a purely kinematic account
- DMP addresses timing, but account of coordination is limited
- DMP for different tasks and their combination…?

credit: Gregor Schöner

How to imitate complex behavior?

Imitation Learning

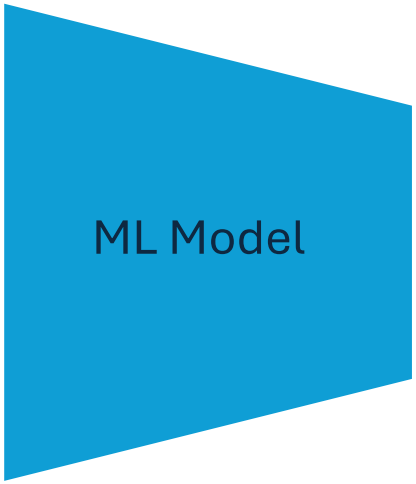


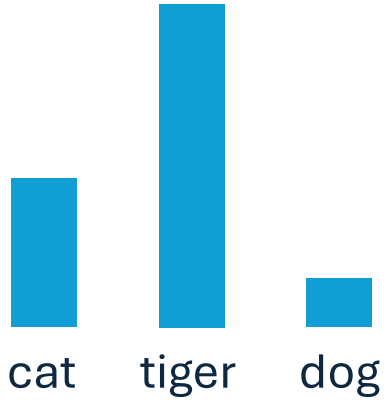
slide credit: Scott Niekum

Berkeley CS285: https://www.youtube.com/watch?v=tbLaFtYpWWU&list=PL_iWQOsE6TfVYGEGiAOMaOzzv41Jfm_Ps&index=4

Supervised Learning

Input
image

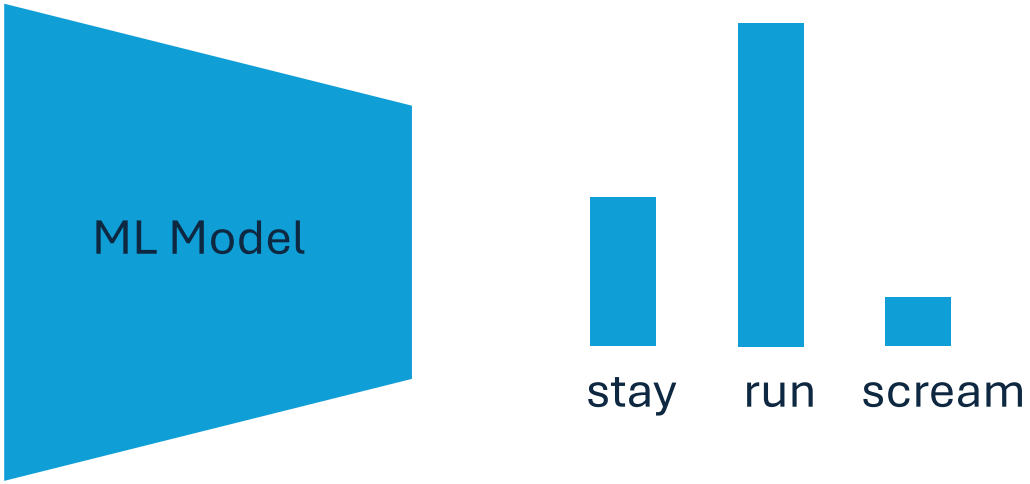




Output
class label

Berkeley CS285: https://www.youtube.com/watch?v=tbLaFtYpWWU&list=PL_iWQOsE6TfVYGEGiAOMaOzzv41Jfm_Ps&index=4

Supervised Learning / Behavior Cloning



Input
state/observation

Policy

Output
action

NOT i.i.d. -> independent and identically distributed

The i.i.d. Assumption

"The training and testing data are **independent** and **identically** distributed."

The i.i.d. Assumption

Input Data Distribution

“Poodle”

“Chihuahua”

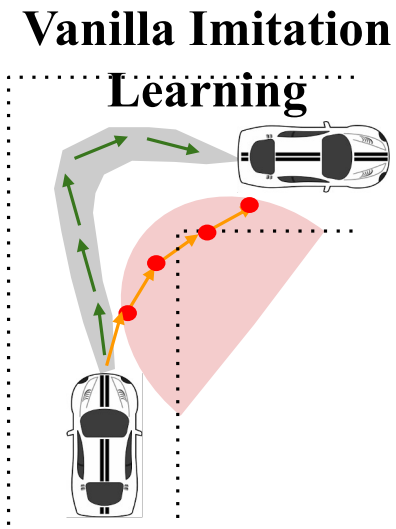
“Shar-Pei”

Not identically distributed…

Robustness refers to the ability of a system, model, or method to maintain performance or produce reliable results **despite variations, noise, errors, or adversarial conditions** in the input or environment.

Input Data Distribution

End-to-End Control Tasks



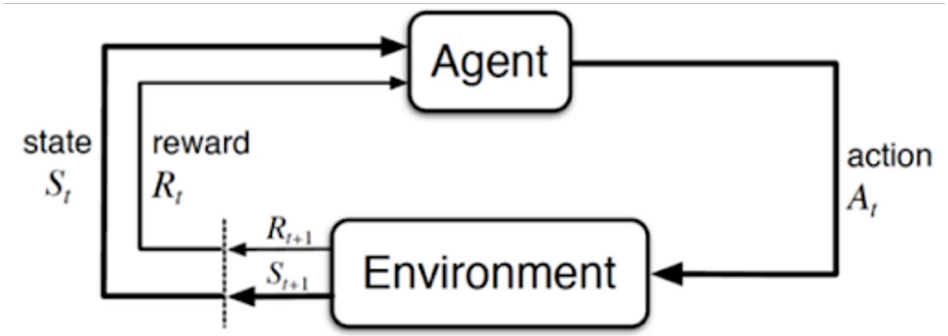
Not independently distributed...

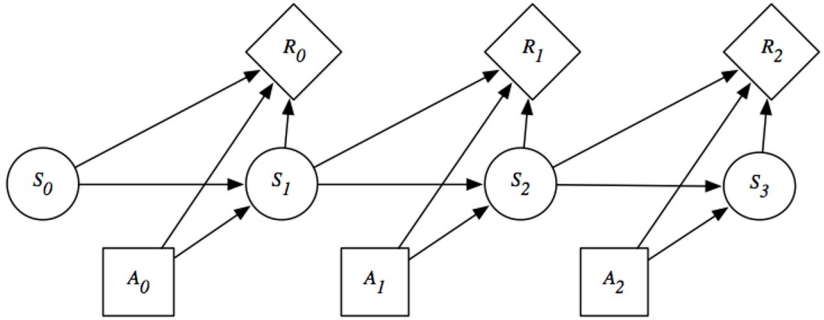
Expert Demo Agent Execution Sub-optimal State Sub-optimal Area

Berkeley CS285: https://www.youtube.com/watch?v=tbLaFtYpWWU&list=PL_iWQOsE6TfVYGEGiAOMaOzzv41Jfm_Ps&index=4

Formalizing Sequential Decision Making

Markov Decision Process <S,A,P,R>





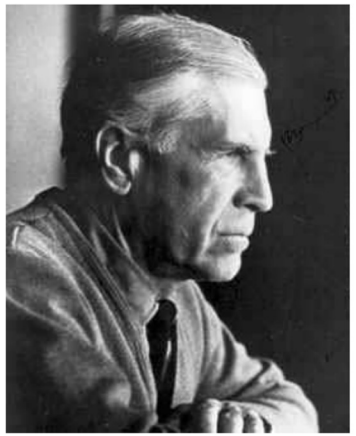
Berkeley CS285: https://www.youtube.com/watch?v=tbLaFtYpWWU&list=PL_iWQOsE6TfVYGEGiAOMaOzzv41Jfm_Ps&index=4

$\mathbf{s}_t$ – state
$\mathbf{a}_t$ – action

Richard Bellman

American applied mathematician
introduced **dynamic programming** in 1953

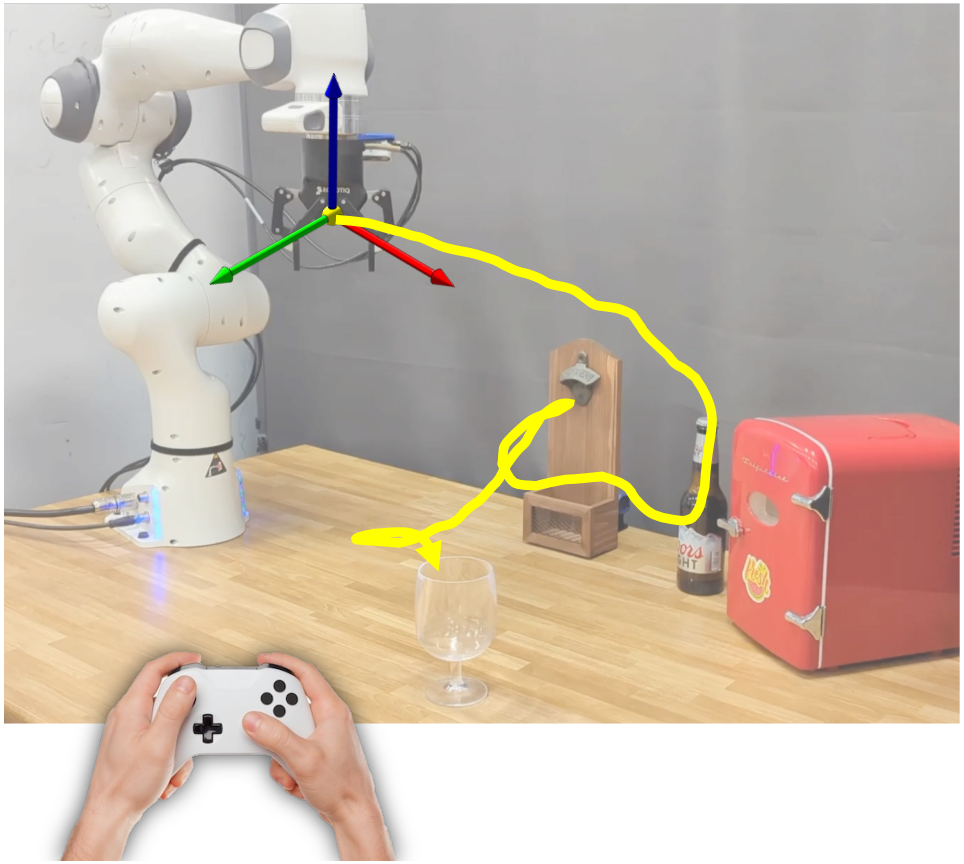
$\mathbf{x}_t$ – state
$\mathbf{u}_t$ – action управление

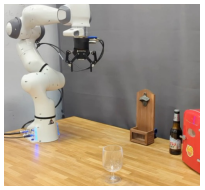
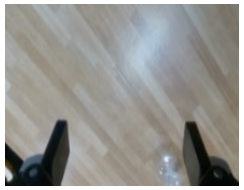


Lev Pontryagin

Soviet mathematician
algebraic topology, differential topology and **optimal control**

Behavioral Cloning



$s = (\text{Third-person View}, \text{Wrist-camera View}, \text{proprio.})$

$a = (\Delta x, \Delta y, \Delta z, \Delta rx, \Delta ry, \Delta rz, close\_gripper)$

Gripper Velocity Grasping?

$D = \{(s, a)\}$

Policy (θ)

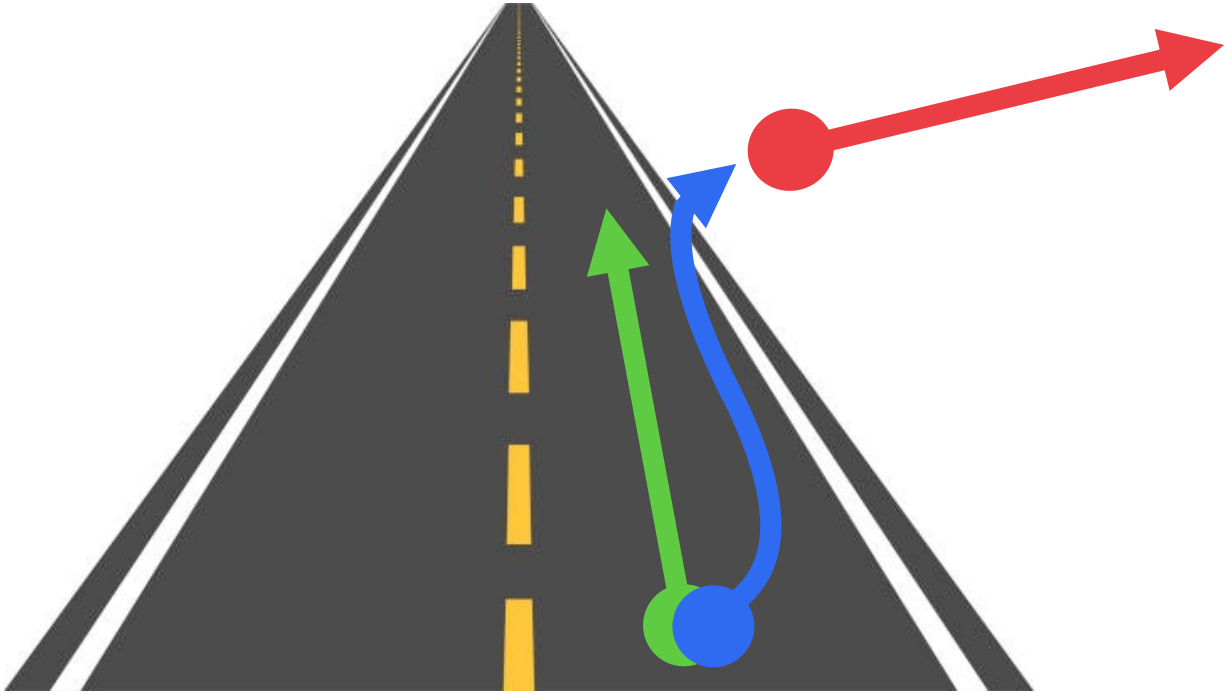
a_t

action

Input State

$$\mathcal{L}(\theta) = -\mathbb{E}_{(s,a)\sim D}[\log \pi_\theta(a|s)]$$

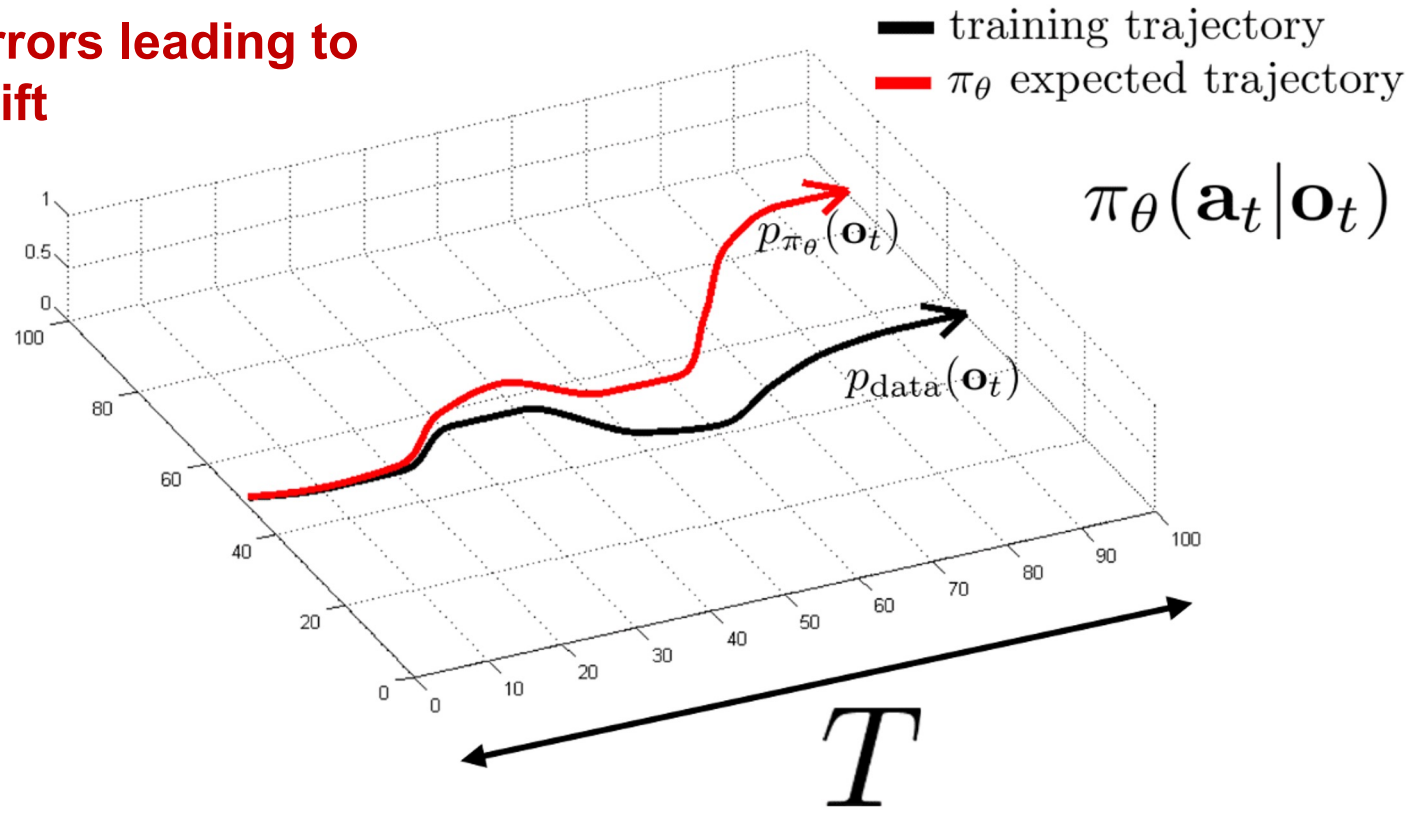
Downsides of behavioral cloning



slide credit: Scott Niekum

Challenges in imitation learning #1

Compounding Errors leading to Distributional Shift



Quadratic Regret

Regret (in decision theory) measures the difference between the reward (or outcome) one actually received and the best possible reward one *could have received* if one had made the optimal choice

$$\hat{\pi}_{sup} = \arg\min_{\pi \in \Pi} \mathbb{E}_{s \sim d_{\pi^*}} [\ell(s, \pi)]$$

Assuming $\ell(s, \pi)$ is the 0-1 loss (or upper bound on the 0-1 loss) implies the following performance guarantee with respect to any task cost function C bounded in $[0, 1]$:

Theorem 2.1. *(Ross and Bagnell, 2010) Let* $\mathbb{E}_{s \sim d_{\pi^*}} [\ell(s, \pi)] = \epsilon$, *then* $J(\pi) \leq J(\pi^*) + T^2 \epsilon$.

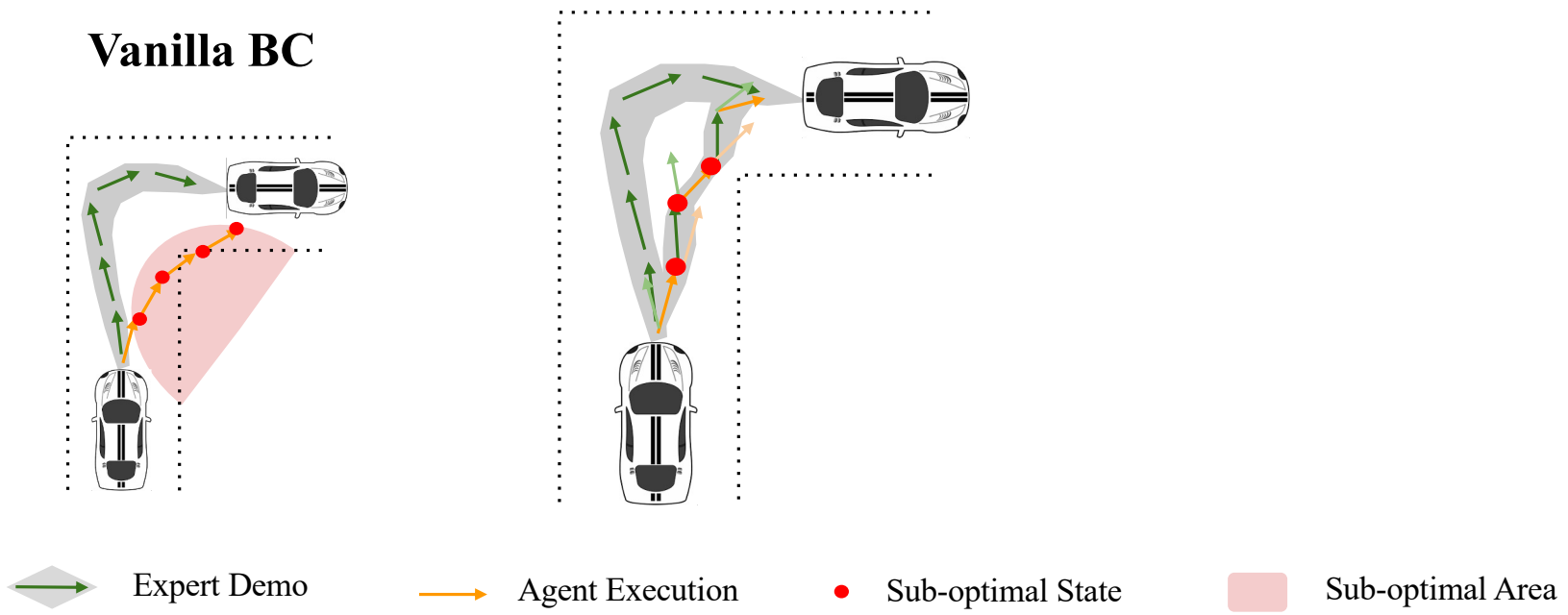
Compare to typical supervised learning loss that grows as: $O(\epsilon T)$

slide credit: Scott Niekum

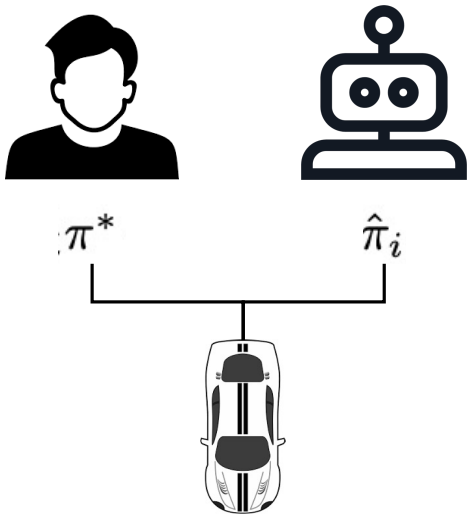
Ross, Stéphane, Geoffrey Gordon, and Drew Bagnell. "**A reduction of imitation learning and structured prediction to no-regret online learning.**" *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011.

DAgger: Dataset Aggregation

End-to-End Control Tasks



DAgger



Initialize $\mathcal{D} \leftarrow \emptyset$.
Initialize $\hat{\pi}_1$ to any policy in Π.
for $i = 1$ **to** N **do**
  Let $\pi_i = \beta_i \pi^* + (1 - \beta_i)\hat{\pi}_i$.
  Sample T-step trajectories using π_i.
  Get dataset $\mathcal{D}_i = \{(s, \pi^*(s))\}$ of visited states by π_i and actions given by expert.
  Aggregate datasets: $\mathcal{D} \leftarrow \mathcal{D} \bigcup \mathcal{D}_i$.
  Train classifier $\hat{\pi}_{i+1}$ on $\mathcal{D}$.
end for
Return best $\hat{\pi}_i$ on validation.

Algorithm 3.1: DAGGER Algorithm.

Key idea: keep collecting demonstration data that is on-distribution for current policy, and reduce dependence on expert over time

slide credit: Scott Niekum

Ross, Stéphane, Geoffrey Gordon, and Drew Bagnell. "**A reduction of imitation learning and structured prediction to no-regret online learning.**" *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011.

DAgger

Theorem 2.2. *Let π be such that $\mathbb{E}_{s\sim d_\pi}[\ell(s,\pi)] = \epsilon$, and $Q^{\pi^*}_{T-t+1}(s,a) - Q^{\pi^*}_{T-t+1}(s,\pi^*) \leq u$ for all action a, $t \in \{1,2,\ldots,T\}$, $d^t_\pi(s) > 0$, then $J(\pi) \leq J(\pi^*) + uT\epsilon$.*

If difference between optimal t-step Q and any other action is u (e.g. the worst single action regret):
The end cost is (no) worse than optimal plus number of mistakes times u, the worst possible regret of each mistake

slide credit: Scott Niekum

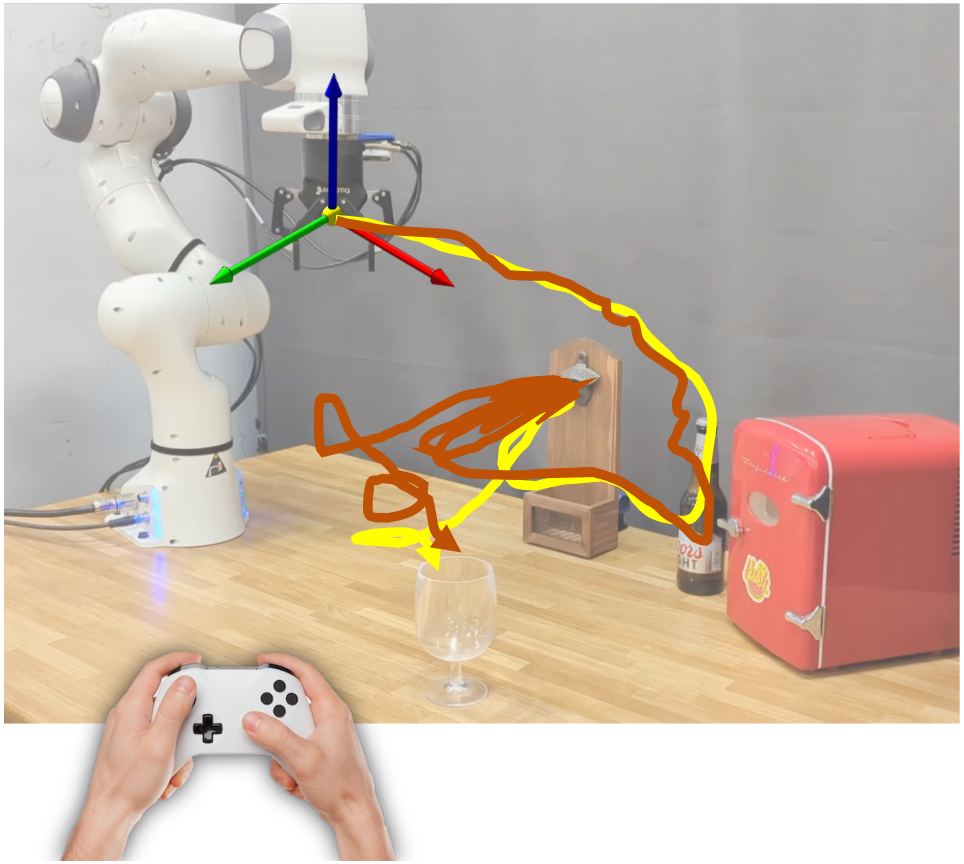
Ross, Stéphane, Geoffrey Gordon, and Drew Bagnell. "**A reduction of imitation learning and structured prediction to no-regret online learning.**" *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011.

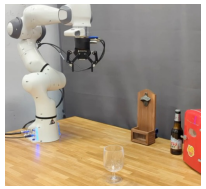
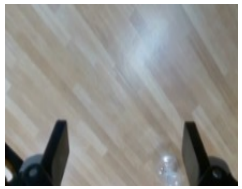
What are some other ways to improve robustness?

Robustness refers to the ability of a system, model, or method to maintain performance or produce reliable results **despite variations, noise, errors, or adversarial conditions** in the input or environment.

$$J(\pi) \leq J(\pi^*) + T^2\epsilon.$$

Behavioral Cloning



$s = ($ Third-person View , Wrist-camera View , proprio.$)$

$a = (\Delta x, \Delta y, \Delta z, \Delta rx, \Delta ry, \Delta rz, close\_gripper)$

Gripper Velocity Grasping?

$D = \{(s, a)\}$

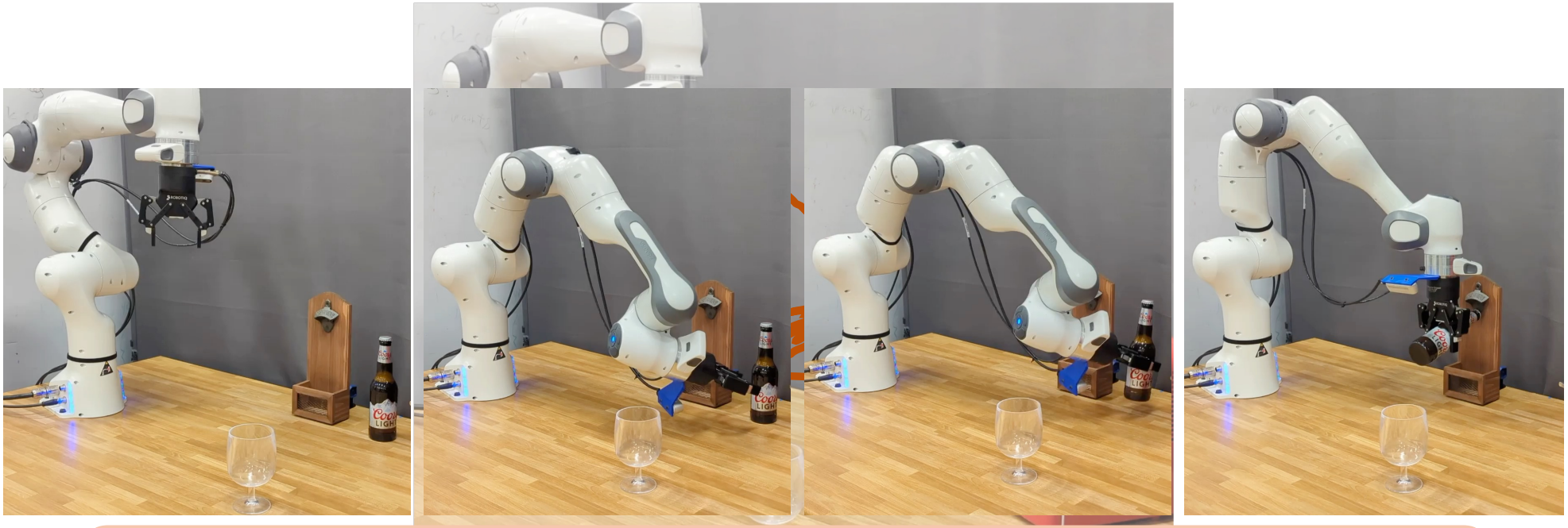
Policy (θ)

a_t

action

Input State

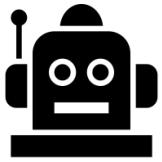
$$\mathcal{L}(\theta) = -\mathbb{E}_{(s,a)\sim D}[\log \pi_\theta(a|s)]$$



reach bottle *pick up bottle* *move bottle to opener* *open bottle*

$$a = (\Delta x, \Delta y, \Delta z, \Delta rx, \Delta ry, \Delta rz, gripper)$$

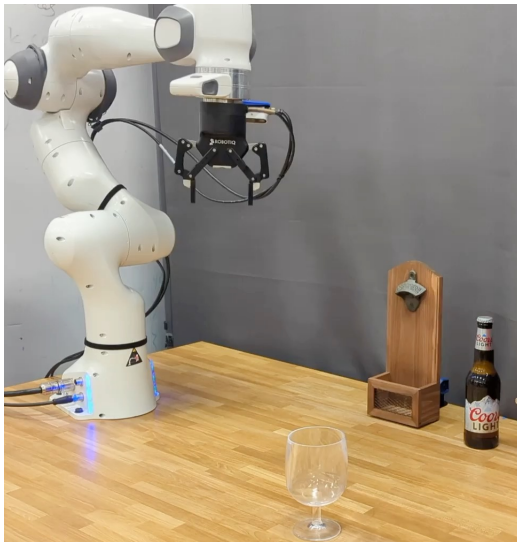
$a_0\, a_1\, a_2 \ldots \quad \ldots a_k \quad \ldots\ aT$



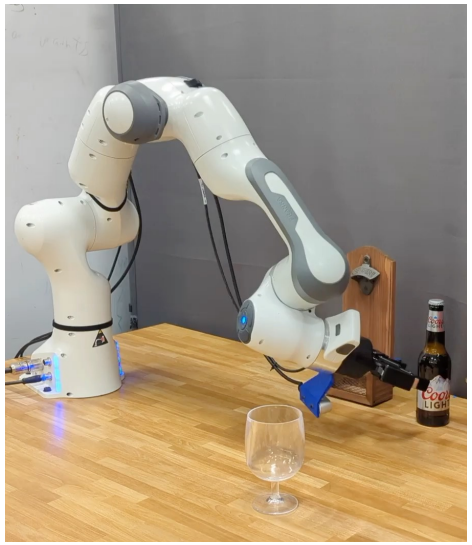
Unintended low-level motions constitute noise in demonstrations.

The demonstrator's **high-level actions** are optimal!

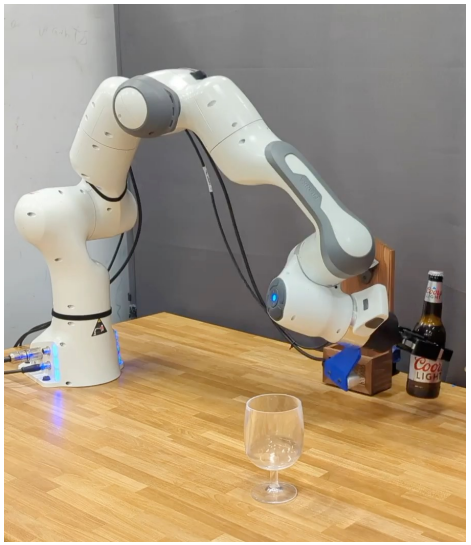
Free-space Reaching



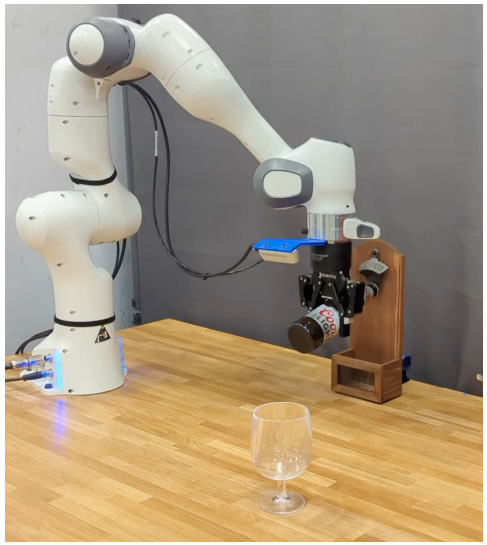
Contact-Rich Interaction



Free-space Reaching



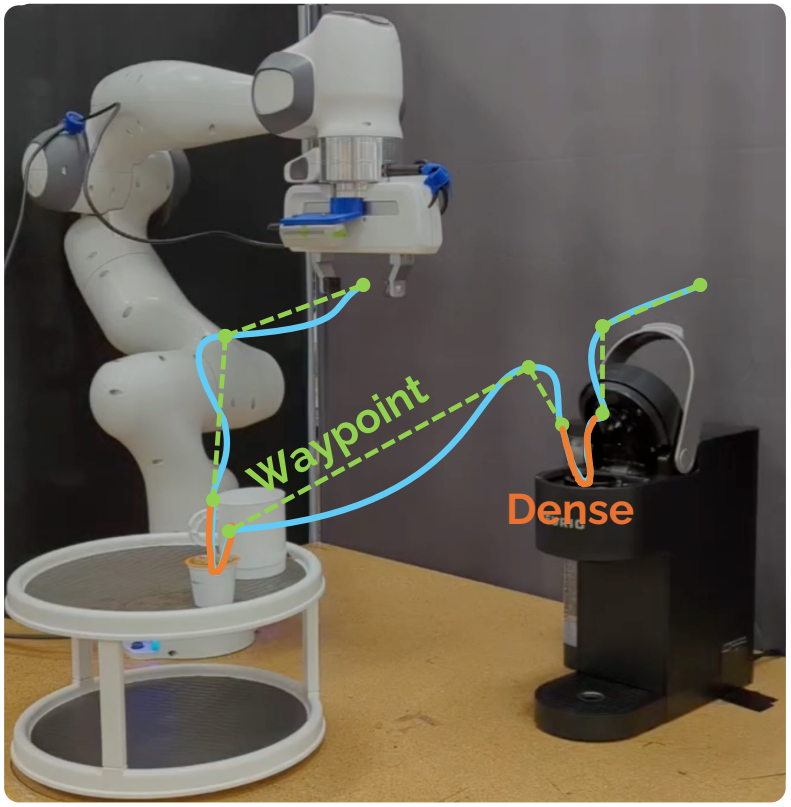
Contact-Rich Interaction



The demonstrator's **high-level actions** are optimal!

These actions can be categorized into ***two general modes***.

HYDRA: Hybrid Robot Actions for Imitation Learning in Manipulation



$$D = \{(s, a, w, m)\}$$

$s = (RGB_{external},\ RGB_{wrist},\ proprio.)$

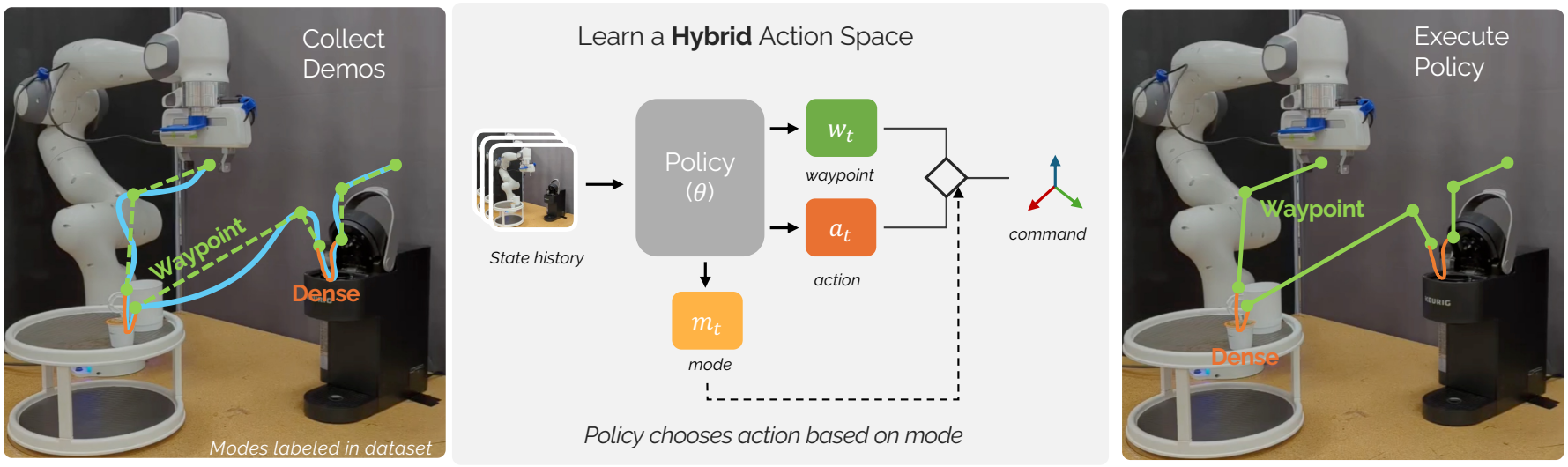
$a = (\Delta x, \Delta y, \Delta z, \Delta rx, \Delta ry, \Delta rz, close\_gripper)$

$w = (x, y, z, rx, ry, rz, gripper\_state)$

$m = 0$ if waypoint, else 1

Belkhale, S., **Cui, Y.**, & Sadigh, D. (2023, December). Hydra: Hybrid robot actions for imitation learning. In Conference on Robot Learning (pp. 2113-2133). PMLR.

HYDRA: Hybrid Robot Actions for Imitation Learning in Manipulation



$$\mathcal{L}_a(\theta) = -\mathbb{E}_{(s,a,w,m)\sim D}[(1-\alpha)\log \pi_\theta^A(a|s) + \alpha \log \pi_\theta^W(w|s)]$$

$$\mathcal{L}_m(\theta) = -\mathbb{E}_{(s,a,w,m)\sim D}[(m \log \pi_\theta^M(m=1|s) + (1-m)\log \pi_\theta^M(m=0|s)]$$

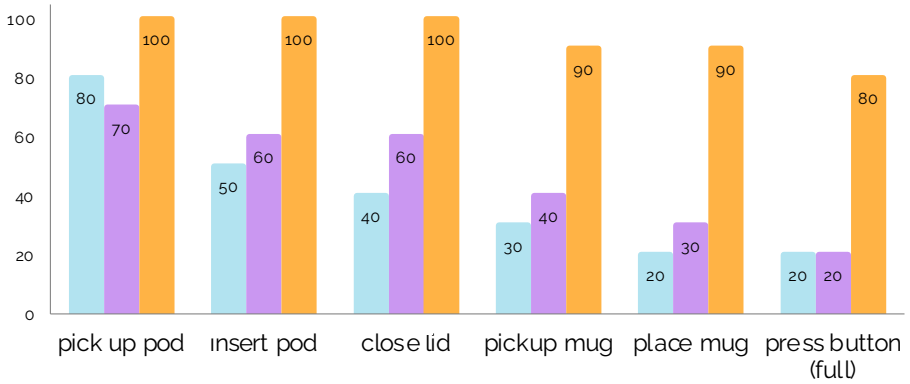
Belkhale, S., **Cui, Y.**, & Sadigh, D. (2023, December). Hydra: Hybrid robot actions for imitation learning. In Conference on Robot Learning (pp. 2113-2133). PMLR.

Make Coffee

BC-RNN — Mandlekar et al. What Matters in Learning from Offline Human Demonstrations for Robot Manipulation. CoRL'21

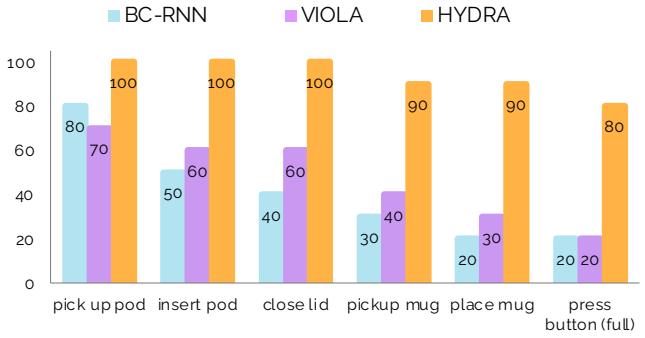
VIOLA — Zhu et al. "VIOLA: Object-Centric Imitation Learning for Vision-Based Robot Manipulation." CoRL'22

HYDRA — OURS

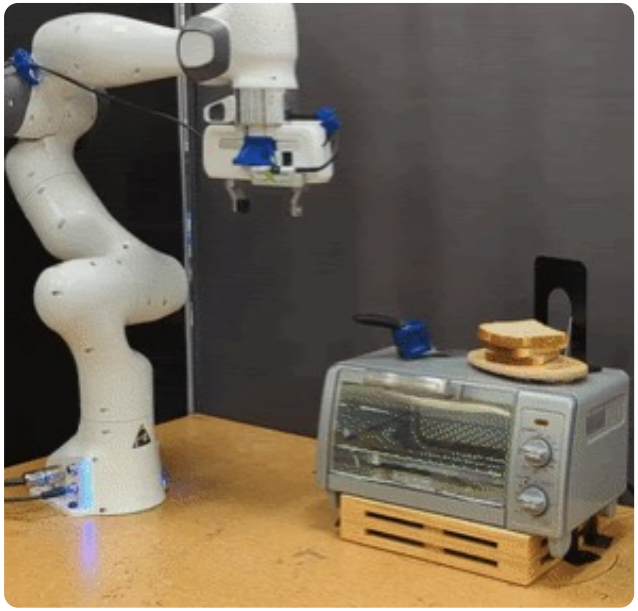


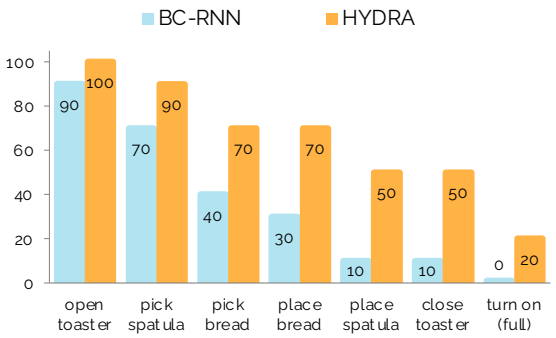
Belkhale, S., **Cui, Y.**, & Sadigh, D. (2023, December). Hydra: Hybrid robot actions for imitation learning. In Conference on Robot Learning (pp. 2113-2133). PMLR.

Make Coffee

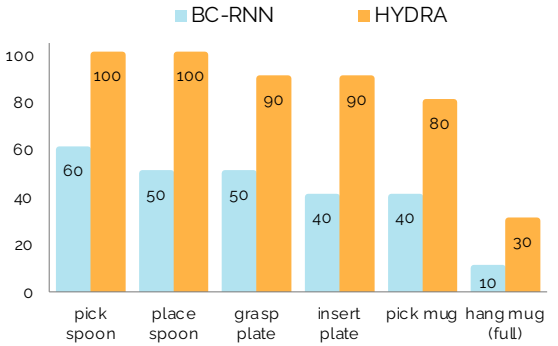


Make Toast



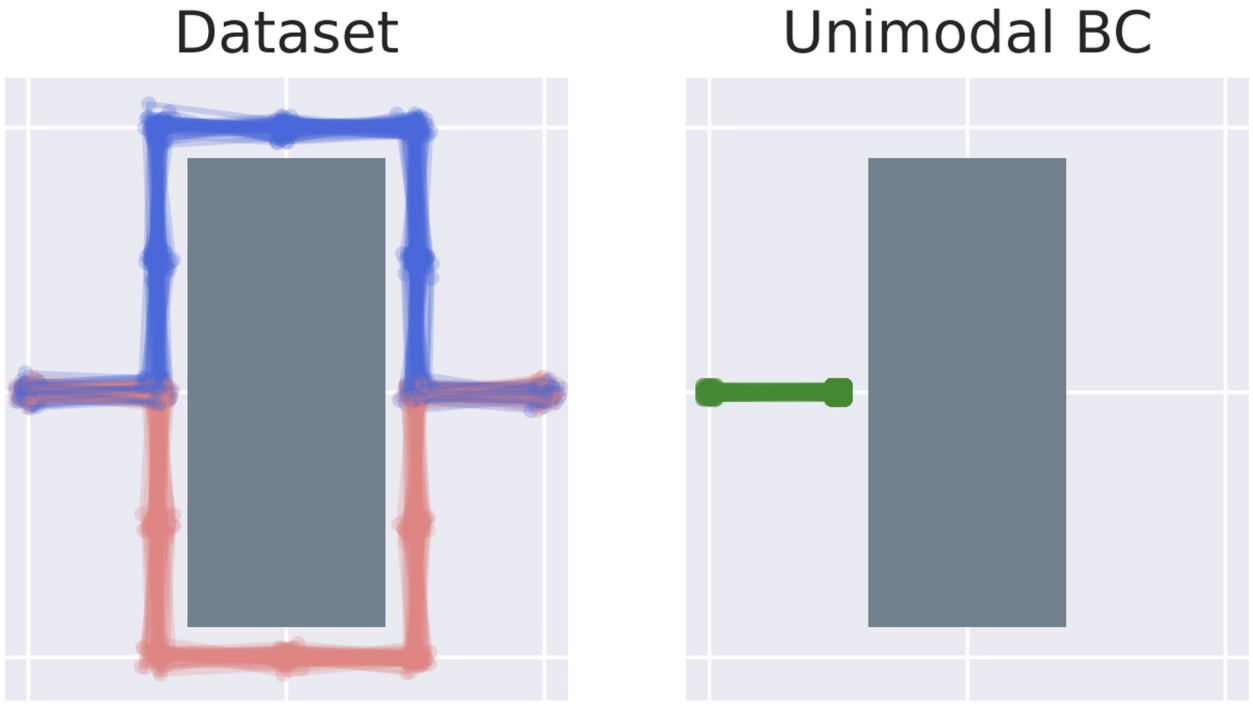


Sort Dishes



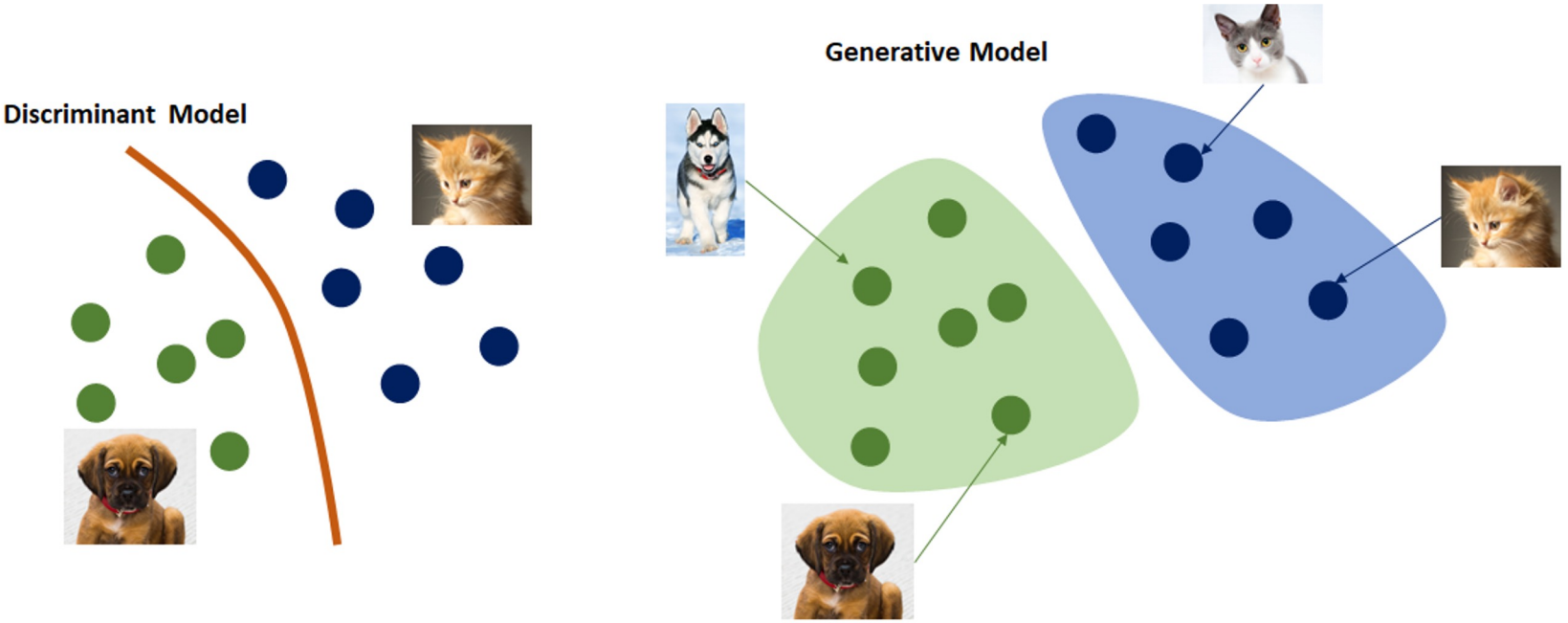
Belkhale, S., **Cui, Y**., & Sadigh, D. (2023, December). Hydra: Hybrid robot actions for imitation learning. In Conference on Robot Learning (pp. 2113-2133). PMLR.

Challenges in imitation learning #2

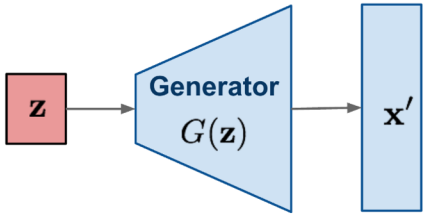


Multimodal demonstrations

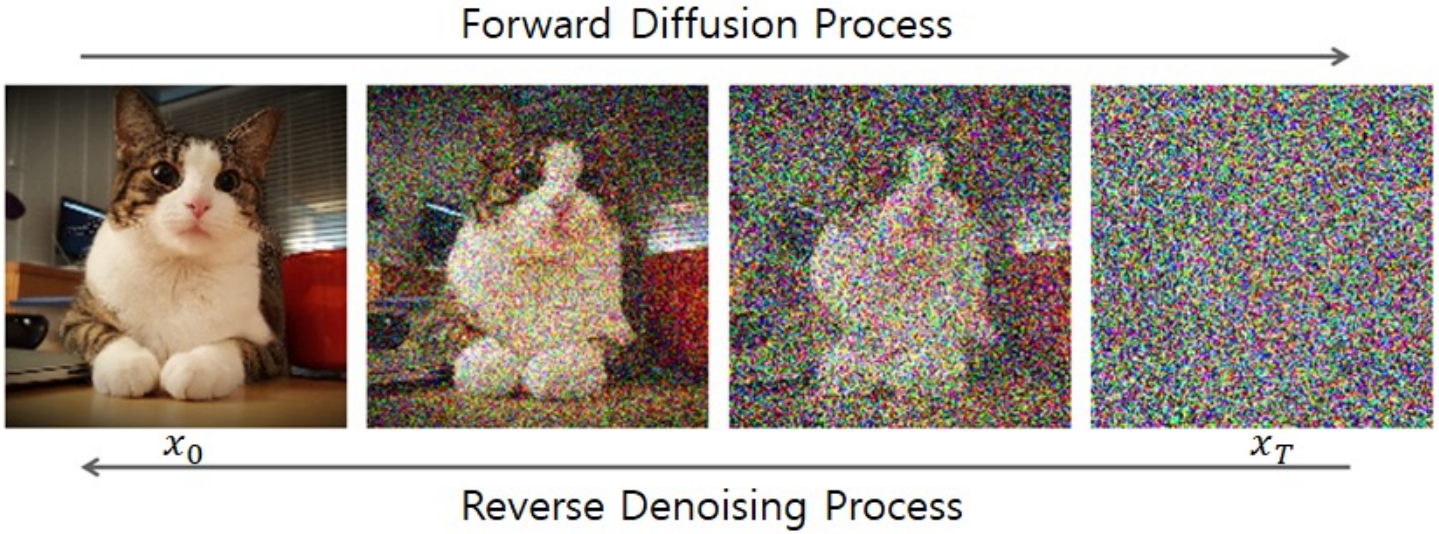
Generative Modeling



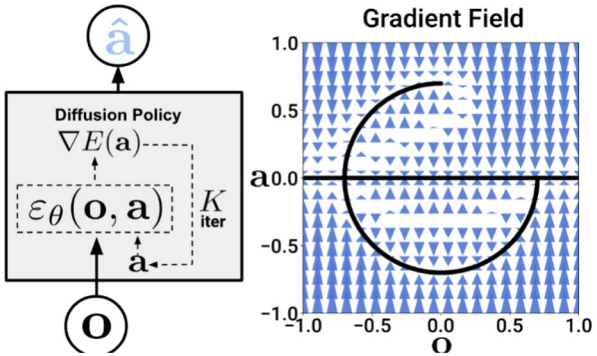
Generative Models

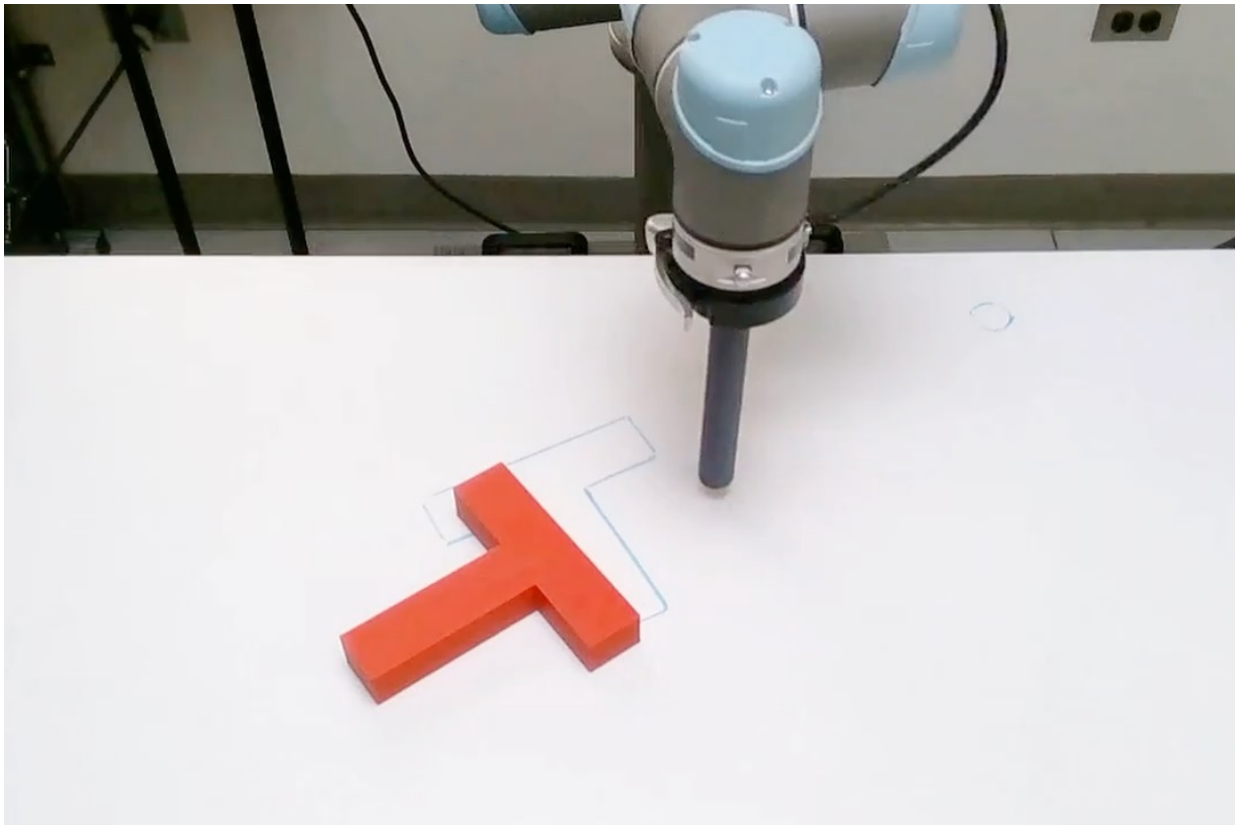


Diffusion


Forward Diffusion Process
x_0
x_T
Reverse Denoising Process

Diffusion Policy

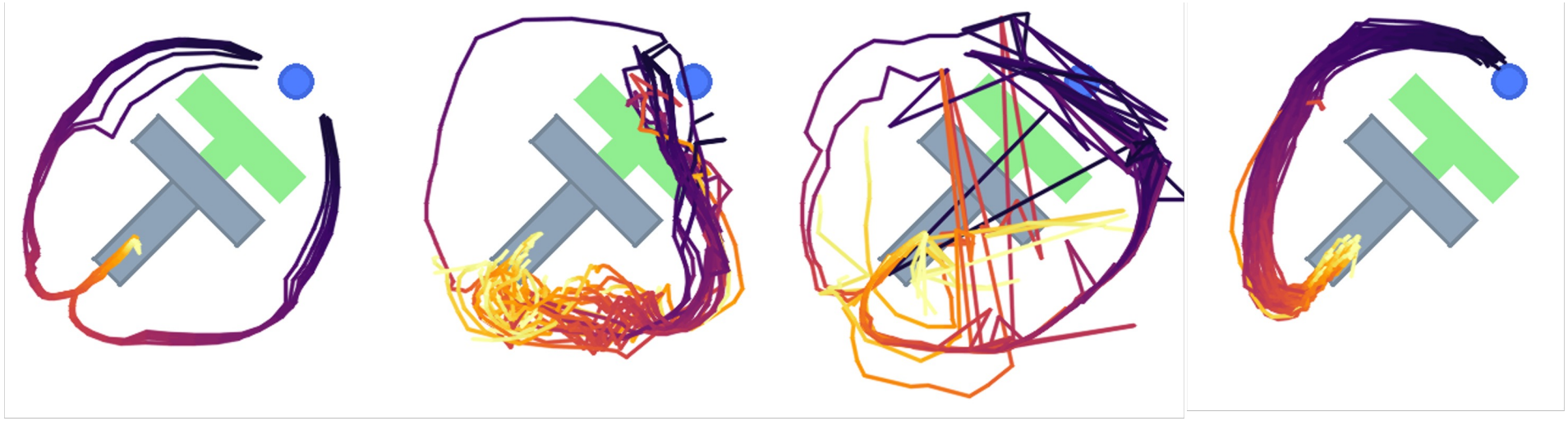




https://diffusion-policy.cs.columbia.edu/

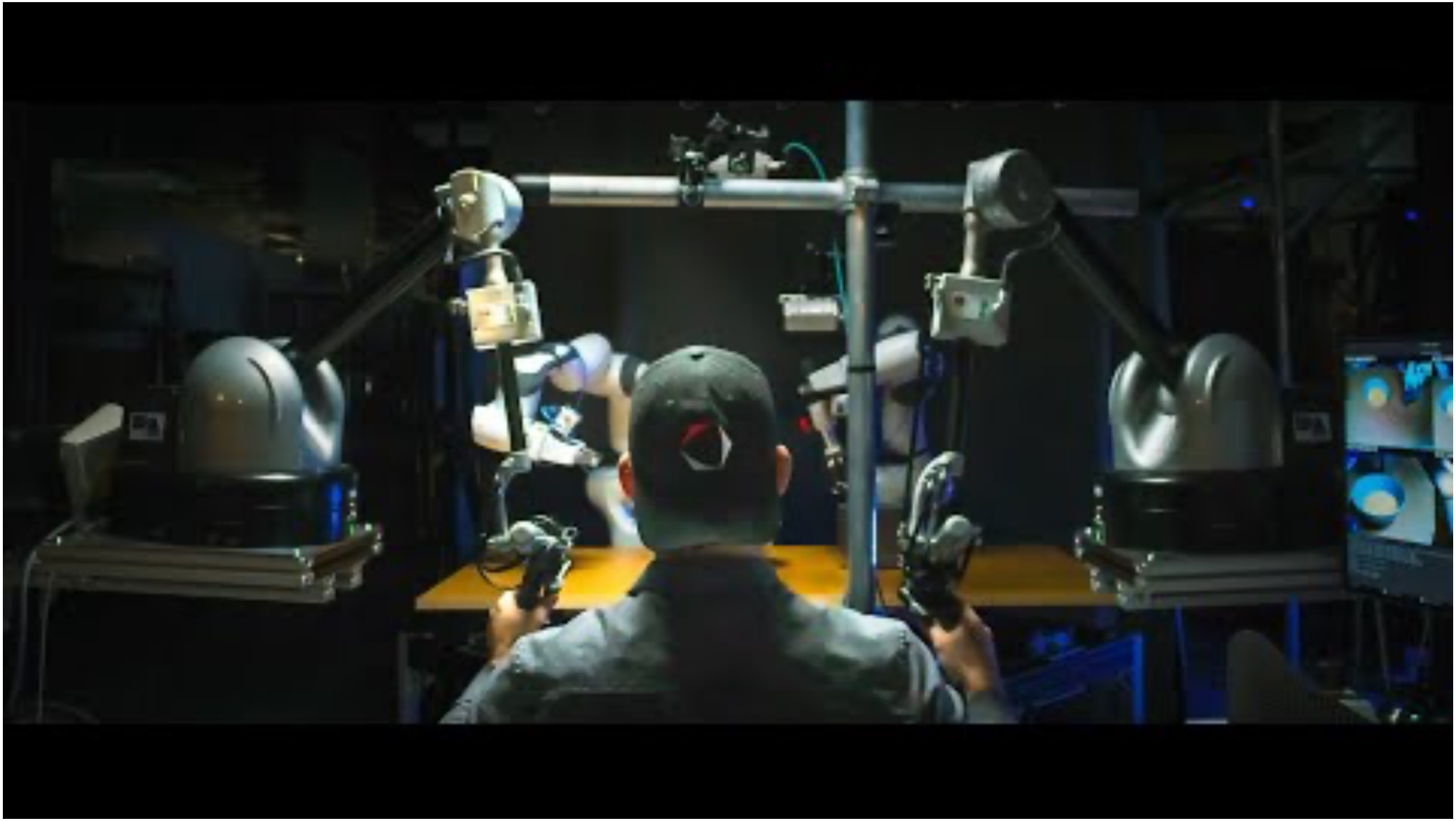
Chi, Cheng, et al. "Diffusion policy: Visuomotor policy learning via action diffusion." *The International Journal of Robotics Research* (2023): 02783649241273668.

Diffusion Policy

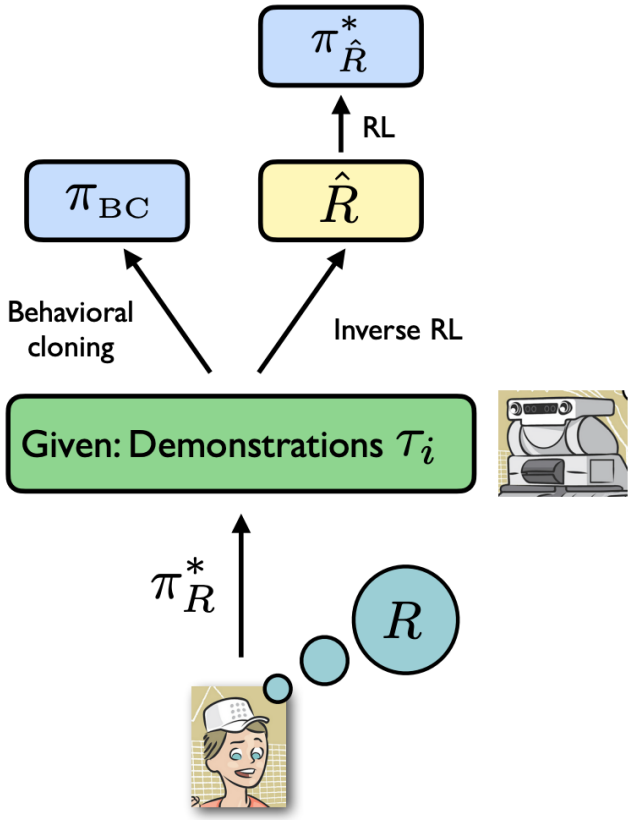


Diffusion Policy LSTM-GMM BET IBC

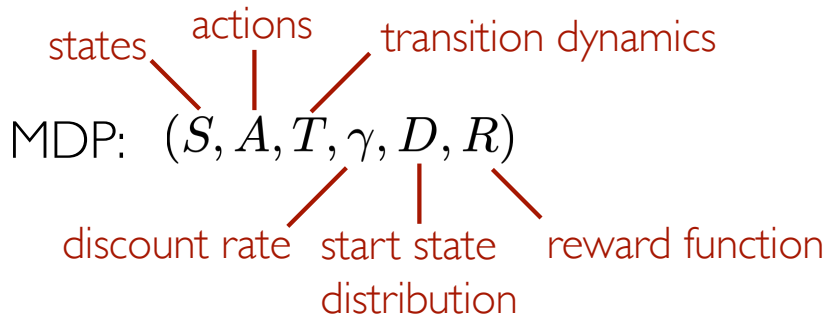
Chi, Cheng, et al. "Diffusion policy: Visuomotor policy learning via action diffusion." *The International Journal of Robotics Research* (2023): 02783649241273668.



Imitation Learning



Inverse reinforcement learning



Policy: $\pi(s, a) \rightarrow [0, 1]$

Value function: $V^{\pi}(s_0) = \sum_{t=0}^{\infty} \gamma^t R(s_t)$

What if we have an **MDP/R**?

slide credit: Scott Niekum

Inverse reinforcement learning

1. Collect user demonstration $(s_0, a_0), (s_1, a_1), \ldots, (s_n, a_n)$ and assume it is sampled from the expert's policy, π^E

2. Explain expert demos by finding R^* such that:

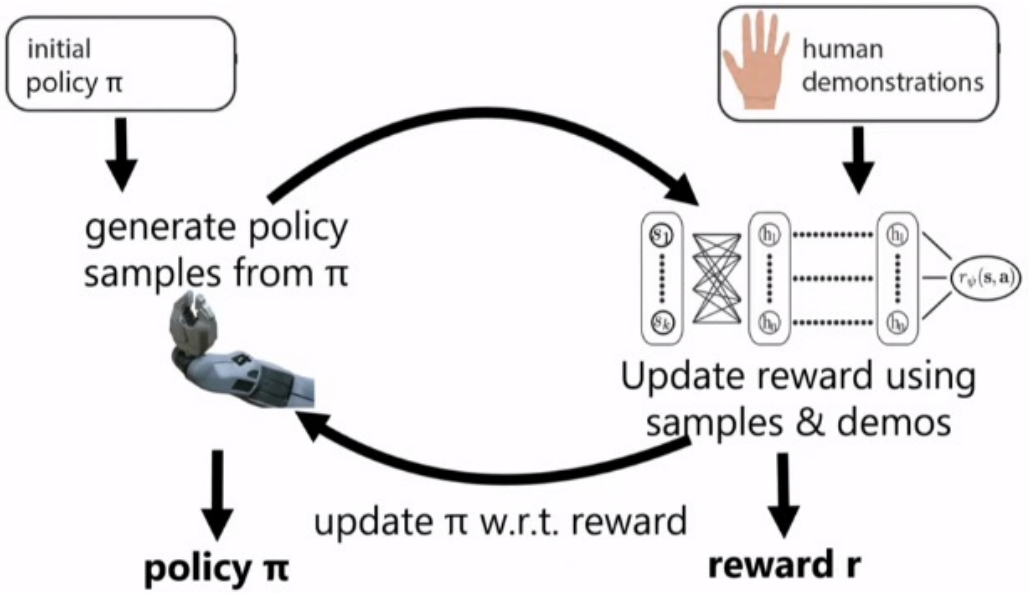
$$E[\textstyle\sum_{t=0}^{\infty} \gamma^t R^*(s_t)|\pi^E] \;\geq\; E[\textstyle\sum_{t=0}^{\infty} \gamma^t R^*(s_t)|\pi] \quad \forall \pi$$

$$E_{s_0 \sim D}[V^{\pi^E}(s_0)] \;\geq\; E_{s_0 \sim D}[V^{\pi}(s_0)] \quad \forall \pi$$

slide credit: Scott Niekum

Abbeel, Pieter, and Andrew Y. Ng. "Apprenticeship learning via inverse reinforcement learning." *Proceedings of the twenty-first international conference on Machine learning*. 2004.

Guided Cost Learning



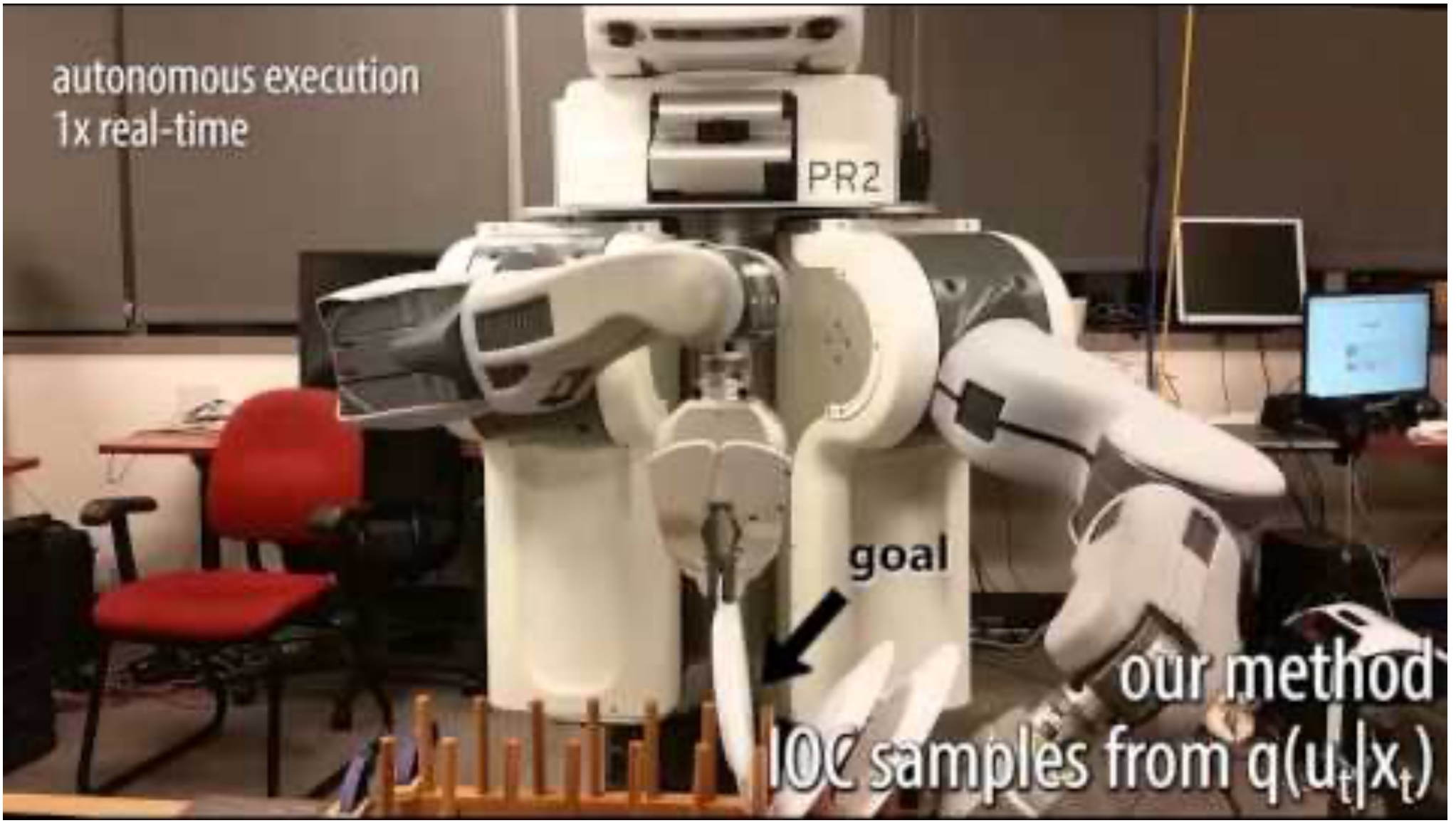
Assume we don't know the dynamics, but we can sample, like in standard RL

Algorithm 2 Nonlinear IOC with stochastic gradients

1: **for** iteration $k = 1$ to K **do**
2: Sample demonstration batch $\hat{\mathcal{D}}_{\text{demo}} \subset \mathcal{D}_{\text{demo}}$
3: Sample background batch $\hat{\mathcal{D}}_{\text{samp}} \subset \mathcal{D}_{\text{samp}}$
4: Append demonstration batch to background batch: $\hat{\mathcal{D}}_{\text{samp}} \leftarrow \hat{\mathcal{D}}_{\text{demo}} \cup \hat{\mathcal{D}}_{\text{samp}}$
5: Estimate $\frac{d\mathcal{L}_{\text{IOC}}}{d\theta}(\theta)$ using $\hat{\mathcal{D}}_{\text{demo}}$ and $\hat{\mathcal{D}}_{\text{samp}}$
6: Update parameters θ using gradient $\frac{d\mathcal{L}_{\text{IOC}}}{d\theta}(\theta)$
7: **end for**
8: **return** optimized cost parameters θ

slide credit: Sergey Levine

Finn, Chelsea, Sergey Levine, and Pieter Abbeel. "Guided cost learning: Deep inverse optimal control via policy optimization." *International conference on machine learning*. PMLR, 2016.


autonomous execution
1x real-time
PR2
goal
our method
100 samples from $q(u_t|x_t)$


That’s it for today!
Questions?