

Lecture 10

Sources of human feedback



Last Time

[✓] embedding predictive human models into safety

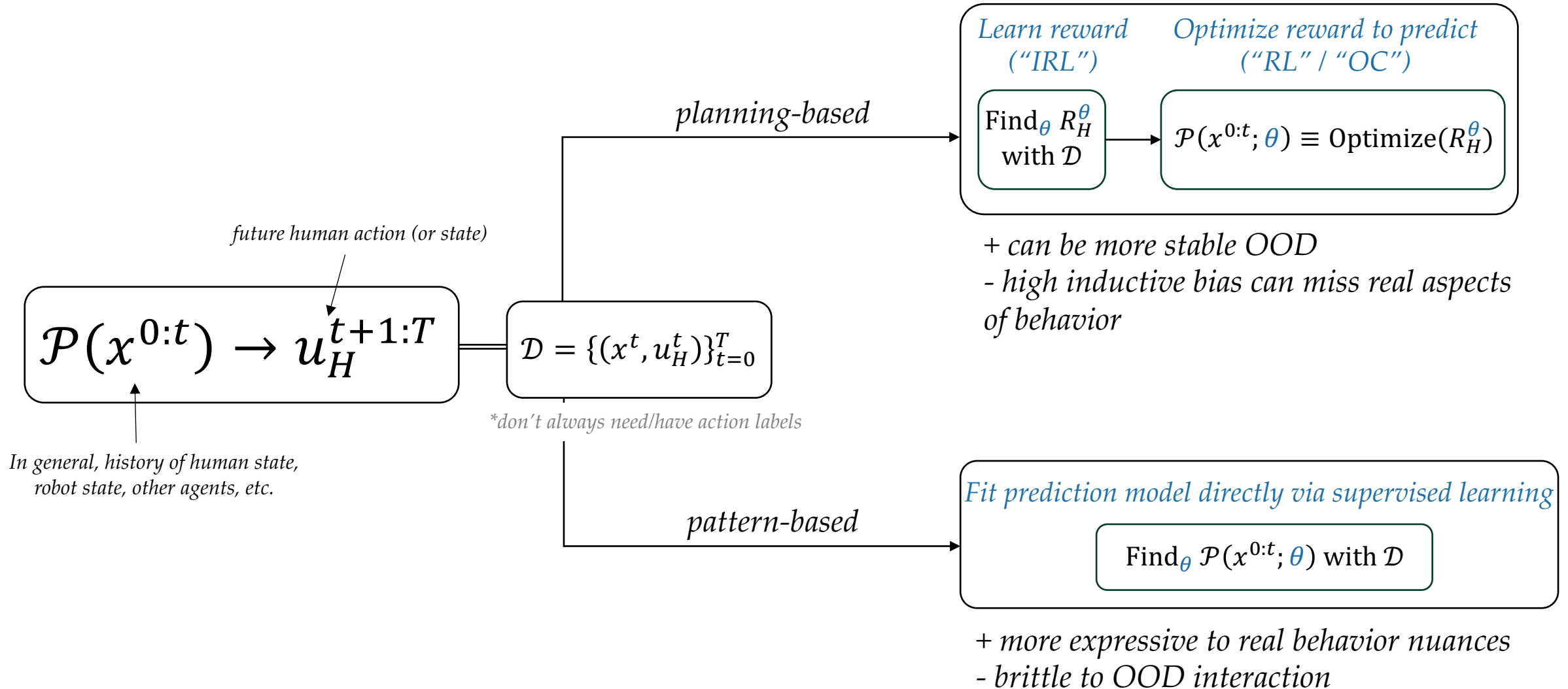
This Time

[] sources of human data

[] robot learning from corrections

[] a unifying formalism for learning from human data

So far... all about human behavior prediction



On complementing end-to-end human behavior predictors with planning

Liting Sun, Xiaogang Jia, and Anca D. Dragan
University of California, Berkeley

Abstract—High capacity end-to-end approaches for human motion (behavior) prediction have the ability to represent subtle nuances in human behavior, but struggle with robustness to out of distribution inputs and tail events. Planning-based prediction, on the other hand, can reliably output decent-but-not-great predictions: it is much more stable in the face of distribution shift (as we verify in this work), but it has high inductive bias, missing important aspects that drive human decisions, and ignoring cognitive biases that make human behavior suboptimal. In this work, we analyze one family of approaches that strive to get the best of both worlds: use the end-to-end predictor on common cases, but do not rely on it for tail events / out-of-distribution inputs – switch to the planning-based predictor there. We contribute an analysis of different approaches for detecting when to make this switch, using an autonomous driving domain. We find that promising approaches based on ensembling or generative modeling of the training distribution might not be reliable, but that there very simple methods which can perform surprisingly well – including training a classifier to pick up on tell-tale issues in predicted trajectories.

I. INTRODUCTION

Robots that need to share their environments with humans learn predictive models of human behavior, which they use to generate their own behavior in response.

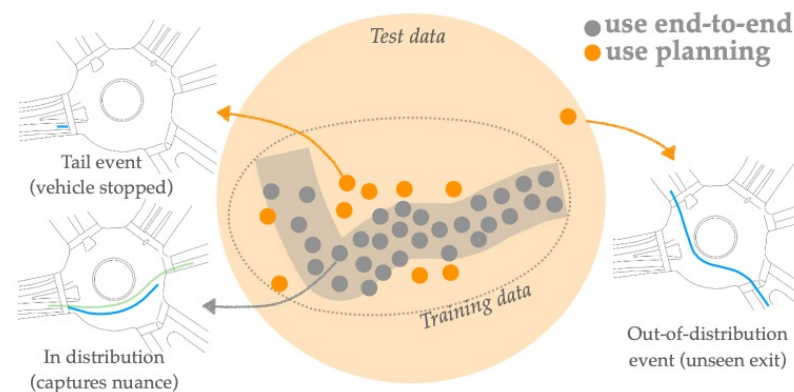
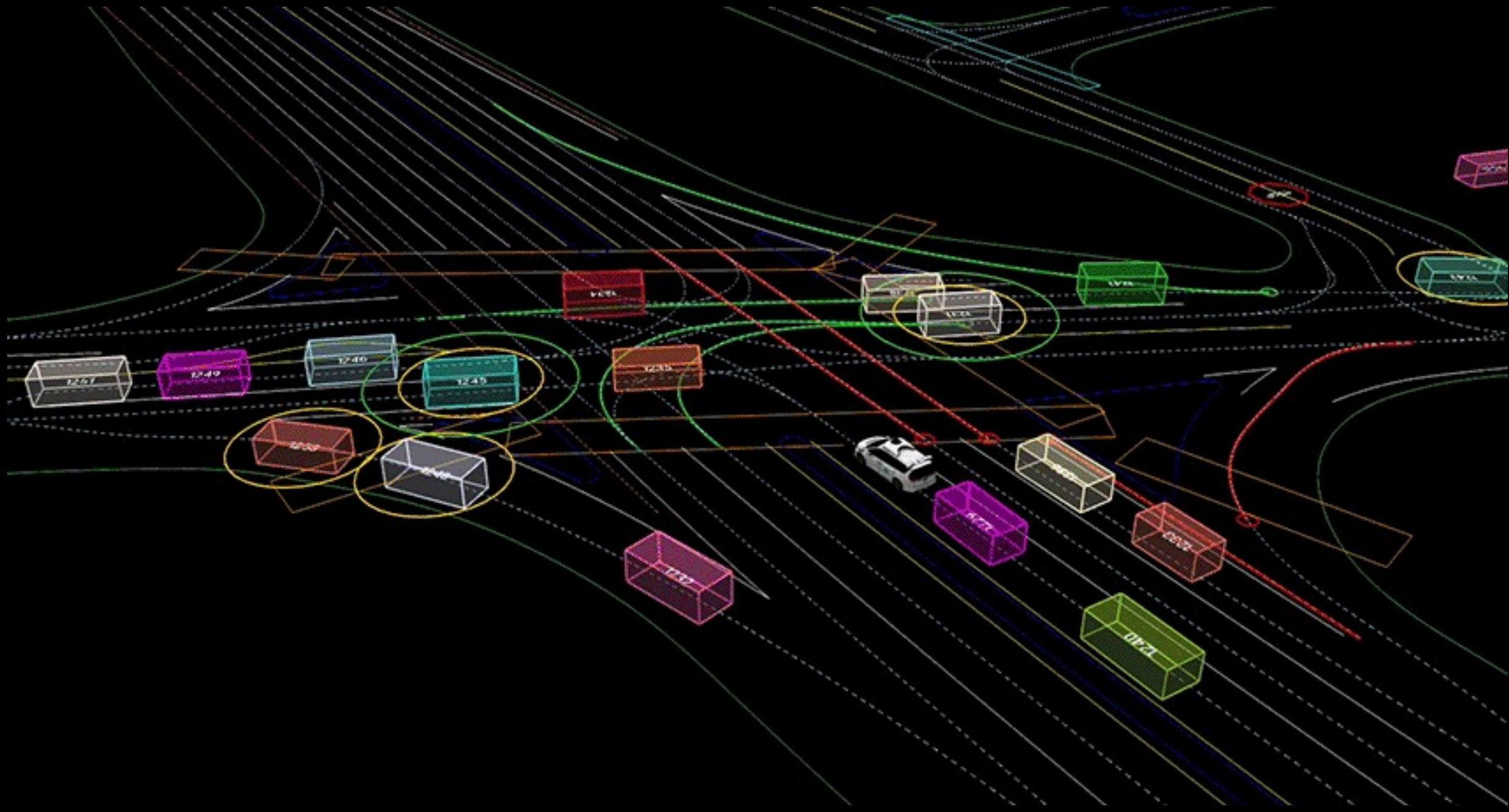


Fig. 1. We analyze methods for using an end-to-end predictor on common cases (gray region), and relying on planning-based prediction outside of that (orange region).

level, or anything else that influences where humans go that would be otherwise very challenging to explicitly write down.

But one challenge that such high capacity, end-to-end models face is their performance in the face of distribution shift or tail events. Our understanding of the nuances of this challenge is still evolving, but there seem



Source: <https://waymo.com/blog/2021/03/expanding-waymo-open-dataset-with-interactive-scenario-data-and-new-challenges/>

Mainprice, Jim, Rafi Hayne, and Dmitry Berenson. "Predicting human reaching motion in collaborative tasks using inverse optimal control and iterative re-planning." *International Conference on Robotics and Automation (ICRA)*, 2015.

<https://www.youtube.com/watch?v=w165L7ZtDws>

PREDICTING HUMAN REACHING MOTION IN COLLABORATIVE TASKS USING INVERSE OPTIMAL CONTROL AND ITERATIVE RE-PLANNING

JIM MAINPRICE, RAFI HAYNE, DMITRY BERENSON

0:00 / 2:12



WPI



ManiCast: Collaborative Manipulation with Cost-Aware Human Forecasting

Kushal Kedia
Cornell University

Prithwish Dan
Cornell University

Atiksh Bhardwaj
Cornell University

Sanjiban Choudhury
Cornell University

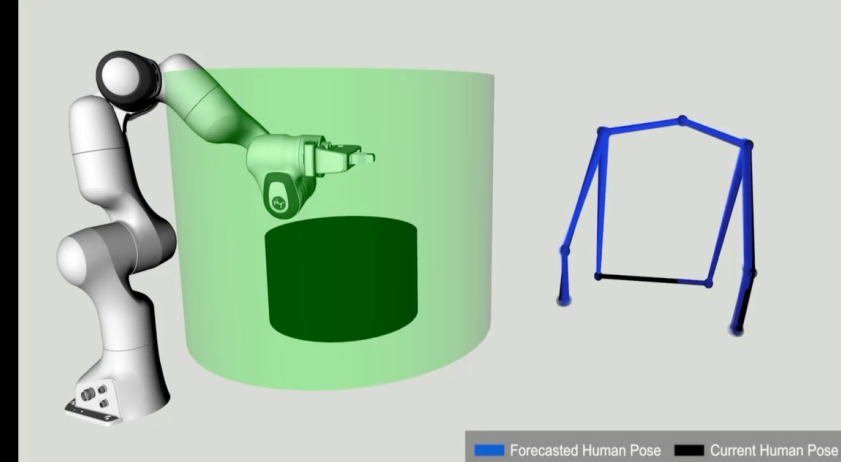
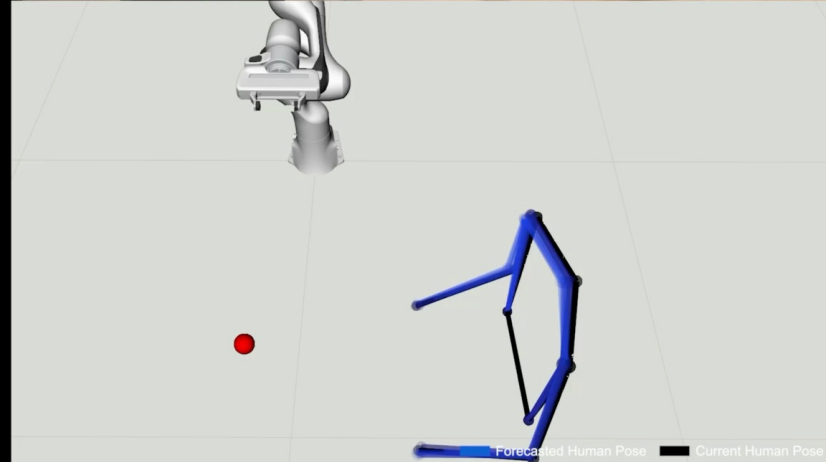
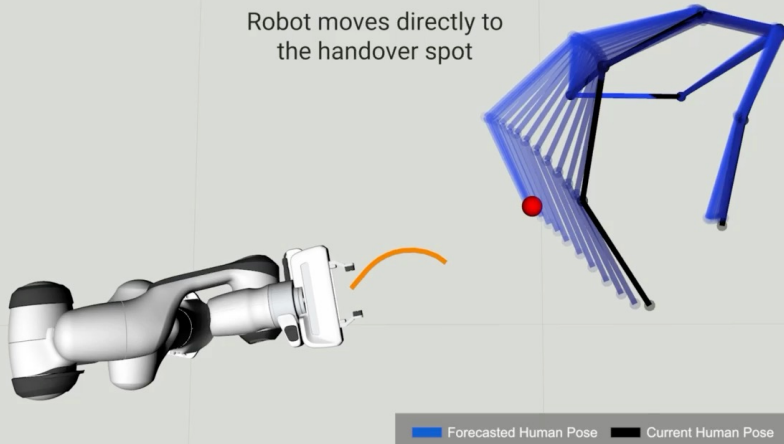
*

Abstract: Seamless human-robot manipulation in close proximity relies on accurate forecasts of human motion. While there has been significant progress in learning forecast models at scale, when applied to manipulation tasks, these models accrue high errors at critical transition points leading to degradation in downstream planning performance. Our key insight is that instead of predicting the most likely human motion, it is sufficient to produce forecasts that capture how future human motion would affect the cost of a robot’s plan. We present MANICAST, a novel framework that learns cost-aware human forecasts and feeds them to a model predictive control planner to execute collaborative manipulation tasks. Our framework enables fluid, real-time interactions between a human and a 7-DoF robot arm across a number of real-world tasks such as reactive stirring, object handovers, and collaborative table setting. We evaluate both the motion forecasts and

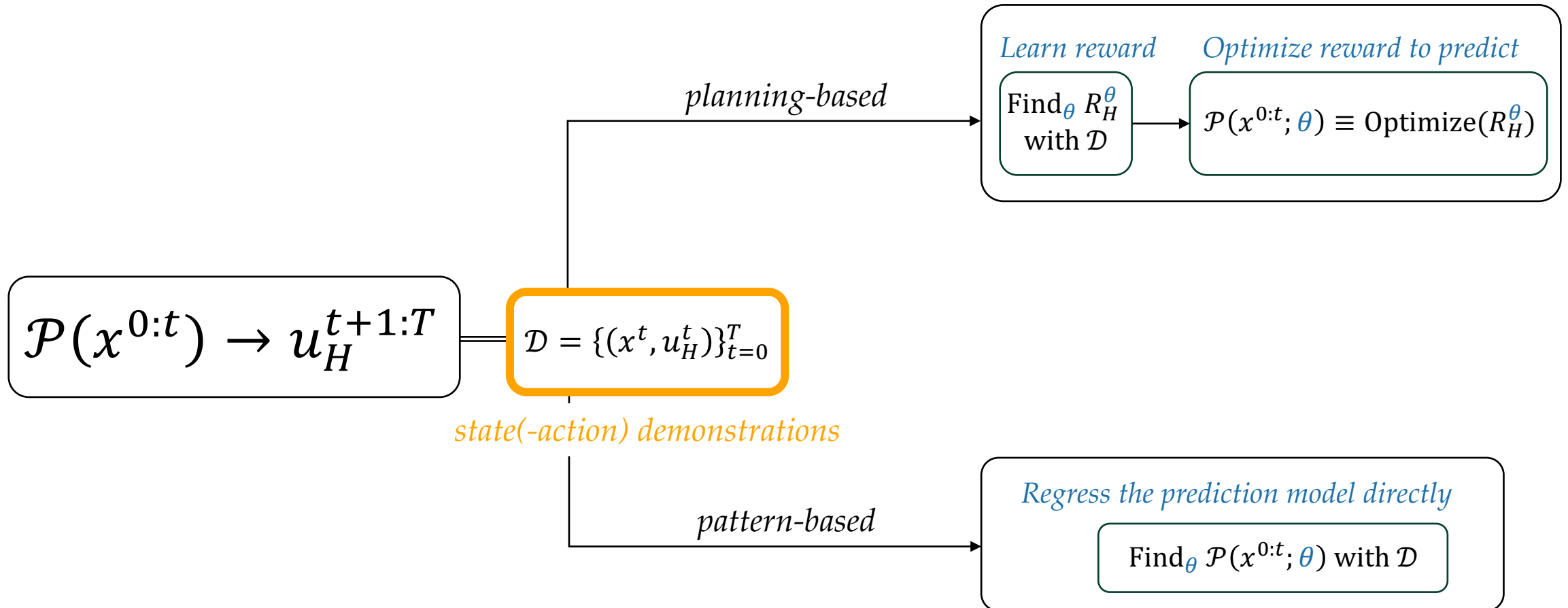
Planning with ManiCast



Robot moves directly to the handover spot



What type of **human data** have we studied so far?



What other kind of human data (or feedback)
can we leverage?

What other kind of human data (or feedback)
can we leverage?

Demonstrations

Corrections

Comparisons
(preferences)

“Initial state”
(i.e., preferences implicit in the state of the world)

Proxy reward

Language

Off-switch

... (and more)

What other kind of human data (or feedback) can we leverage?

[1] Andrew Y Ng and Stuart J Russell. "Algorithms for inverse reinforcement learning." ICML, 2000.

[2] Wirth, Christian, et al. "A survey of preference-based reinforcement learning methods." JMLR, 2017

Demonstrations

Comparisons
(preferences)

Proxy reward

Off-switch

[3] Hadfield-Menell, Dylan, et al. "Inverse reward design." *Neurips* 2017.

[4] Hadfield-Menell, Dylan, et al. "The off-switch game." Workshops at AAI, 2017.

Corrections

"Initial state"
(i.e., preferences implicit in the state of the world)

Language

... (and more)

[5] Bajcsy, Andrea, et al. "Learning robot objectives from physical human interaction." CoRL, 2017.

[6] Shah, Rohin, et al. "Preferences implicit in the state of the world." *ICLR*, 2019.

[7] Matuszek, Cynthia, et al. "A joint model of language and perception for grounded attribute learning." ICML, 2012.

What other kind of human data (or feedback) can we leverage?

Demonstrations

Corrections

*Next Wednesday: RLHF
(i.e., alignment)*

Comparisons
(preferences)

“Initial state”
(i.e., preferences implicit in the state of the world)

Proxy reward

Language

Off-switch

... (and more)

What other kind of human data (or feedback) can we leverage?

Demonstrations

Corrections

Today

Comparisons
(preferences)

“Initial state”
(i.e., preferences implicit in the state of the world)

Proxy reward

Language

Off-switch

... (and more)

What other kind of human data (or feedback) can we leverage?

Today: Unifying Framework

Demonstrations

Corrections

Comparisons
(preferences)

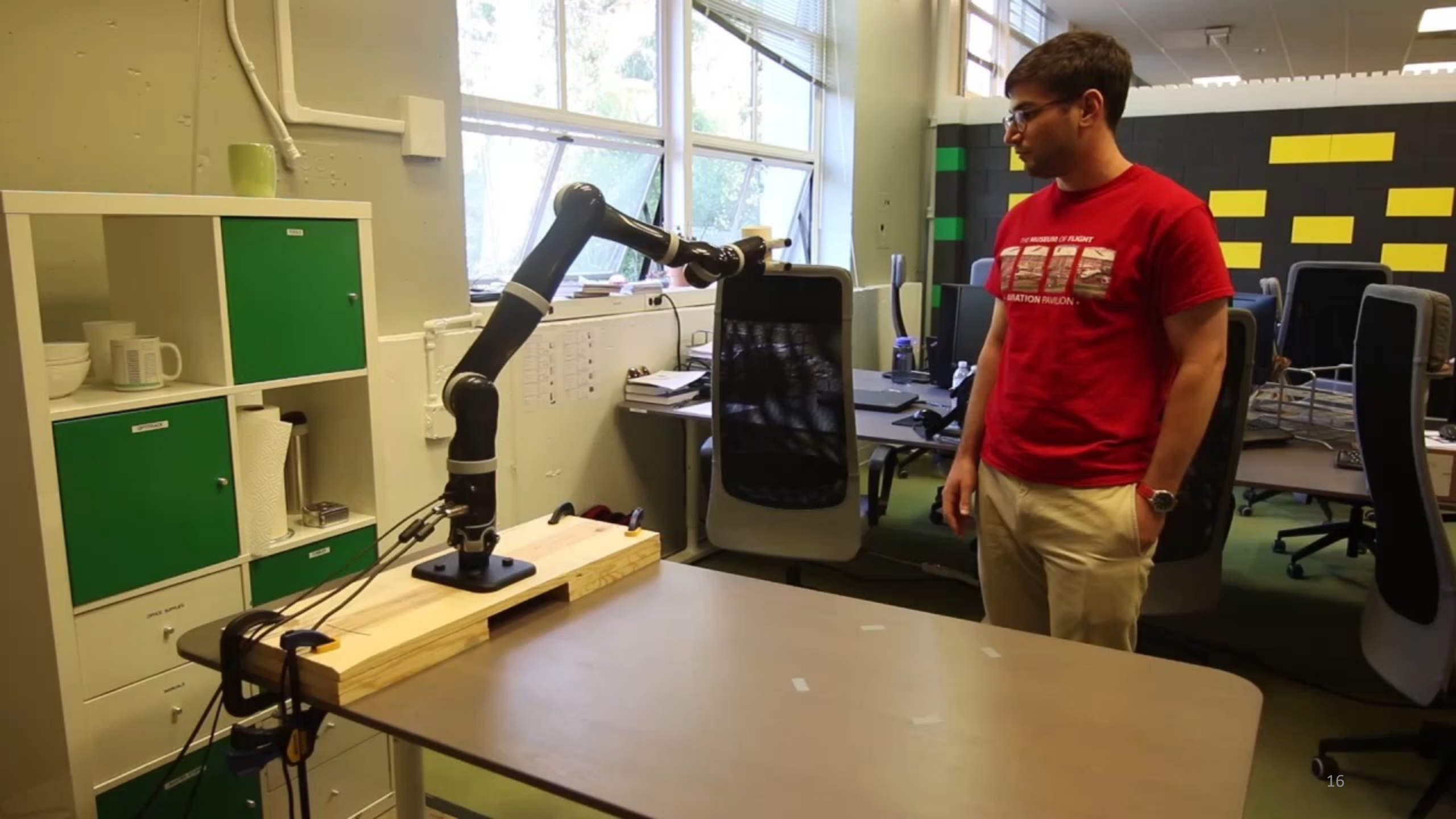
“Initial state”
(i.e., preferences implicit in the state of the world)

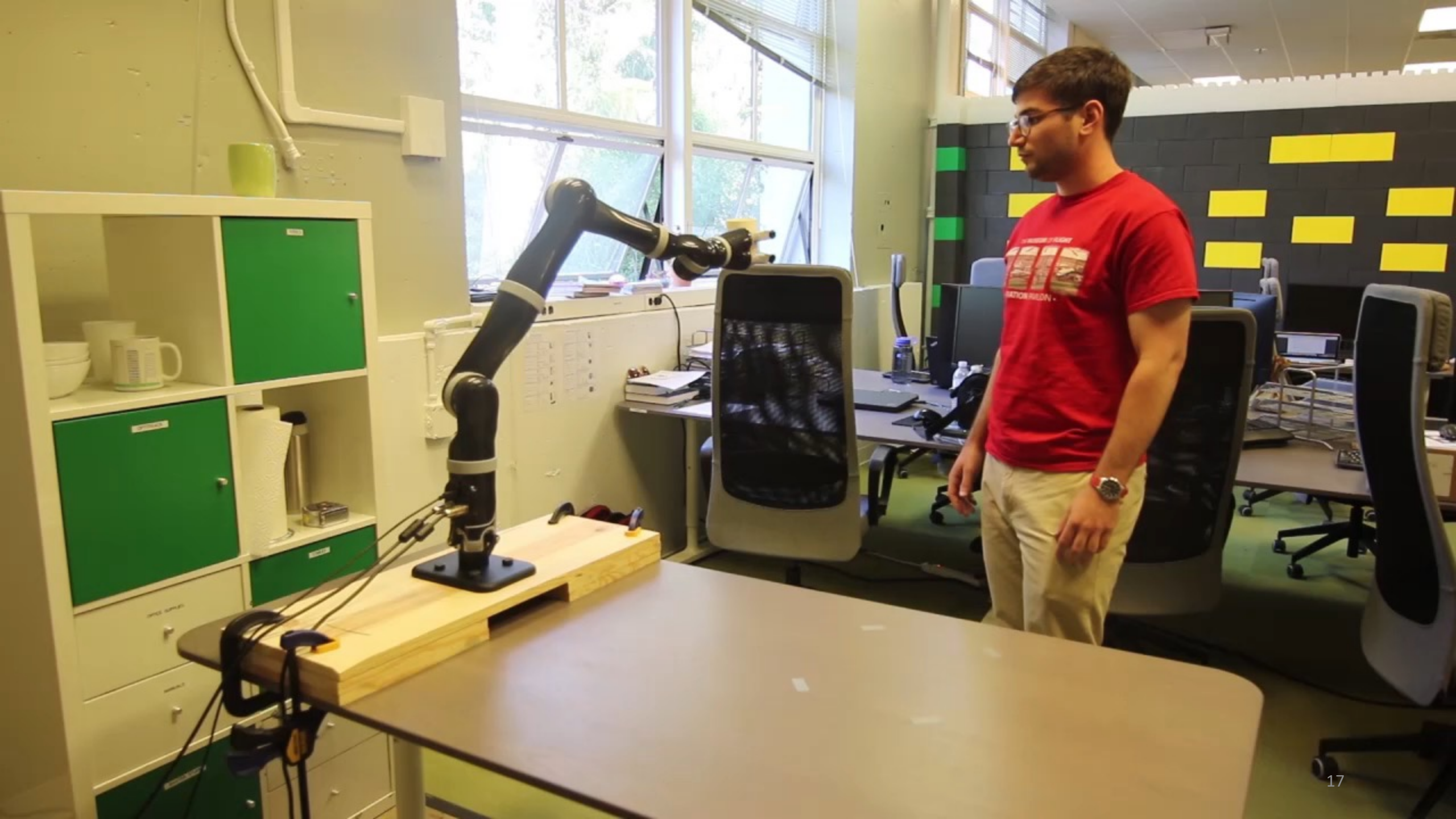
Proxy reward

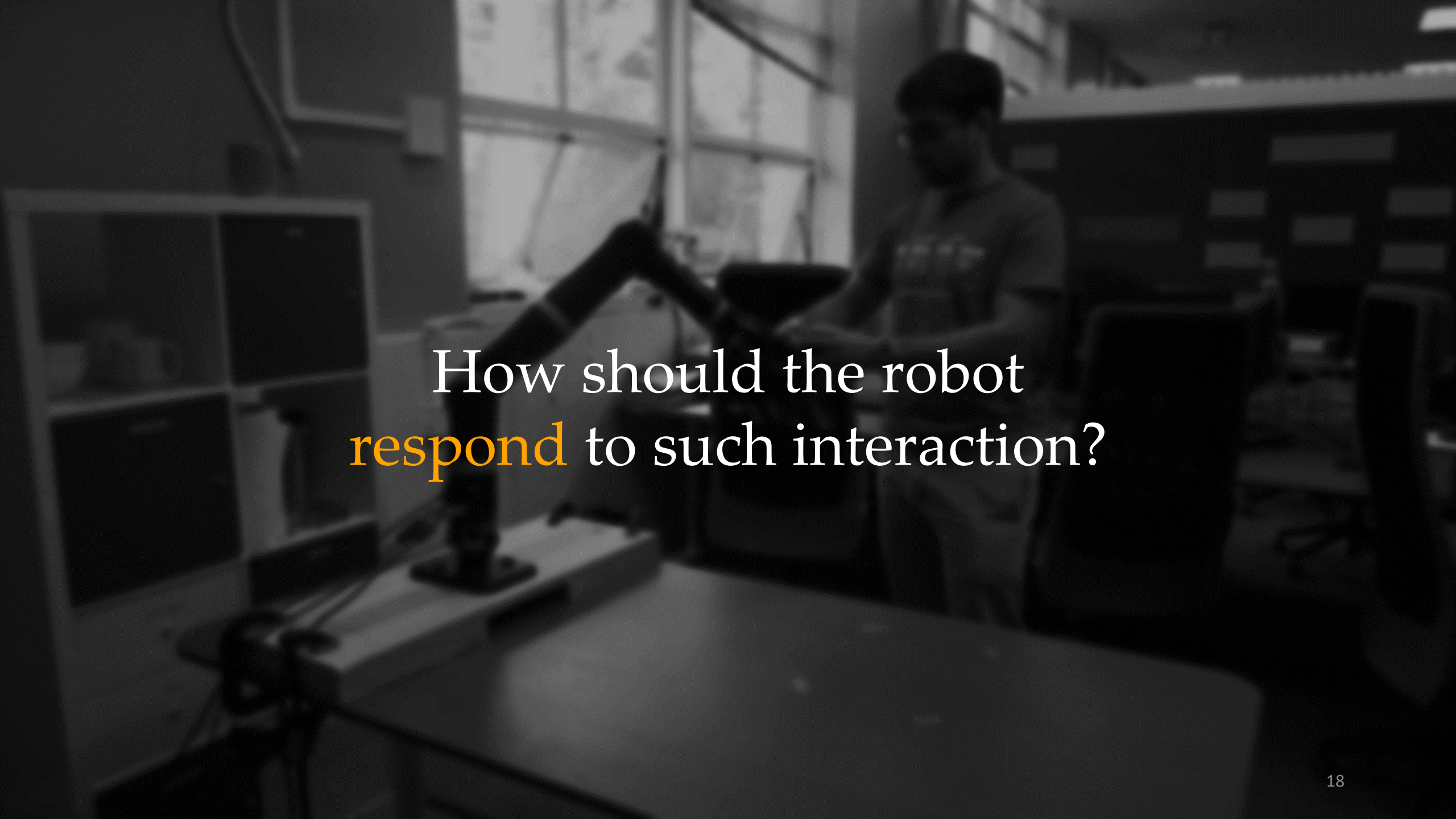
Language

Off-switch

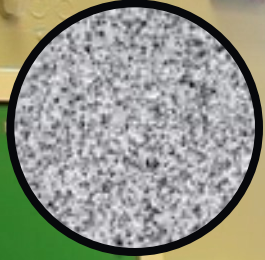
... (and more)





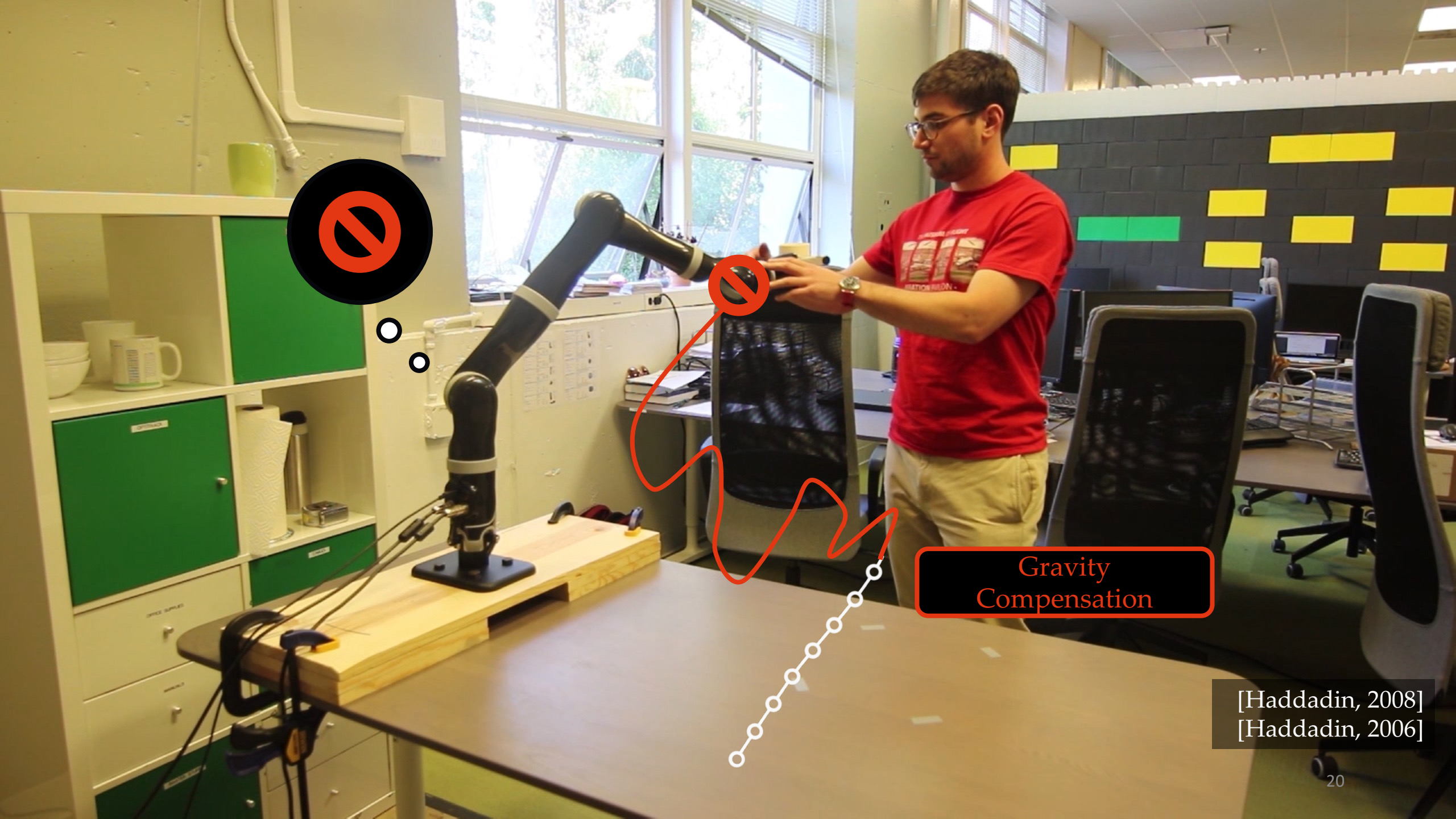
A grayscale photograph of a person in a laboratory or office setting interacting with a robotic arm. The person is standing and reaching towards the arm, which is positioned over a table. The background shows a window and some equipment. The text "How should the robot respond to such interaction?" is overlaid on the image, with the word "respond" in orange.

How should the robot
respond to such interaction?



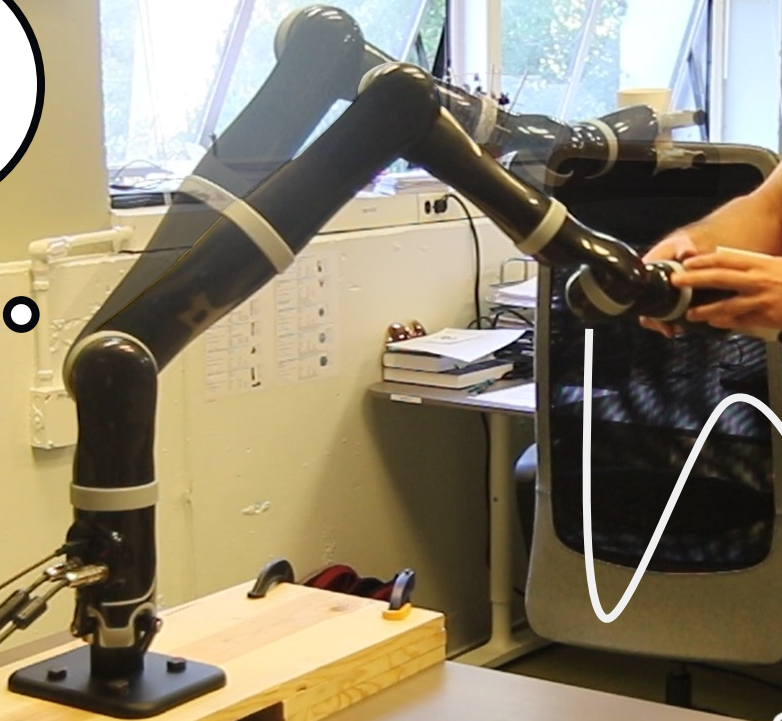
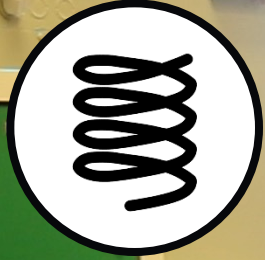
Reject Interaction Force

[Yang, 2011]



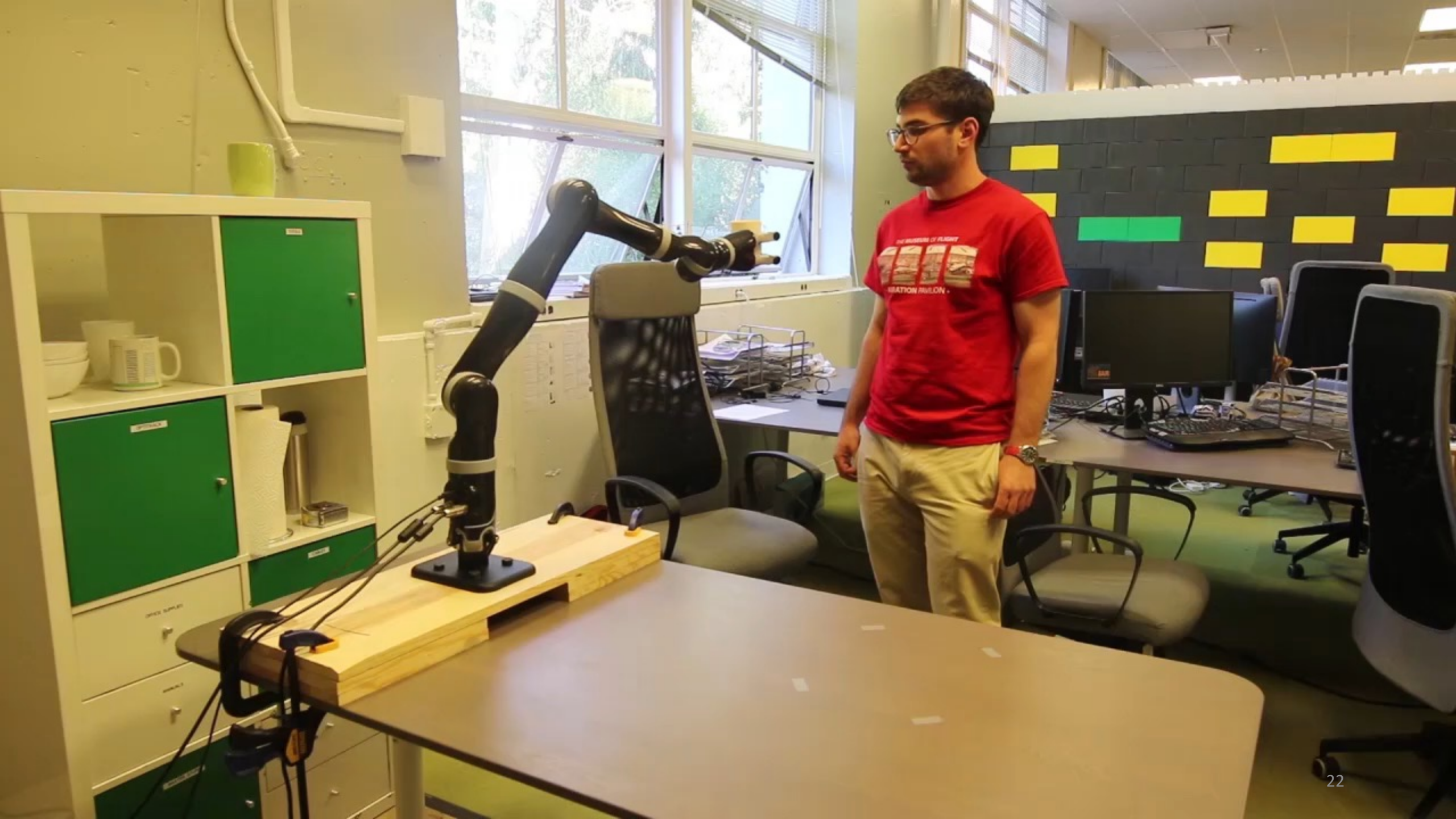
Gravity
Compensation

[Haddadin, 2008]
[Haddadin, 2006]



Impedance Control

[Hogan, 1985]
[Haddadin, 2006]





A grayscale photograph of a laboratory. A person wearing a lab coat is standing and interacting with a robotic arm. The arm is extended towards the person. The background shows a window and some equipment. The text is overlaid on the image.

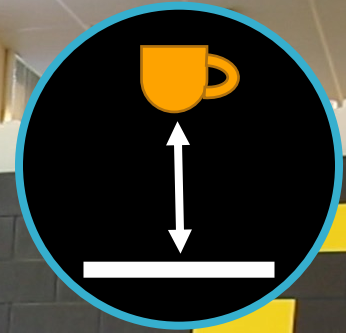
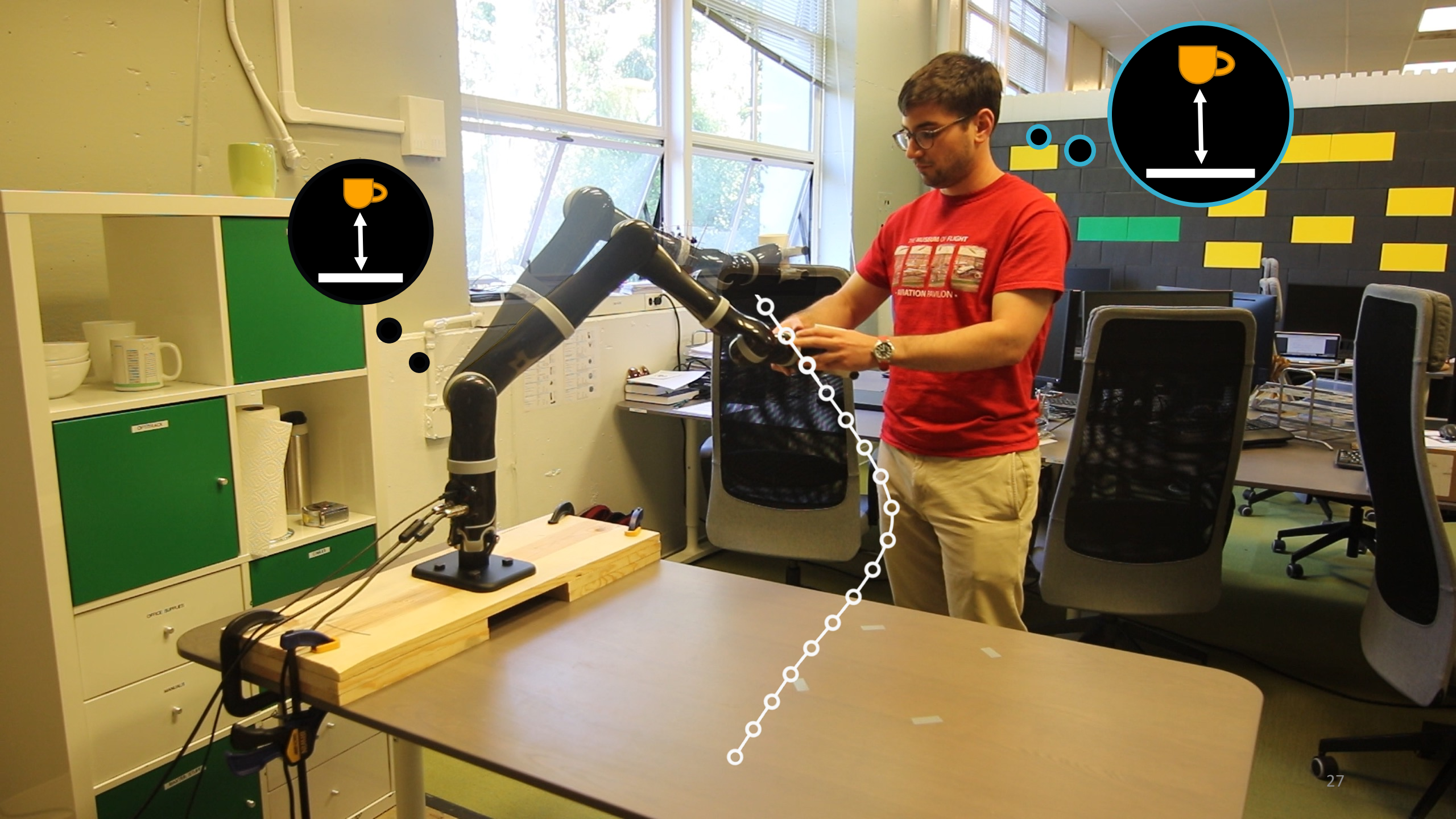
In these strategies,
the robot **resumes** its original behavior!

$$\xi^* = \operatorname{argmax}_{\xi} R(\xi)$$



$$\xi^* = \operatorname{argmax}_{\xi} R(\xi)$$



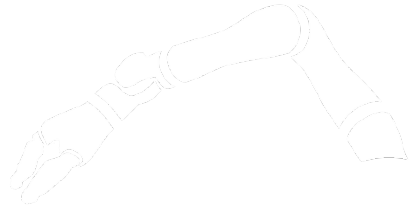


$$\xi^* = \operatorname{argmax}_{\xi} R'(\xi)$$

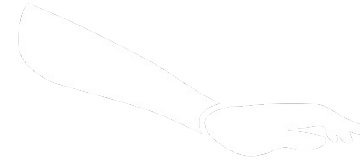


Physical human corrections
provides **observations** about the
correct robot objective function

Formalizing Reacting to pHRI



Robot



Human

State

x

Action

u_R

Observation

u_H

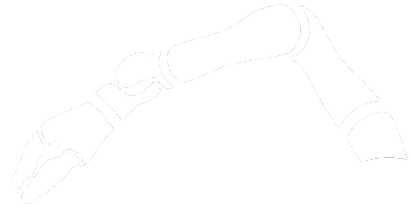
Dynamics

$$x^{t+1} = f(x^t, u_R^t + u_H^t)$$

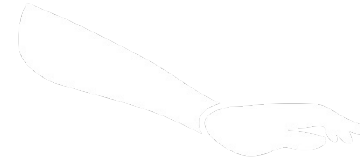
Reward function

$$r(x, u_R, u_H; \theta)$$

Formalizing Reacting to pHRI



Robot



Human

State

x

Action

u_R

Observation

u_H

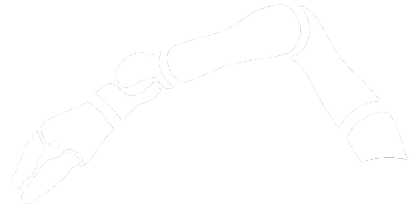
Dynamics

$$x^{t+1} = f(x^t, u_R^t + u_H^t)$$

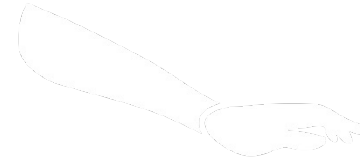
Reward function

$$r(x, u_R, u_H; \theta) = \theta^T \phi(x, u_R, u_H) - \lambda \|u_H\|^2$$

Formalizing Reacting to pHRI



Robot



Human

State

x

Action

u_R

Observation

u_H

Dynamics

$$x^{t+1} = f(x^t, u_R^t + \underbrace{\text{Weight}}_{\text{Feature vector}})$$

Reward function

$$r(x, u_R, u_H; \theta) = \underbrace{\theta^T \phi(x, u_R, u_H)}_{\text{Task reward}} - \underbrace{\lambda \|u_H\|^2}_{\text{Human effort}}$$

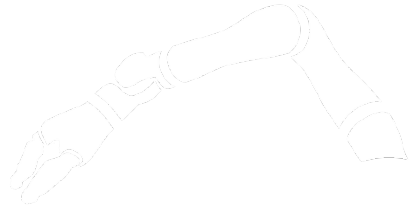
Hidden variable

Task reward

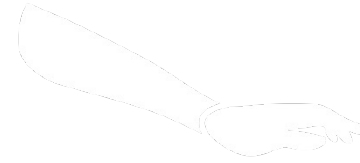
Human effort

Unknown to robot!

Formalizing Reacting to pHRI



Robot



Human

State

x

Action

u_R

Observation

u_H

The human's actions are observations about hidden variable

Dynamics

$$x^{t+1} = f(x^t, u_R^t + u_H^t)$$

Reward function

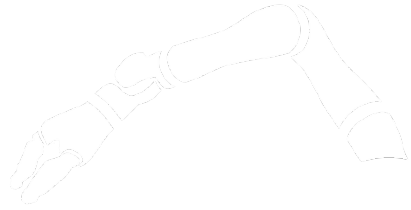
$$r(x, u_R, u_H; \theta) = \theta^T \phi(x, u_R, u_H) - \lambda \|u_H\|^2$$

Observation Model

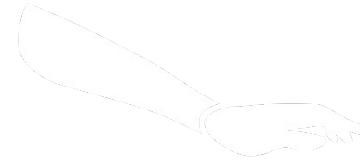
$$P(u_H | x, u_R; \theta) \propto e^{Q(x, u_R + u_H; \theta)} \text{ i.e., planning-based predictive model!}$$

Assume human chooses actions that approximately maximize utility [Baker '07, Ziebart '08]

Formalizing Reacting to pHRI



Robot



Human

State

x

Action

u_R

Observation

u_H

Dynamics

$$x^{t+1} = f(x^t, u_R^t + u_H^t)$$

Reward function

$$r(x, u_R, u_H; \theta) = \theta^T \phi(x, u_R, u_H) - \lambda \|u_H\|^2$$

Observation
Model

$$P(u_H | x, u_R; \theta) \propto e^{Q(x, u_R + u_H; \theta)}$$

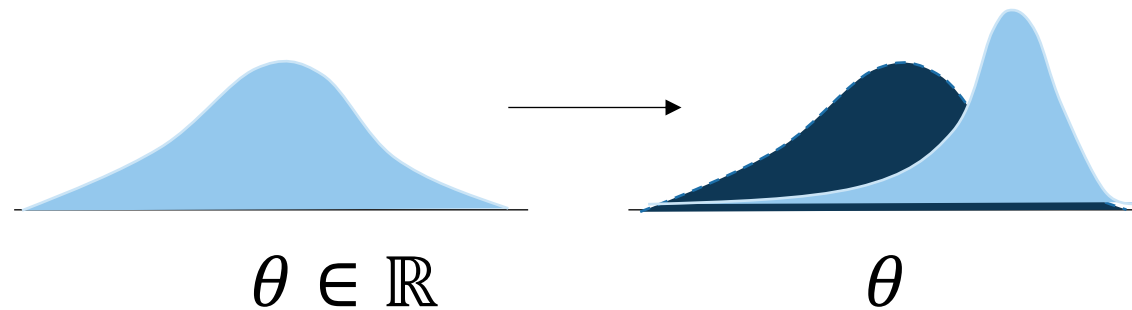
POMDP

Issues:

- (1) Finding u_R^* through POMDP planning is challenging
- (2) Computing Q-values is challenging

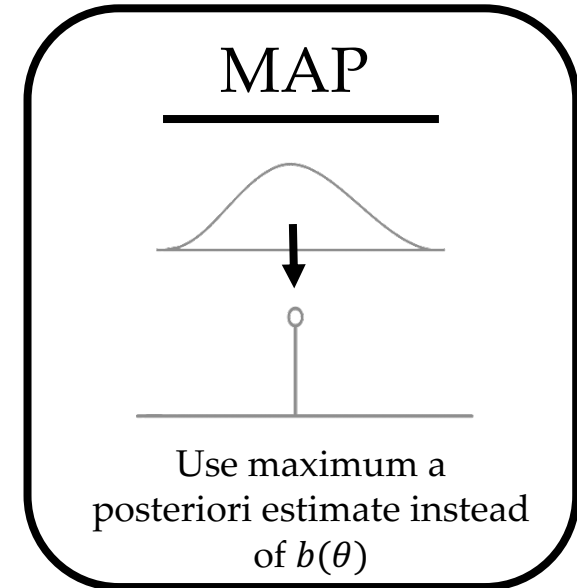
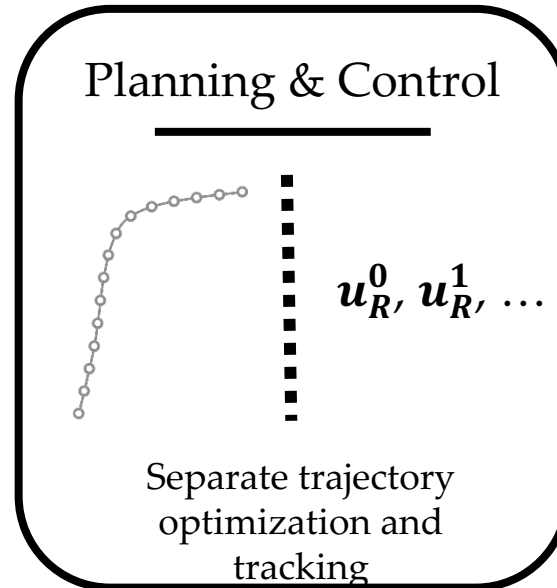
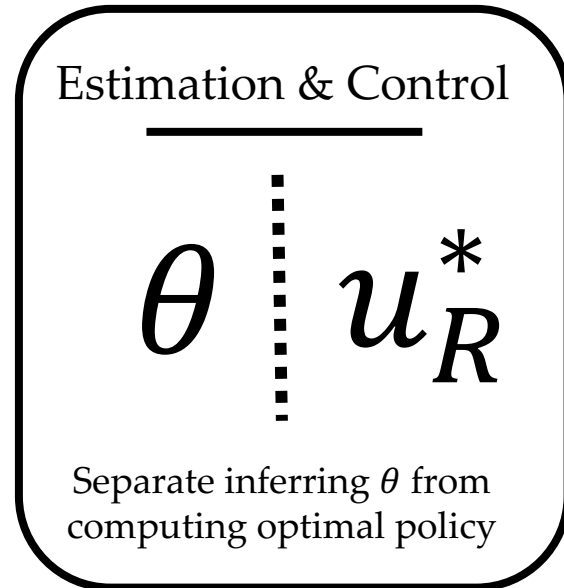
$$P(u_H | u_R, x; \theta) = \frac{e^{Q(x, u_H + u_R; \theta)}}{\int e^{Q(x, \tilde{u}_H + u_R; \theta)} d\tilde{u}_H}$$

- (3) Updating continuous distributions over $b(\theta)$ is challenging

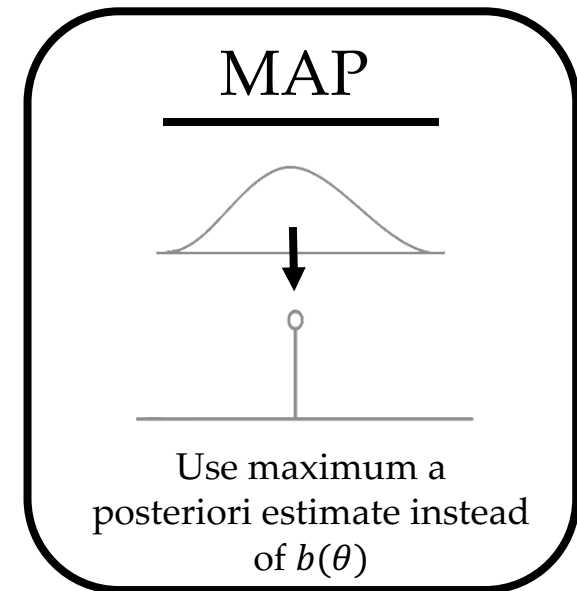
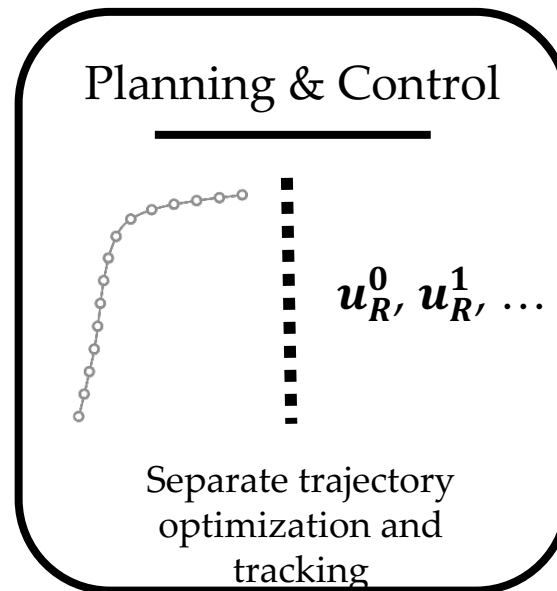
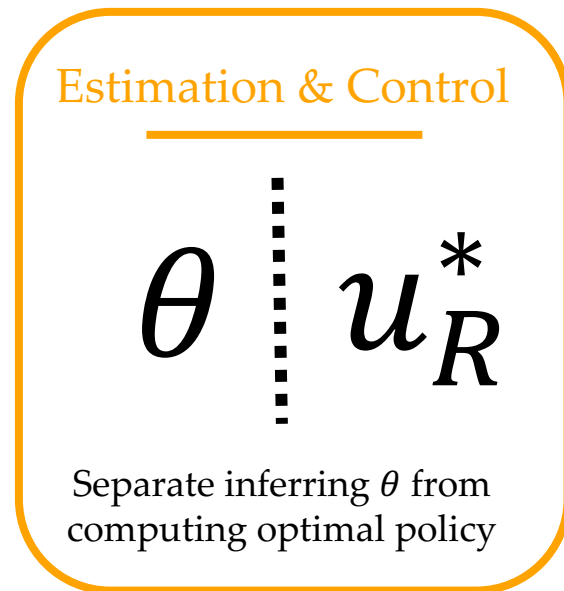


Goal: Make 3 approximations to get online solution!

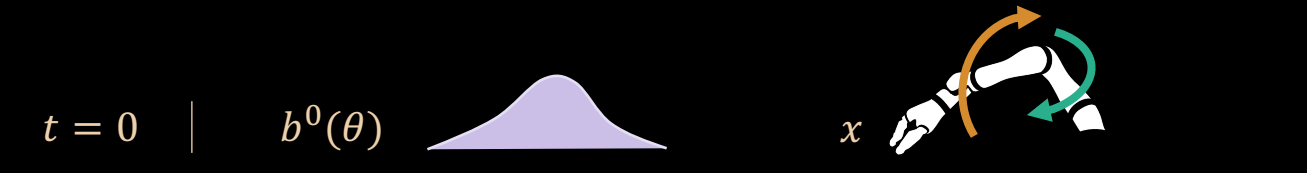
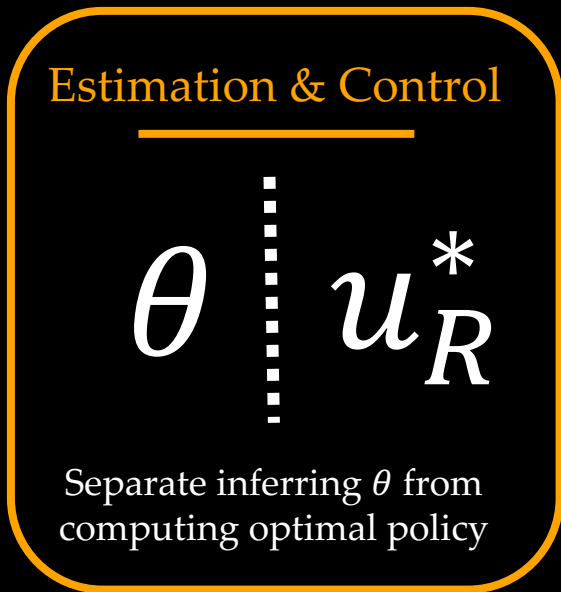
Online Learning of Robot Objectives from pHRI



Online Learning of Robot Objectives from pHRI



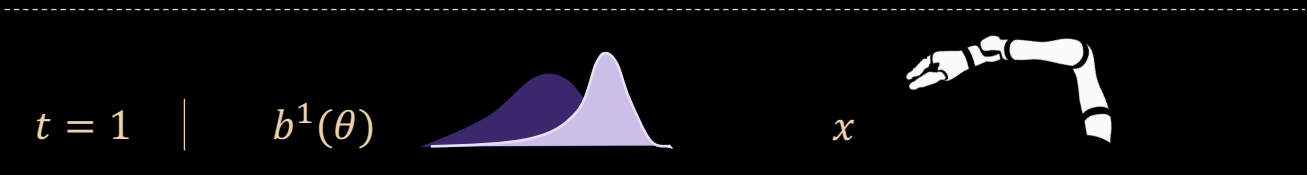
“QMDP”
[Littman et al, 1995]



CONTROL $u_R^0 = \operatorname{argmax}_{u_R} \mathbb{E}_{b^0(\theta)} [Q(x, u_R; \theta)]$

SENSE u_H^0

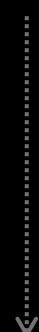
ESTIMATION $b^1(\theta) \propto P(u_H^0 | u_R^0, x; \theta) b^0(\theta)$



CONTROL $u_R^1 = \operatorname{argmax}_{u_R} \mathbb{E}_{b^1(\theta)} [Q(x, u_R; \theta)]$

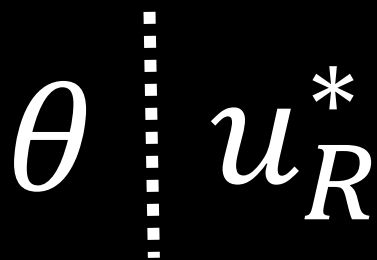
SENSE u_H^1

ESTIMATION $b^2(\theta) \propto P(u_H^1 | u_R^1, x; \theta) b^1(\theta)$



We still have the Q-value issue!

Estimation & Control



Separate inferring θ from computing optimal policy

$t = 0$

$b^0(\theta)$



x



CONTROL

$$u_R^0 = \operatorname{argmax}_{u_R} \mathbb{E}_{b^0(\theta)} [Q(x, u_R; \theta)]$$

SENSE

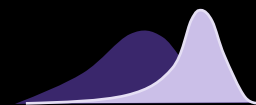
u_H^0

ESTIMATION

$$b^1(\theta) \propto P(u_H^0 | u_R^0, x; \theta) b^0(\theta)$$

$t = 1$

$b^1(\theta)$



x



CONTROL

$$u_R^1 = \operatorname{argmax}_{u_R} \mathbb{E}_{b^1(\theta)} [Q(x, u_R; \theta)]$$

SENSE

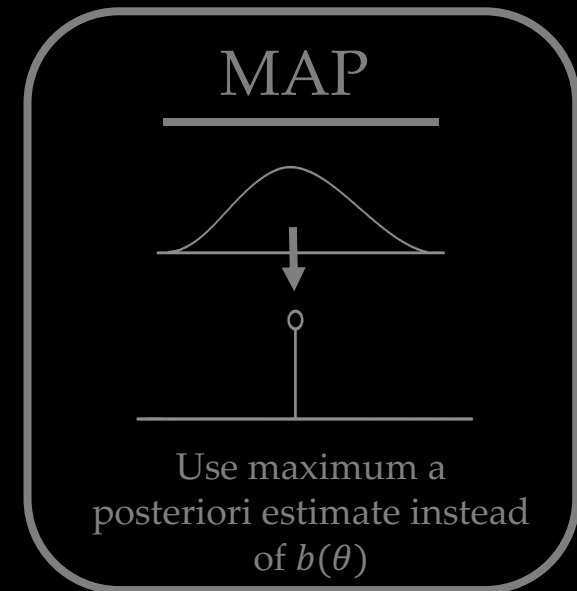
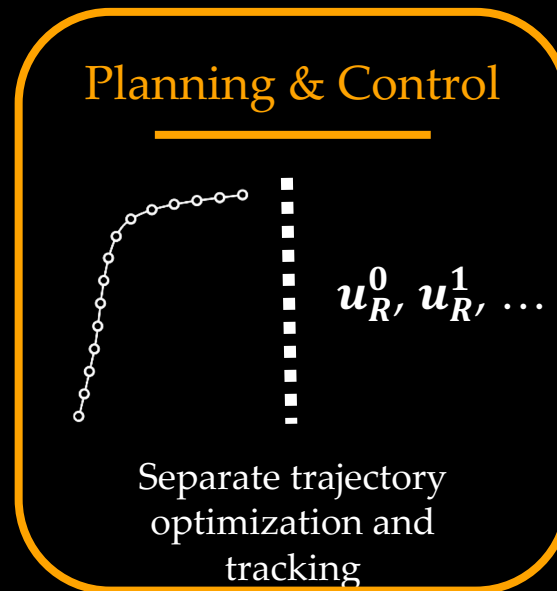
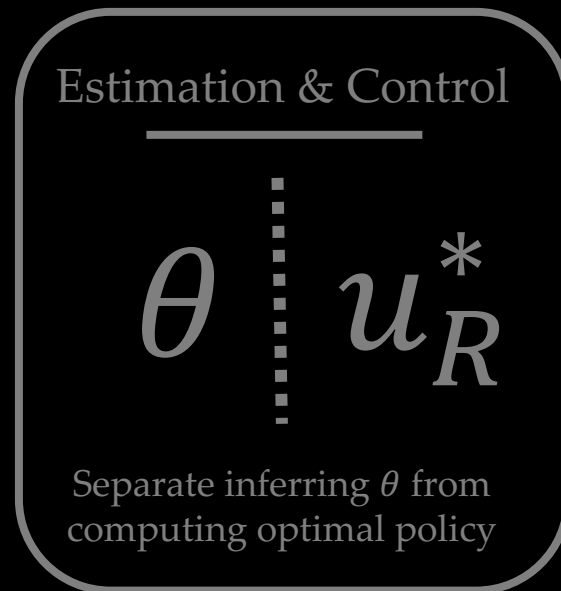
u_H^1

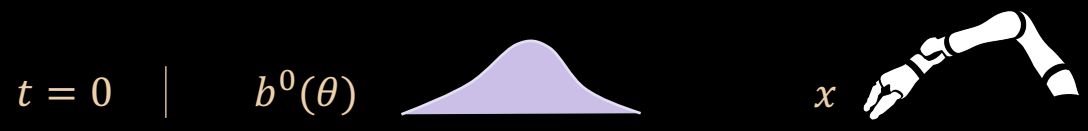
ESTIMATION

$$b^2(\theta) \propto P(u_H^1 | u_R^1, x; \theta) b^1(\theta)$$

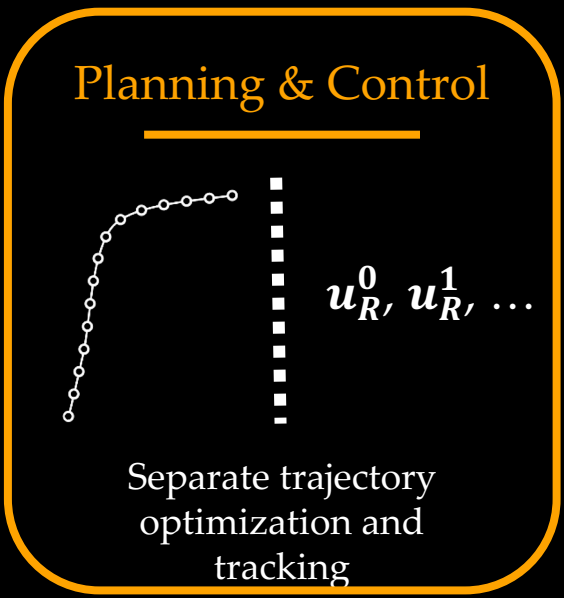


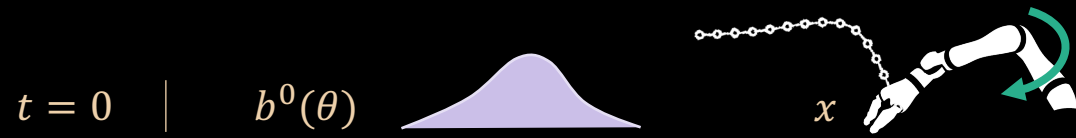
Online Learning of Robot Objectives from pHRI





CONTROL $u_R^0 = \operatorname{argmax}_{u_R} \mathbb{E}_{b^0(\theta)} [Q(x, u_R; \theta)]$



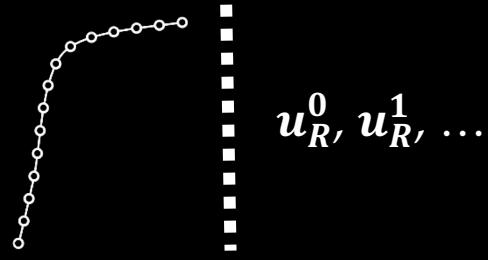


CONTROL $u_R^0 = \operatorname{argmax}_{u_R} \mathbb{E}_{b^0(\theta)} [Q(x, u_R; \theta)]$

PLAN $\xi_R^0 = \operatorname{arg max}_{\xi} \theta^T \Phi(\xi)$

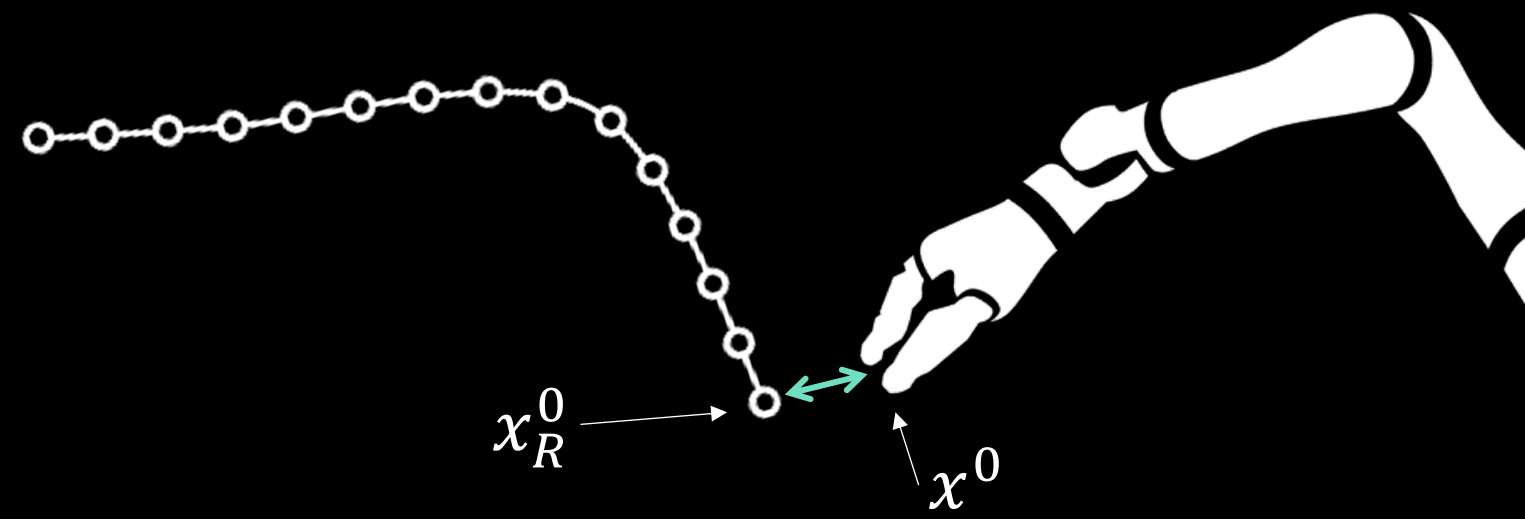
CONTROL $u_R^0 = B_R (\dot{x}_R^0 - \dot{x}^0) + K_R (x_R^0 - x^0)$

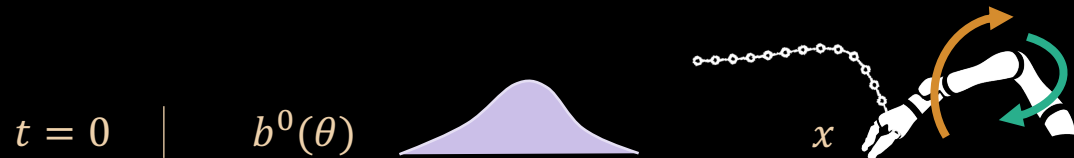
Planning & Control



u_R^0, u_R^1, \dots

Separate trajectory optimization and tracking





PLAN $\xi_R^0 = \arg \max_{\xi} \theta^T \Phi(\xi)$

CONTROL $u_R^0 = B_R(\dot{x}_R^0 - \dot{x}^0) + K_R(x_R^0 - x^0)$

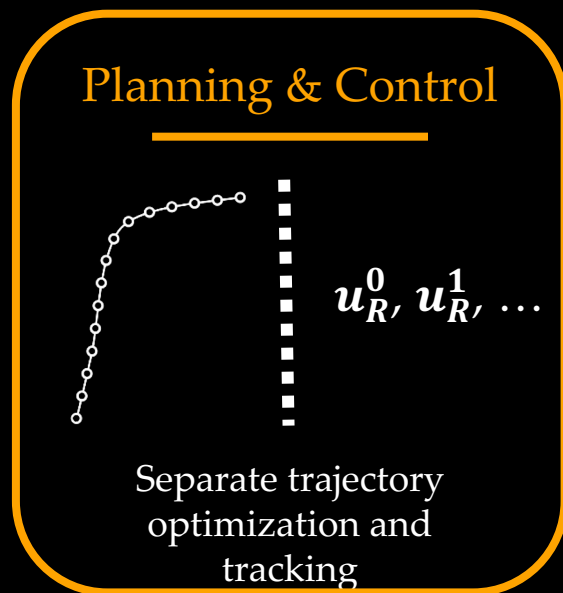
SENSE u_H^0

ESTIMATION $b^1(\theta) \propto P(u_H^0 | u_R^0, x; \theta) b^0(\theta)$

$P(u_H^0 | u_R^0, x; \theta) \propto e^{Q(x, u_R^0 + u_H^0; \theta)}$

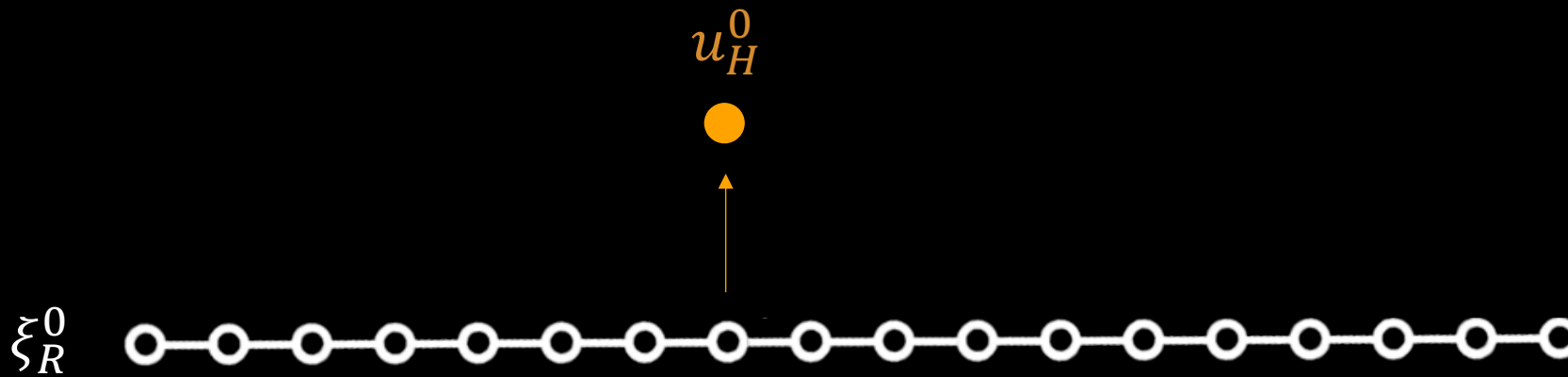
$P(\xi_H^0 | \xi_R^0; \theta) \propto e^{R(\xi_H^0, \xi_R^0; \theta)}$

Q: What is ξ_H^0 ?



Q: What is ξ_H^0 ?

$$P(\xi_H^0 | \xi_R^0; \theta) \propto e^{R(\xi_H^0, \xi_R^0; \theta)}$$



Q: What is ξ_H^0 ?

$$P(\xi_H^0 | \xi_R^0; \theta) \propto e^{R(\xi_H^0, \xi_R^0; \theta)}$$

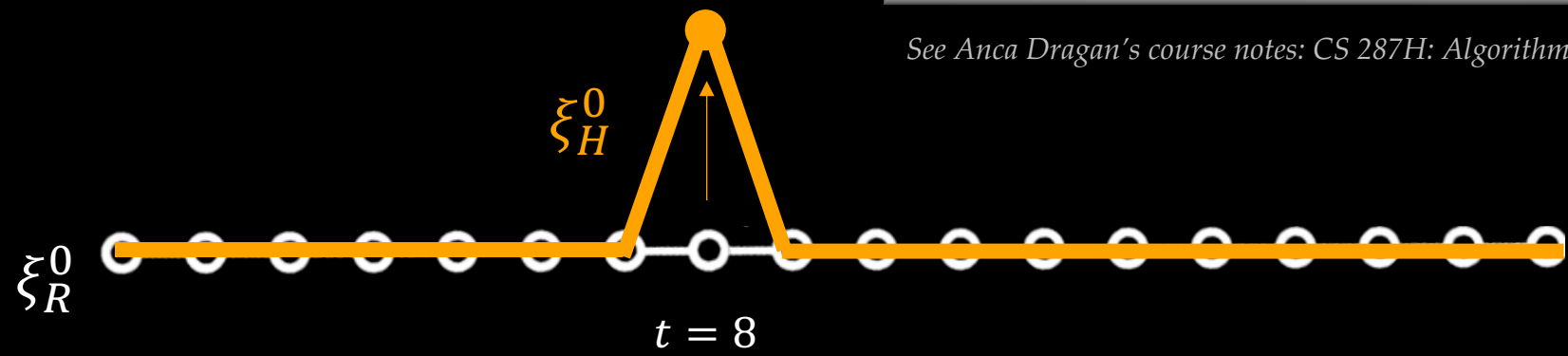
Different Inner Products

$$\langle g_1, g_2 \rangle = g_1^T g_2$$

$\|a-b\|^2 = (a-b)^T (a-b) = 100$
 $\|a-c\|^2 = (a-c)^T (a-c) = 150$

Idea: $\langle g_1, g_2 \rangle = g_1^T A g_2$ make c closer to a than b is

See Anca Dragan's course notes: CS 287H: Algorithmic HRI



$$\xi_H^0 = \arg \min_{\xi} (\xi_R^0 - \xi)^T I (\xi_R^0 - \xi)$$

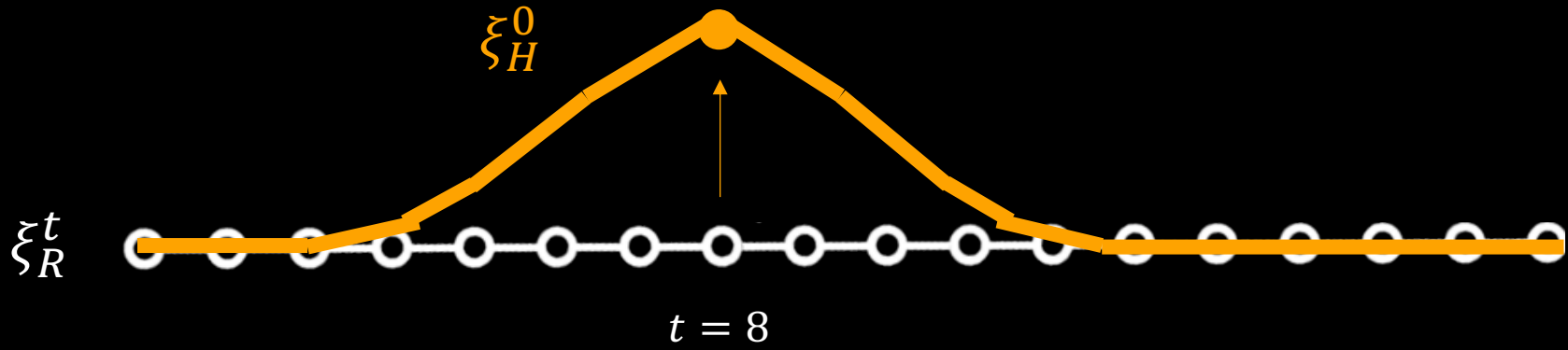
s.t. $\xi(8) = \xi_R^0(8) + u_H^0$

$$\xi_H^0 = \xi_R^0 + I \begin{bmatrix} 0 \\ \vdots \\ u_H^0 \\ \vdots \\ 0 \end{bmatrix}$$

Q: What is ξ_H^0 ?

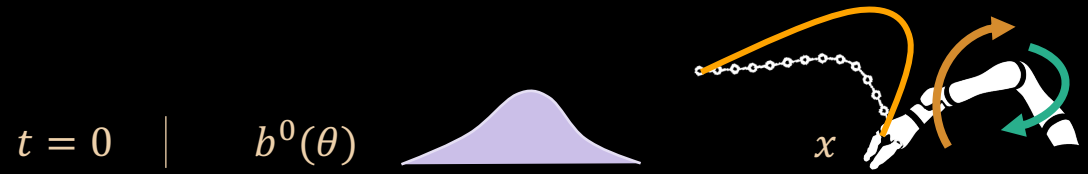
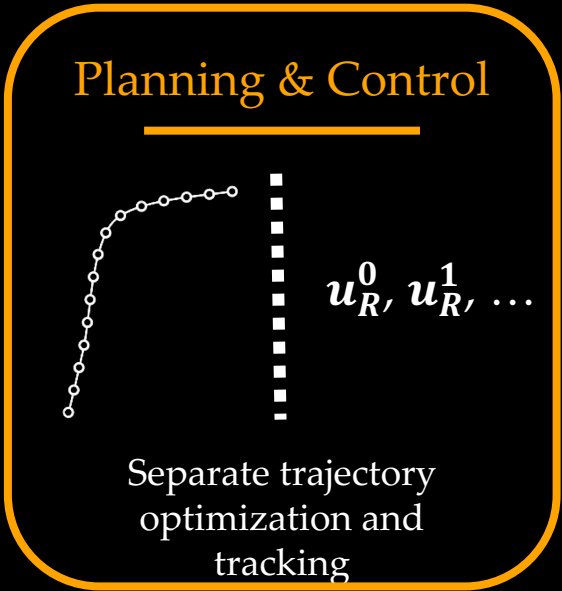
$$P(\xi_H^0 | \xi_R^0; \theta) \propto e^{R(\xi_H^0, \xi_R^0; \theta)} \longrightarrow P(\xi_H^0 | \xi_R^0; \theta) \propto e^{\theta^T \Phi(\xi_H^0) - \lambda \|\xi_H^0 - \xi_R^0\|^2}$$

Simplified observation model



$$\xi_H^0 = \xi_R^0 + A^{-1} \begin{bmatrix} 0 \\ \vdots \\ u_H^0 \\ \vdots \\ 0 \end{bmatrix}$$

[Dragan, 2015]
[Losey, 2017]



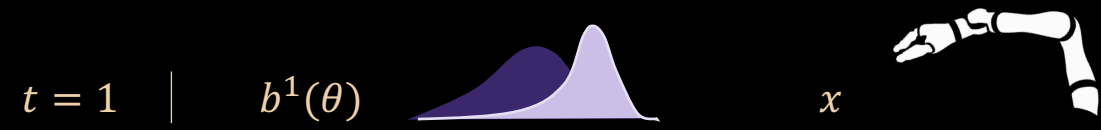
PLAN $\xi_R^0 = \arg \max_{\xi} \theta^T \Phi(\xi)$

CONTROL $u_R^0 = B_R(\dot{x}_R^0 - \dot{x}^0) + K_R(x_R^0 - x^0)$

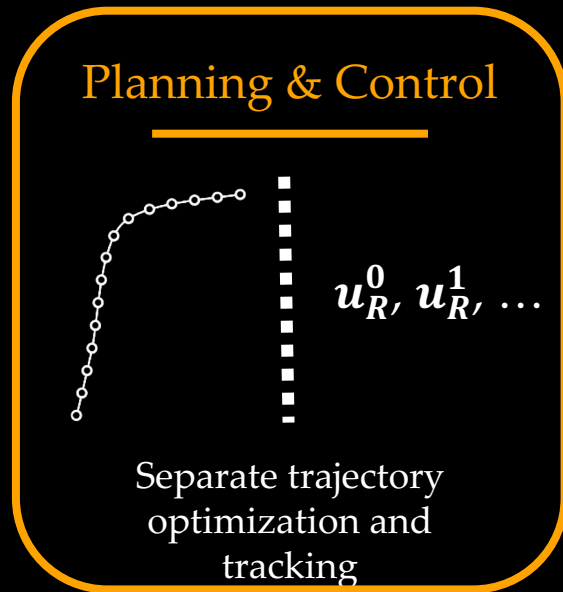
SENSE u_H^0

DEFORM $\xi_H^0 = \xi_R^0 + A^{-1} \begin{bmatrix} 0 \\ \vdots \\ u_H^0 \\ \vdots \\ 0 \end{bmatrix}$

ESTIMATION $b^1(\theta) \propto P(\xi_H^0 | \xi_R^0; \theta) b^0(\theta)$



We still have to update continuous distribution over $\theta \in \mathbb{R}^n$



$t = 0$

$b^0(\theta)$



PLAN

$$\xi_R^0 = \arg \max_{\xi} \theta^T \Phi(\xi)$$

CONTROL

$$u_R^0 = B_R(\dot{x}_R^0 - \dot{x}^0) + K_R(x_R^0 - x^0)$$

SENSE

u_H^0

DEFORM

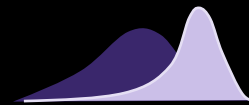
$$\xi_H^0 = \xi_R^0 + A^{-1} \begin{bmatrix} 0 \\ \vdots \\ u_H^0 \\ \vdots \\ 0 \end{bmatrix}$$

ESTIMATION

$$b^1(\theta) \propto P(\xi_H^0 | \xi_R^0; \theta) b^0(\theta)$$

$t = 1$

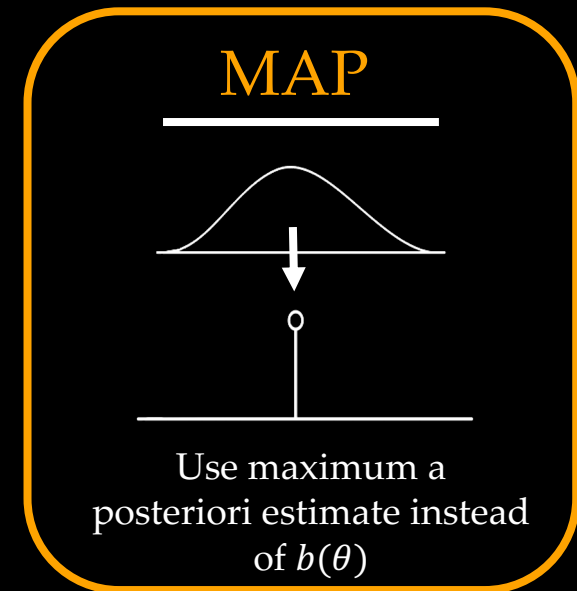
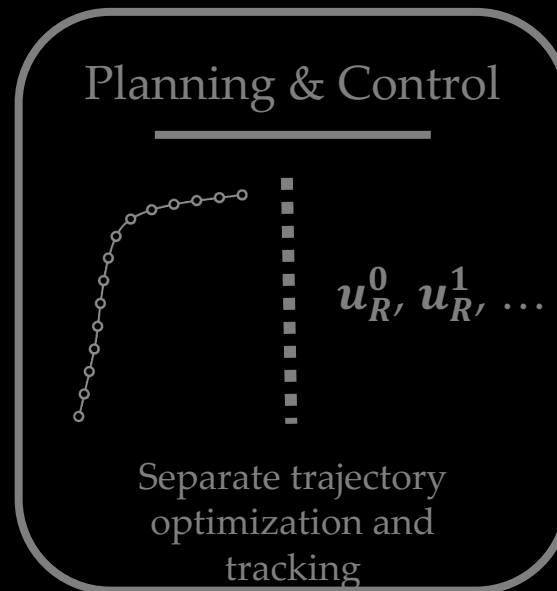
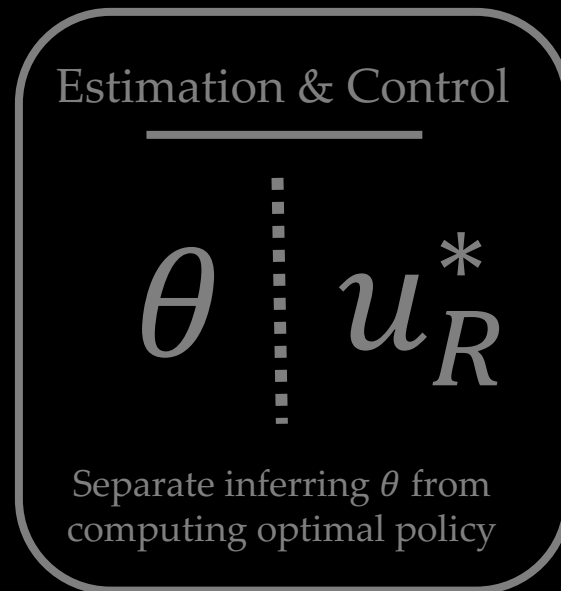
$b^1(\theta)$

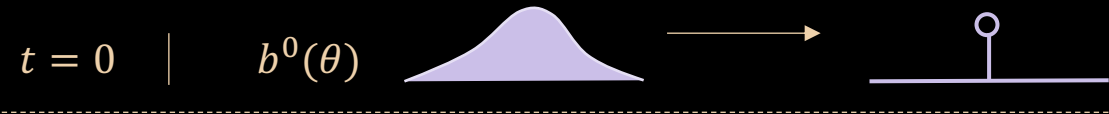


x

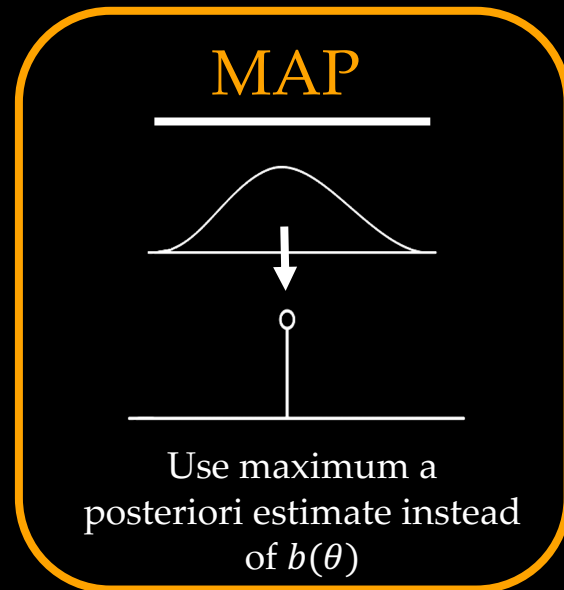


Online Learning of Robot Objectives from pHRI





$$b^1(\theta) \propto P(\xi_H | \xi_R; \theta) b^0(\theta)$$

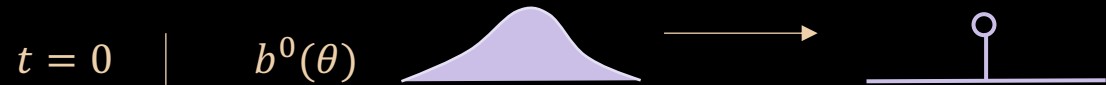


$$\hat{\theta}^1 = \arg \max_{\theta} b^1(\theta)$$

$$= \arg \max_{\theta} P(\xi_H | \xi_R; \theta) P(\theta)$$

$$= \arg \max_{\theta} \frac{e^{\theta^T \Phi(\xi_H) - \lambda \|\xi_H - \xi_R^0\|^2}}{\int e^{\theta^T \Phi(\xi_H) - \lambda \|\xi_H - \xi_R^0\|^2} d\xi_H} P(\theta)$$

One last approximation (I promise) 😊

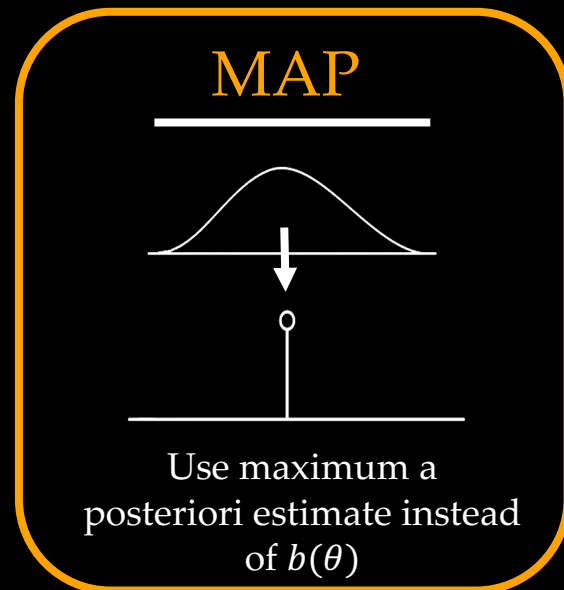


$$b^1(\theta) \propto P(\xi_H | \xi_R; \theta) b^0(\theta)$$

$$\hat{\theta}^1 = \arg \max_{\theta} b^1(\theta)$$

$$= \arg \max_{\theta} P(\xi_H | \xi_R; \theta) P(\theta)$$

$$= \arg \max_{\theta} \frac{e^{\theta^T \Phi(\xi_H) - \lambda \|\xi_H - \xi_R^0\|^2}}{\int e^{\theta^T \Phi(\xi_H) - \lambda \|\xi_H - \xi_R^0\|^2} d\xi_H} P(\theta)$$

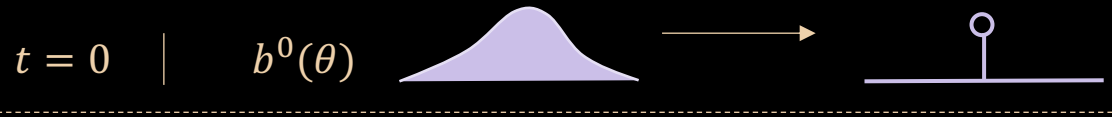


Laplace's Method

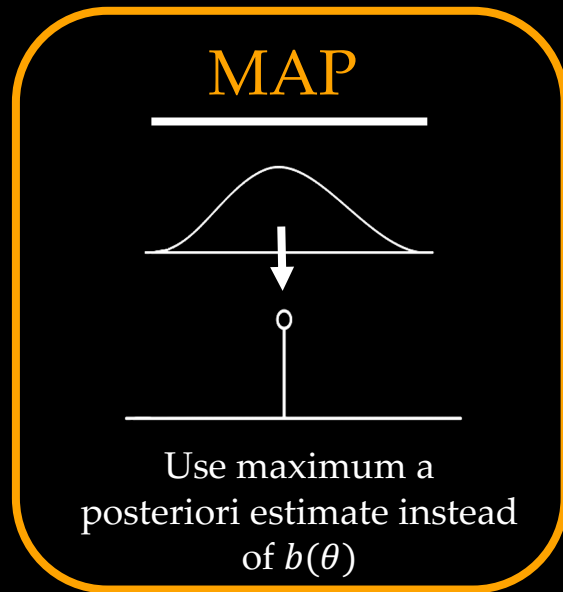
$$\int e^{f(x)} dx \quad f(x) \text{ is twice differentiable}$$

- (1) 2nd Order Taylor Series Expansion around optimum
- (2) Get Gaussian Integral and closed form solution!

<http://www.inference.org.uk/mackay/itprnm/ps/341.342.pdf>



$$b^1(\theta) \propto P(\xi_H | \xi_R; \theta) b^0(\theta)$$

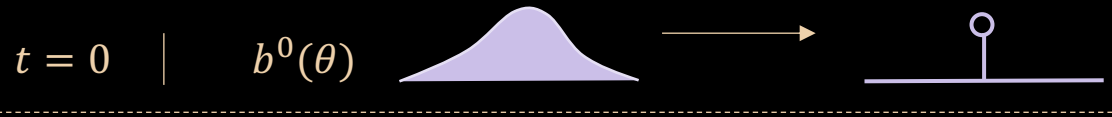


$$\hat{\theta}^1 = \arg \max_{\theta} b^1(\theta)$$

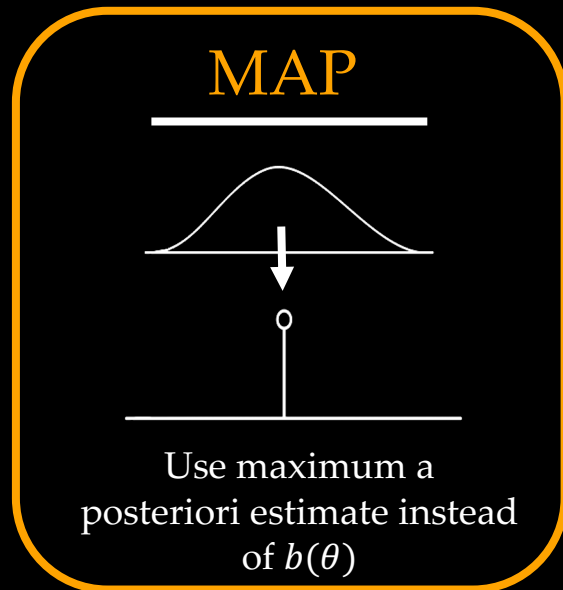
$$= \arg \max_{\theta} P(\xi_H | \xi_R; \theta) P(\theta)$$

$$\approx \arg \max_{\theta} \frac{e^{\theta^T \Phi(\xi_H) - \lambda \|\xi_H - \xi_R^0\|^2}}{e^{\theta^T \Phi(\xi_R)} \frac{\sqrt{(2\pi)^n}}{\sqrt{|H(\xi_R)|}}} P(\theta)$$

$$= \arg \max_{\theta} e^{\theta^T (\Phi(\xi_H) - \Phi(\xi_R)) - \lambda \|\xi_H - \xi_R^0\|^2} \frac{\sqrt{|H(\xi_R)|}}{\sqrt{(2\pi)^n}} P(\theta)$$



$$b^1(\theta) \propto P(\xi_H | \xi_R; \theta) b^0(\theta)$$



log + simplify \curvearrowright

$$= \arg \max_{\theta} e^{\theta^T (\Phi(\xi_H) - \Phi(\xi_R)) - \lambda \|\xi_H - \xi_R^0\|^2} \frac{\sqrt{|H(\xi_R)|}}{\sqrt{(2\pi)^n}} P(\theta)$$

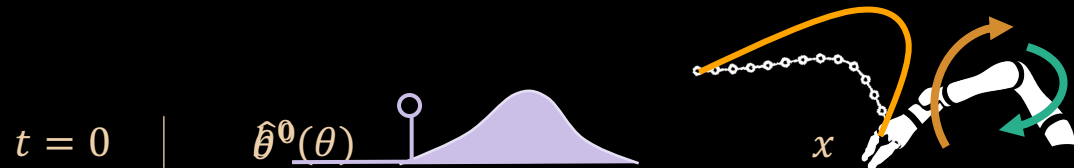
$$= \arg \max_{\theta} \theta^T (\Phi(\xi_H) - \Phi(\xi_R)) - \lambda \|\xi_H - \xi_R^0\|^2 + \log(P(\theta))$$

$$P(\theta) = e^{-\frac{1}{2\alpha} \|\theta - \hat{\theta}^0\|^2}$$

Take gradient and $\nabla_{\theta} = 0$

$$= \arg \max_{\theta} \theta^T (\Phi(\xi_H) - \Phi(\xi_R)) - \frac{1}{2\alpha} \|\theta - \hat{\theta}\|^2$$

$$\hat{\theta}^1 = \hat{\theta}^0 + \alpha \left(\theta^T (\Phi(\xi_H) - \Phi(\xi_R)) \right)$$



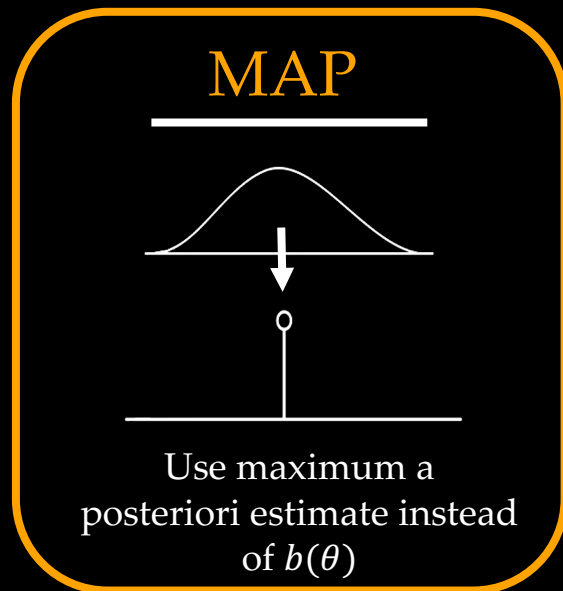
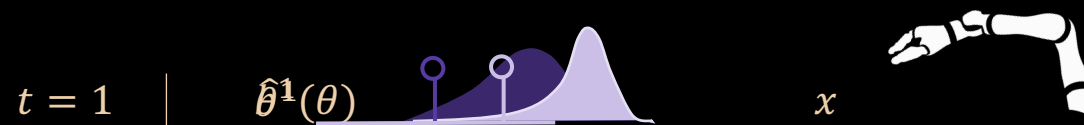
PLAN $\xi_R^0 = \arg \max_{\xi} \theta^T \Phi(\xi)$

CONTROL $u_R^0 = B_R(\dot{x}_R^0 - \dot{x}^0) + K_R(x_R^0 - x^0)$

SENSE u_H^0

DEFORM $\xi_H^0 = \xi_R^0 + A^{-1} \begin{bmatrix} 0 \\ \vdots \\ u_H^0 \\ \vdots \\ 0 \end{bmatrix}$

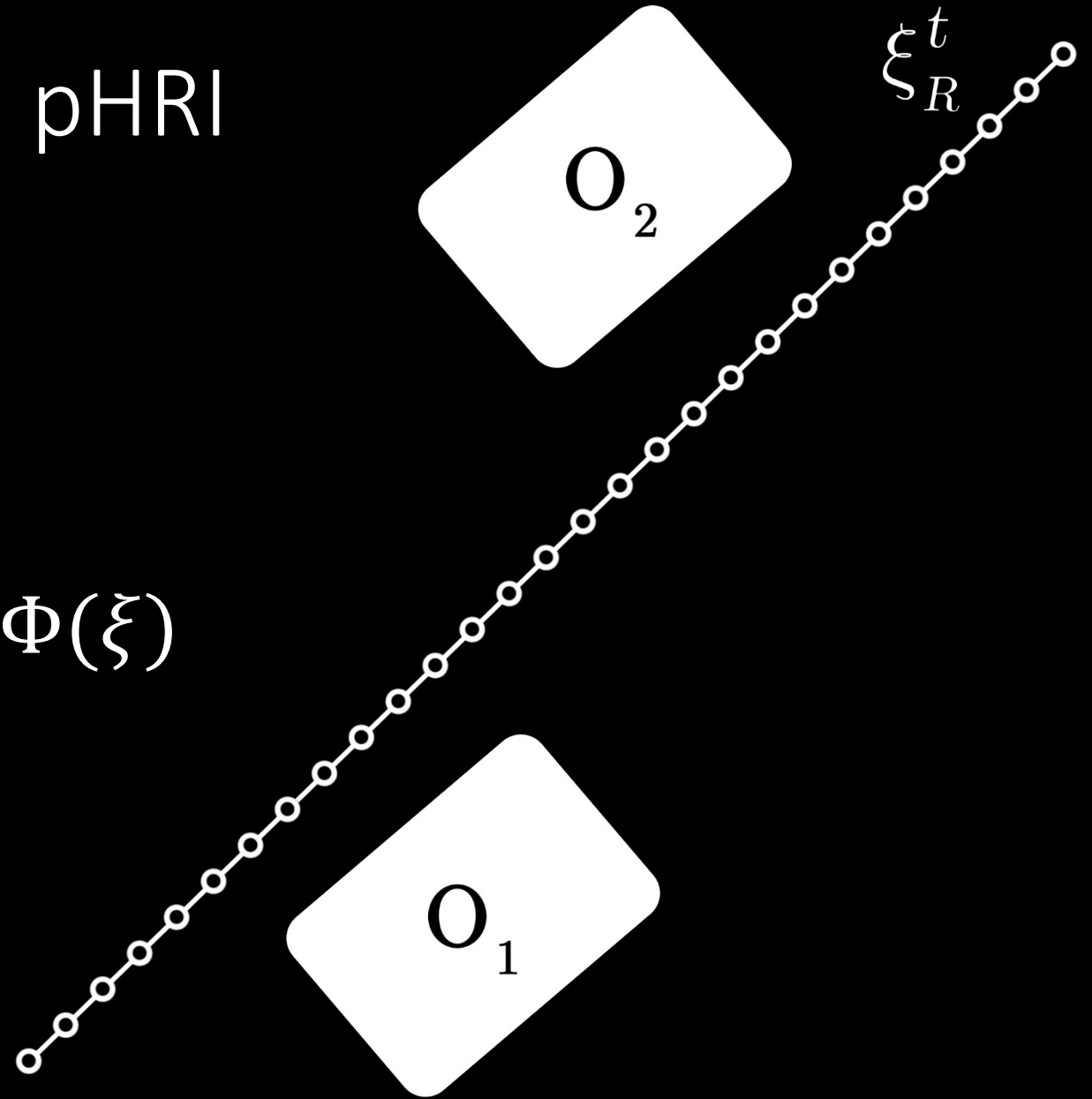
ESTIMATION $\hat{\theta}^1 = \hat{\theta}^0 + P \alpha (\xi_H^0 \Phi(\xi_H^0) - \theta^T \Phi(\xi_H^0) + \xi_R^0 \Phi(\xi_R^0) - \theta^T \Phi(\xi_R^0))$



Online Learning from pHRI

Plan robot trajectory
from start to goal

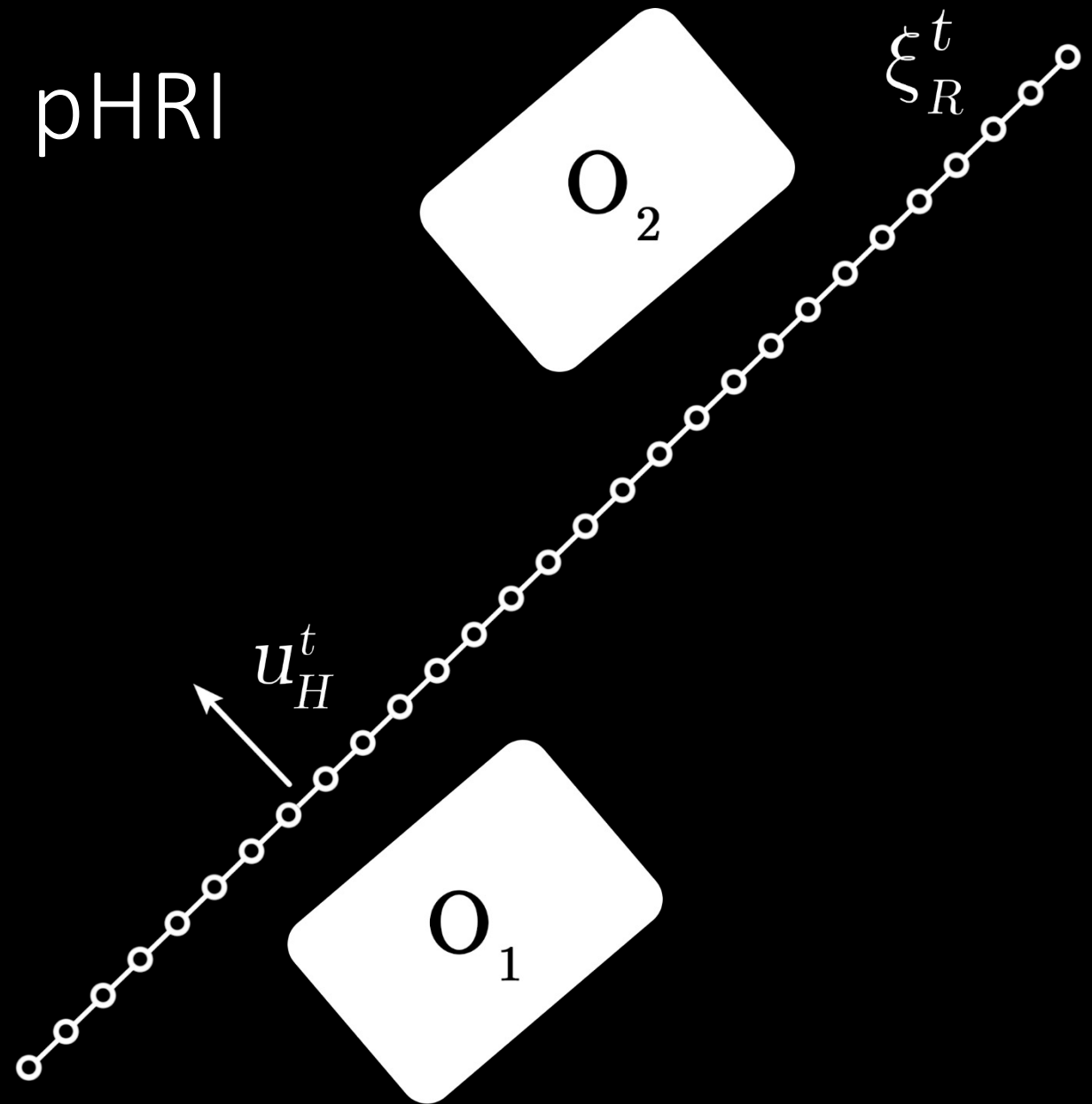
$$\xi_R^t \leftarrow \arg \max_{\xi} \hat{\theta}^t \Phi(\xi)$$



Online Learning from pHRI

Sense human's
applied force

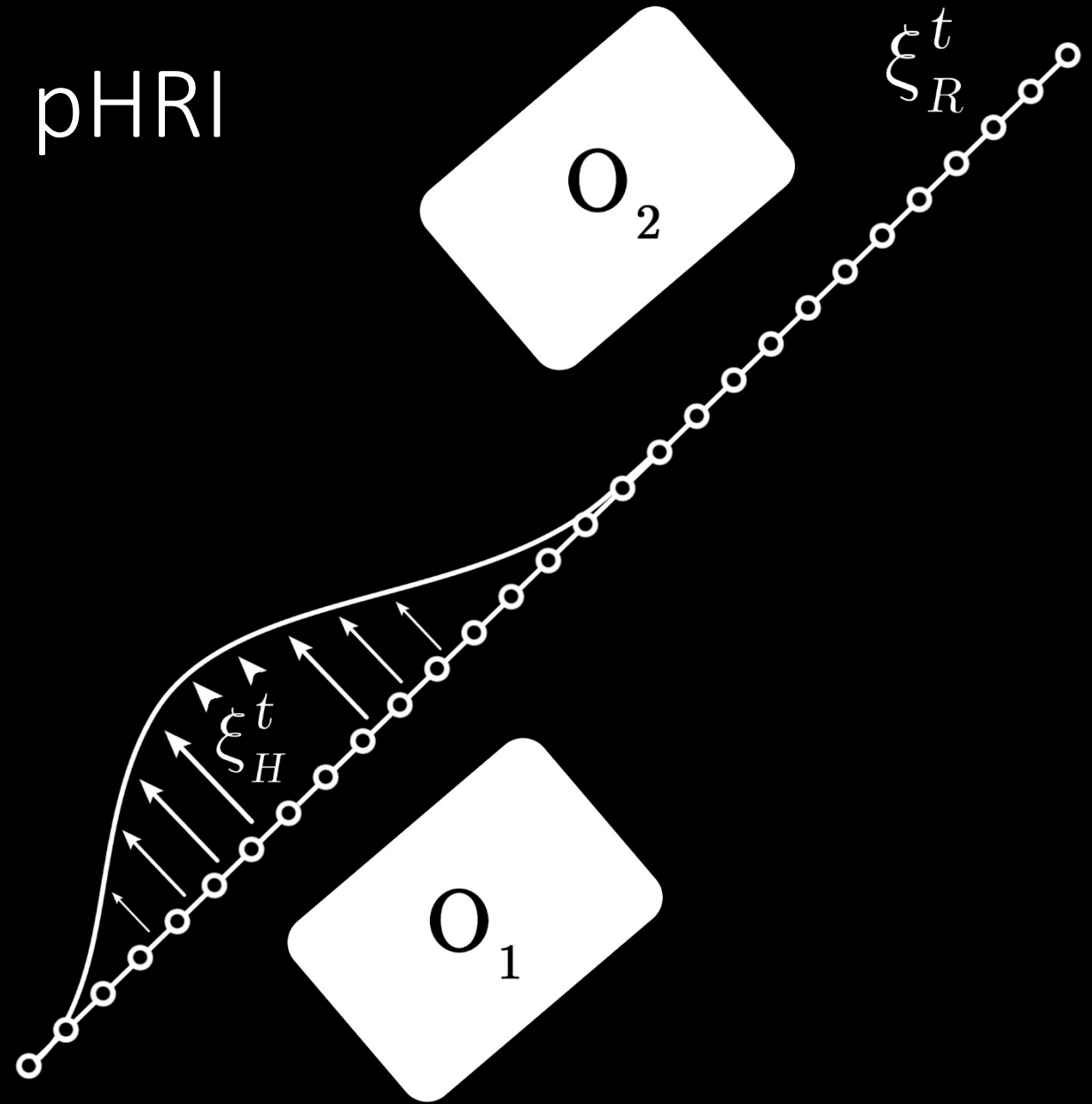
$$u_H^t$$



Online Learning from pHRI

Deform to get human's preferred trajectory

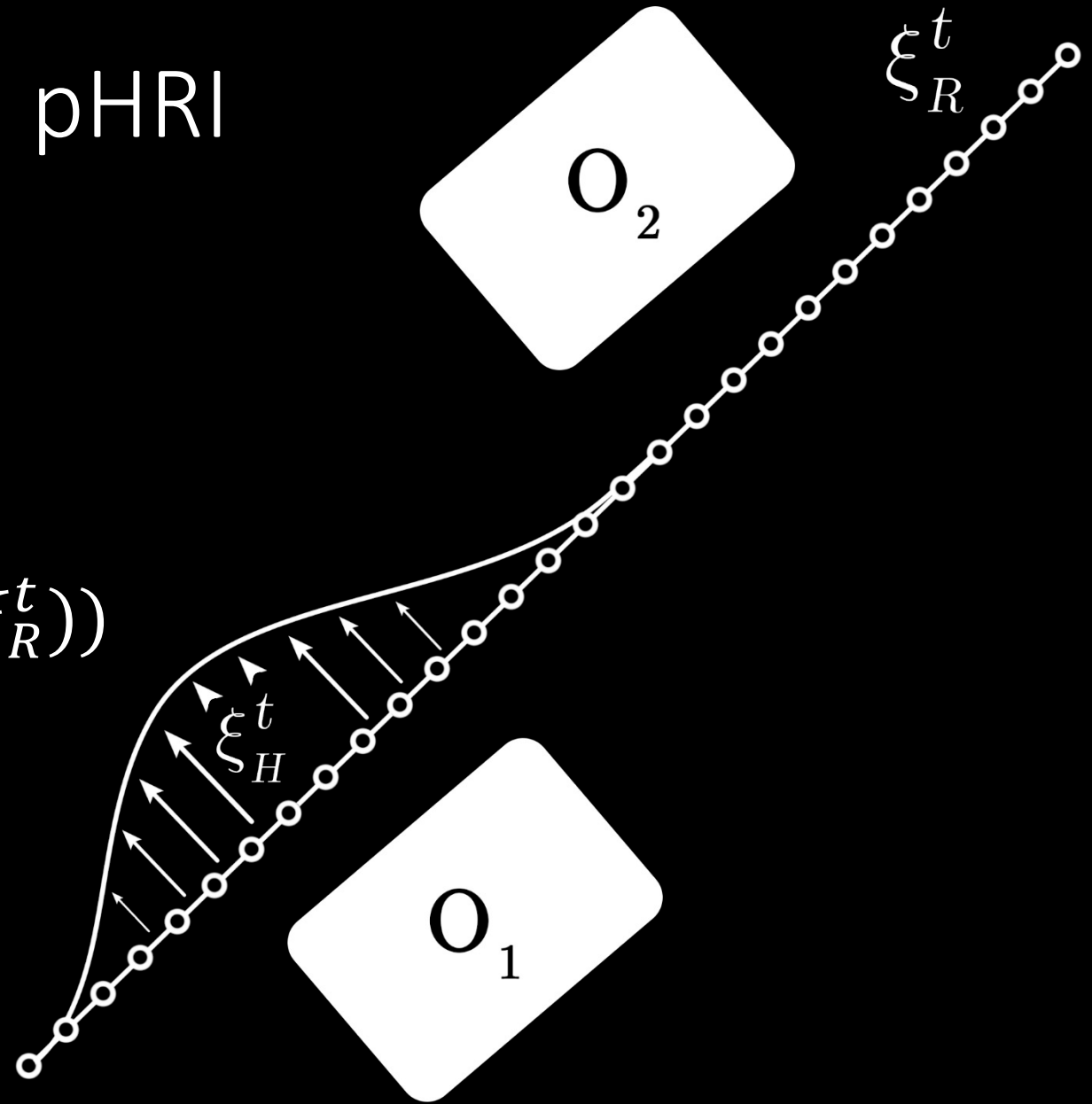
$$\begin{matrix} \zeta_H^t \\ \xi_H^t \end{matrix} = \begin{matrix} \zeta_R^t \\ \xi_R^t \end{matrix} + A^{-1} \begin{bmatrix} 0 \\ \vdots \\ u_H^t \\ \vdots \\ 0 \end{bmatrix}$$



Online Learning from pHRI

Update robot objective

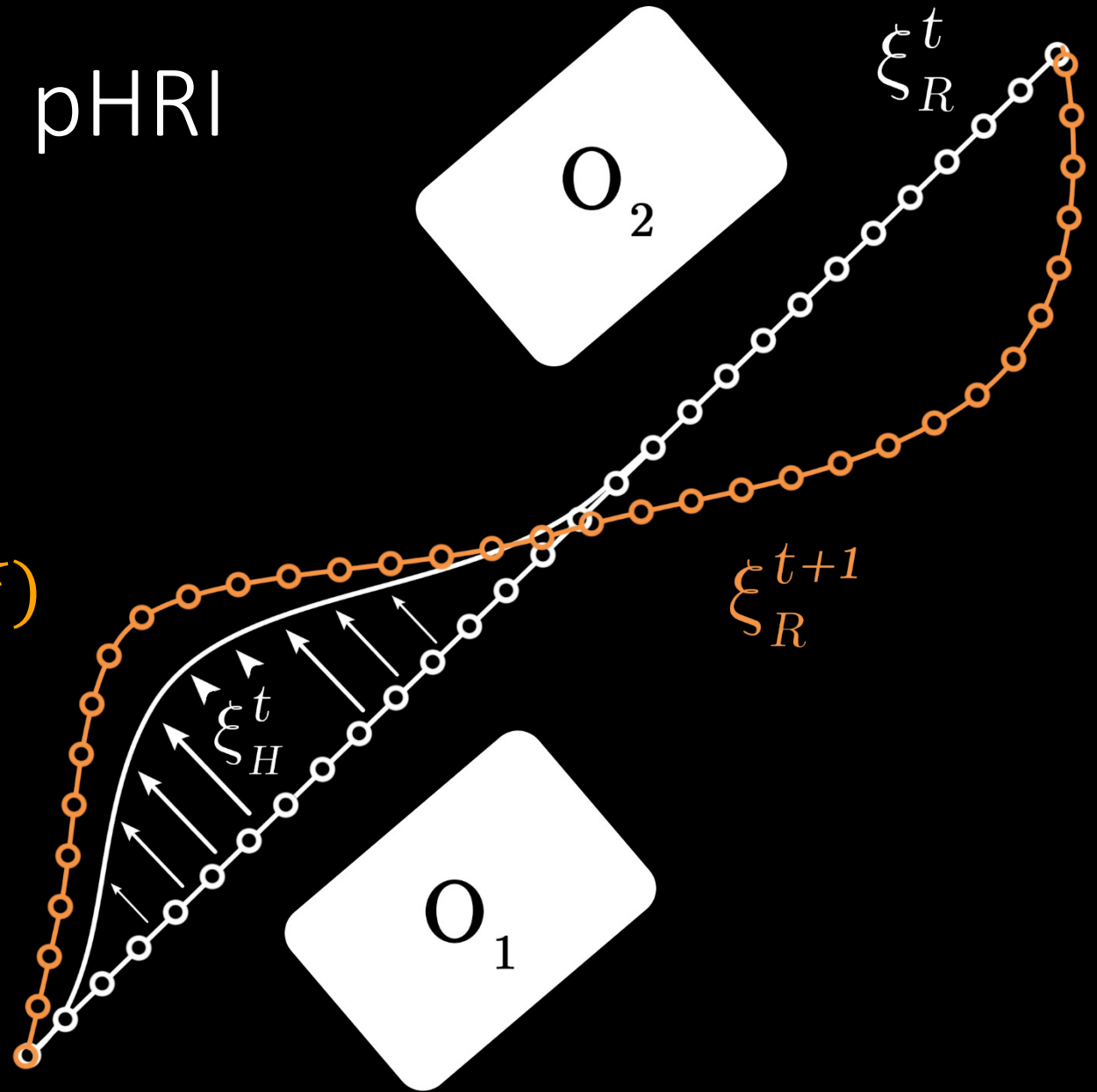
$$\hat{\theta}^{t+1} \leftarrow \hat{\theta}^t + \alpha(\Phi(\xi_H^t) - \Phi(\xi_R^t))$$



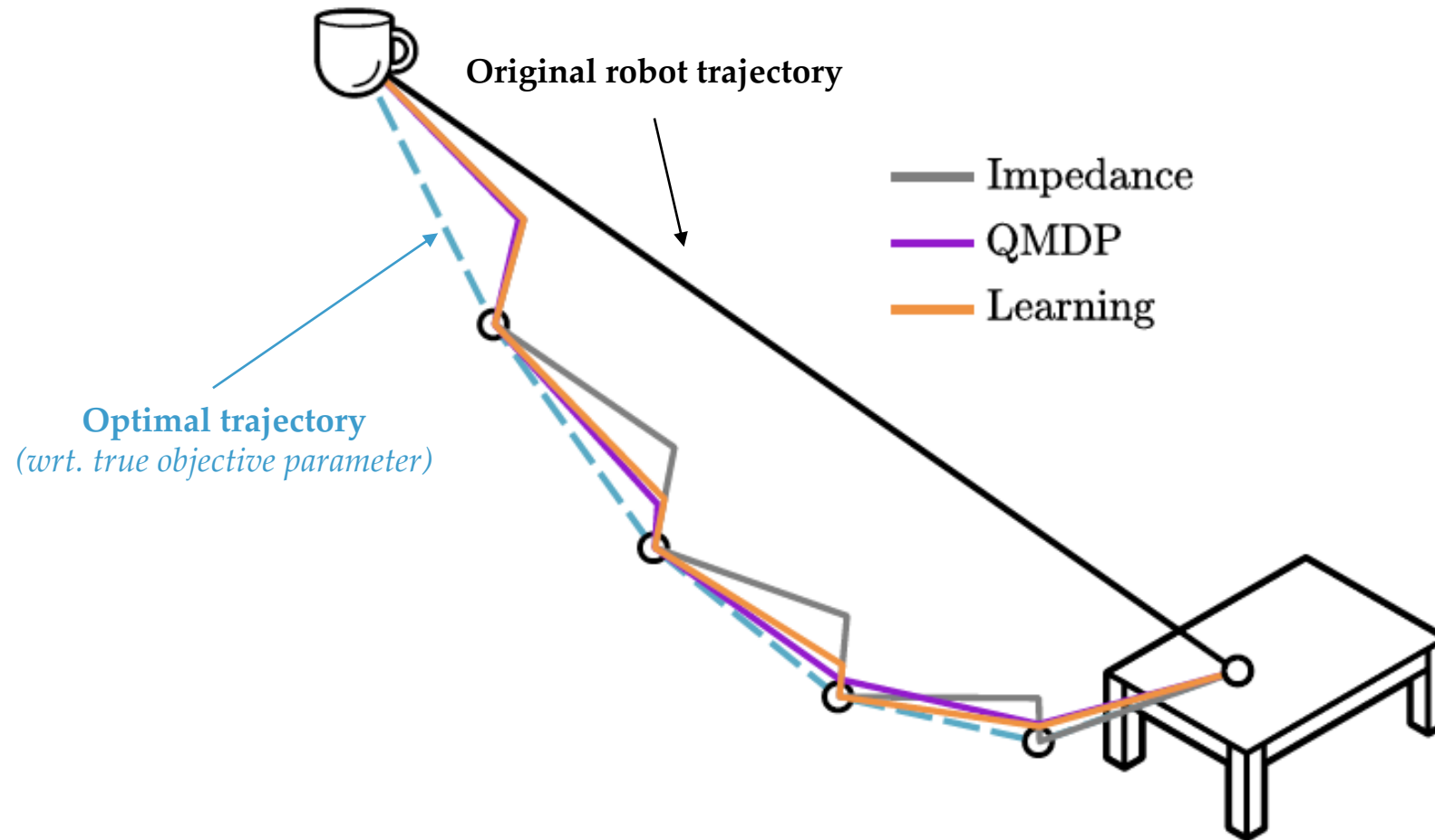
Online Learning from pHRI

Replan with new objective

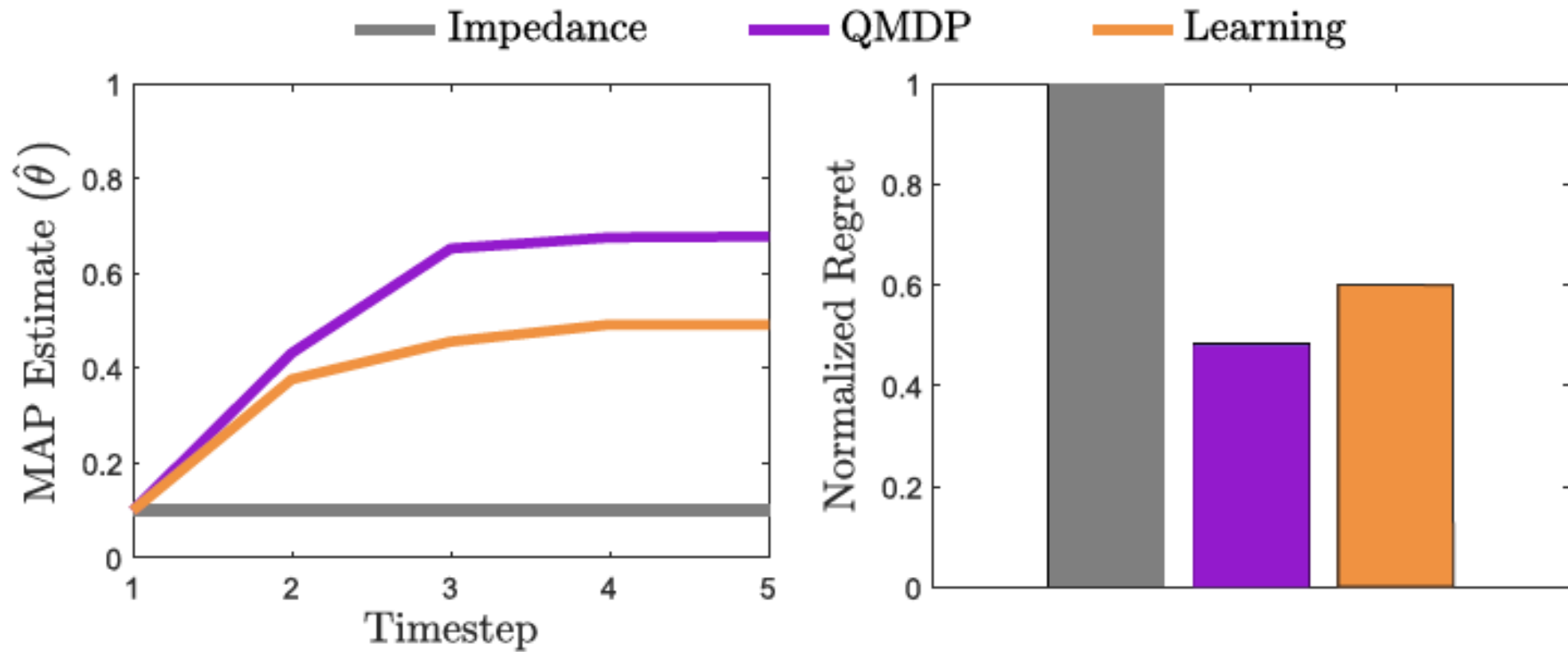
$$\zeta_{\mathcal{R}}^{t+1} \leftarrow \arg \max_{\xi} \hat{\theta}^{t+1} \Phi(\xi)$$



Learning vs. QMDP vs. No Learning.

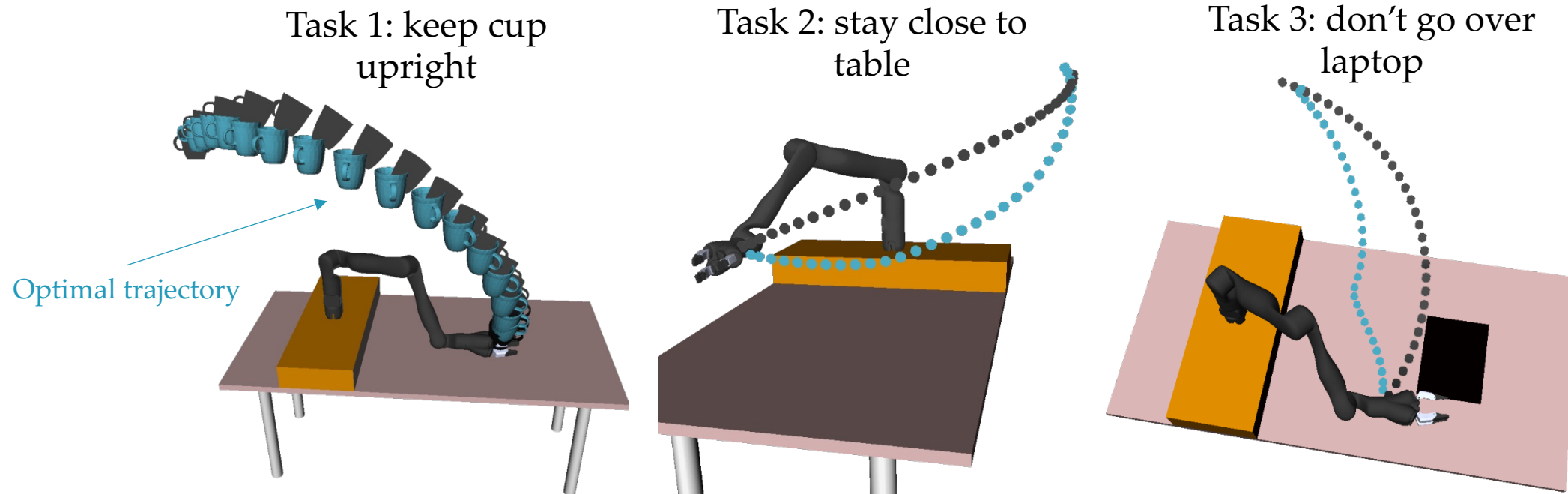


Learning vs. QMDP vs. No Learning.



User Study

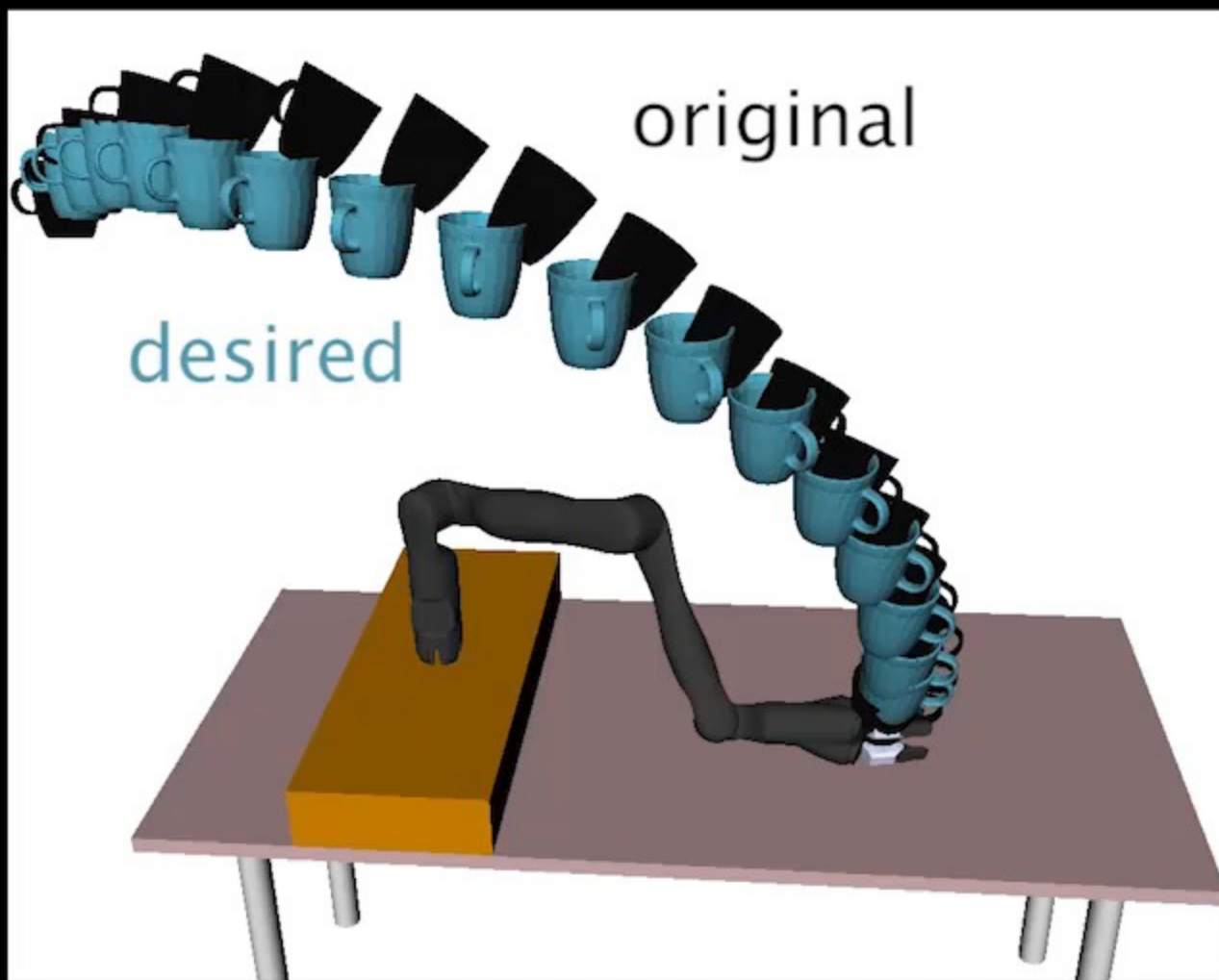
- 3 household manipulation tasks in a shared workspace with **10 participants**
- The robot moves from start to goal pose with an **initially incorrect objective**



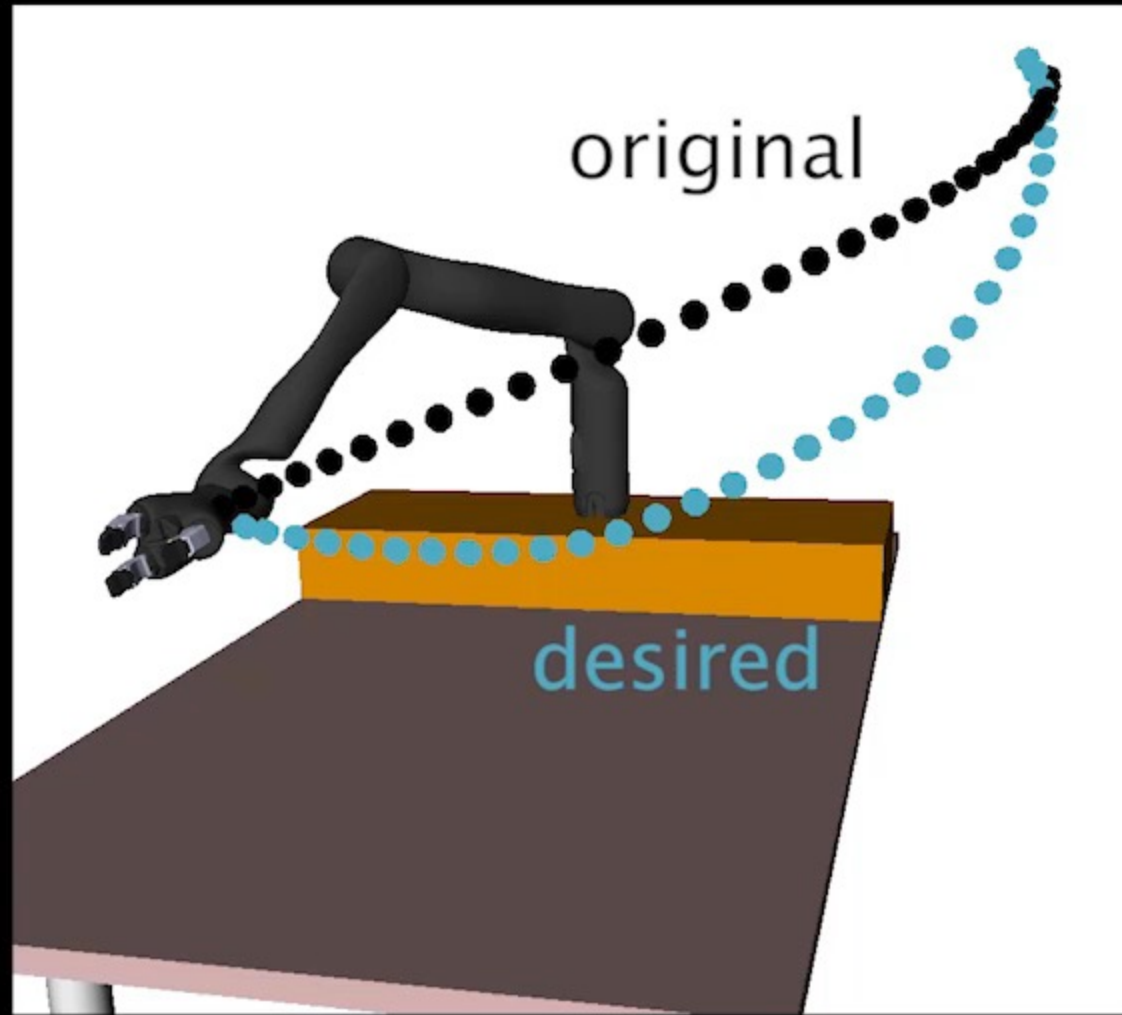
Hypotheses

H1. Learning significantly decreases **interaction time, effort,** and cumulative trajectory **cost.**

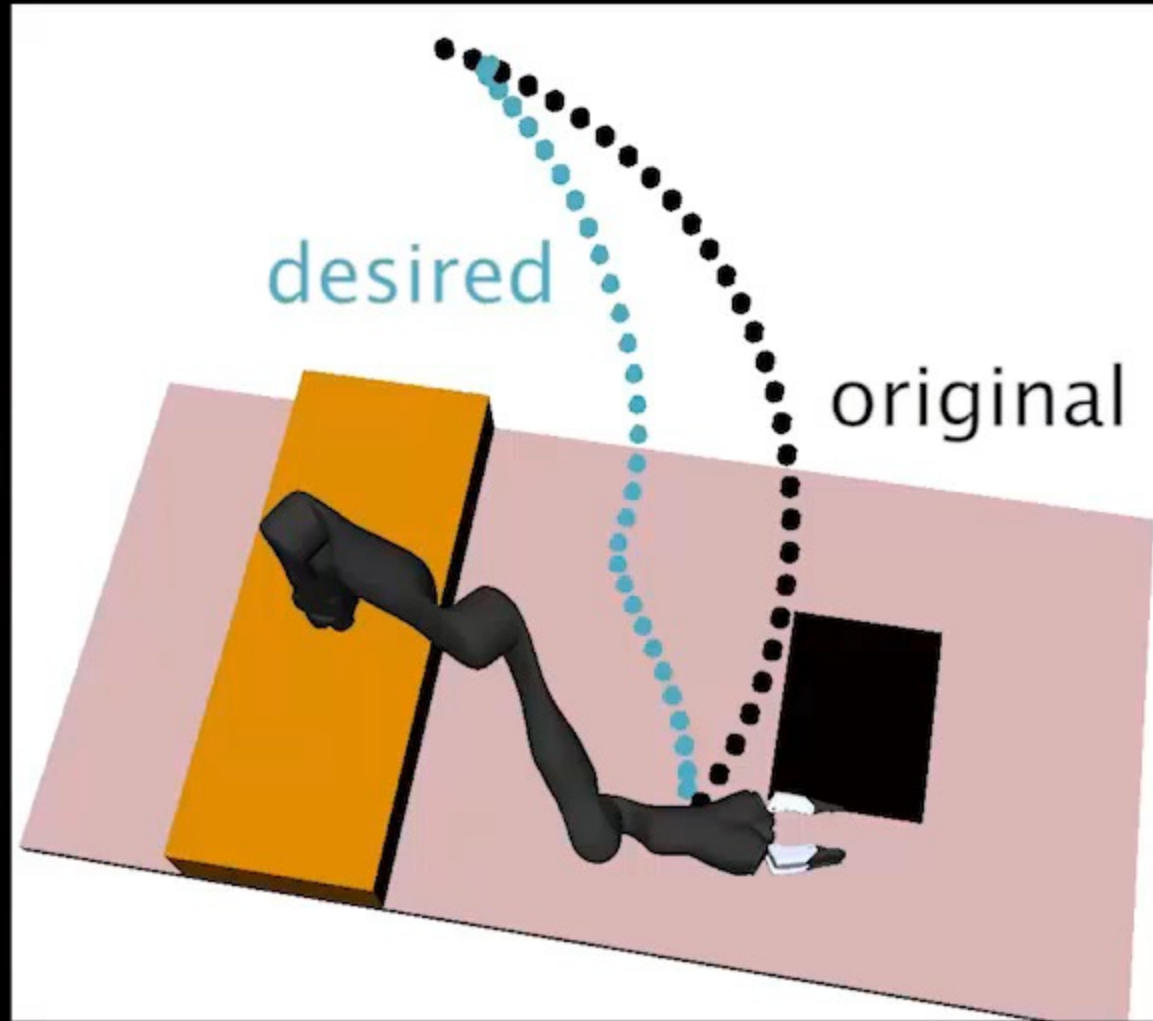
H2. Participants will better know if the robot **understood their preferences,** feel less interaction **effort,** perceive the robot as more **predictable,** and believe the robot is more **collaborative** in the learning condition.



Task 1: keep the cup upright



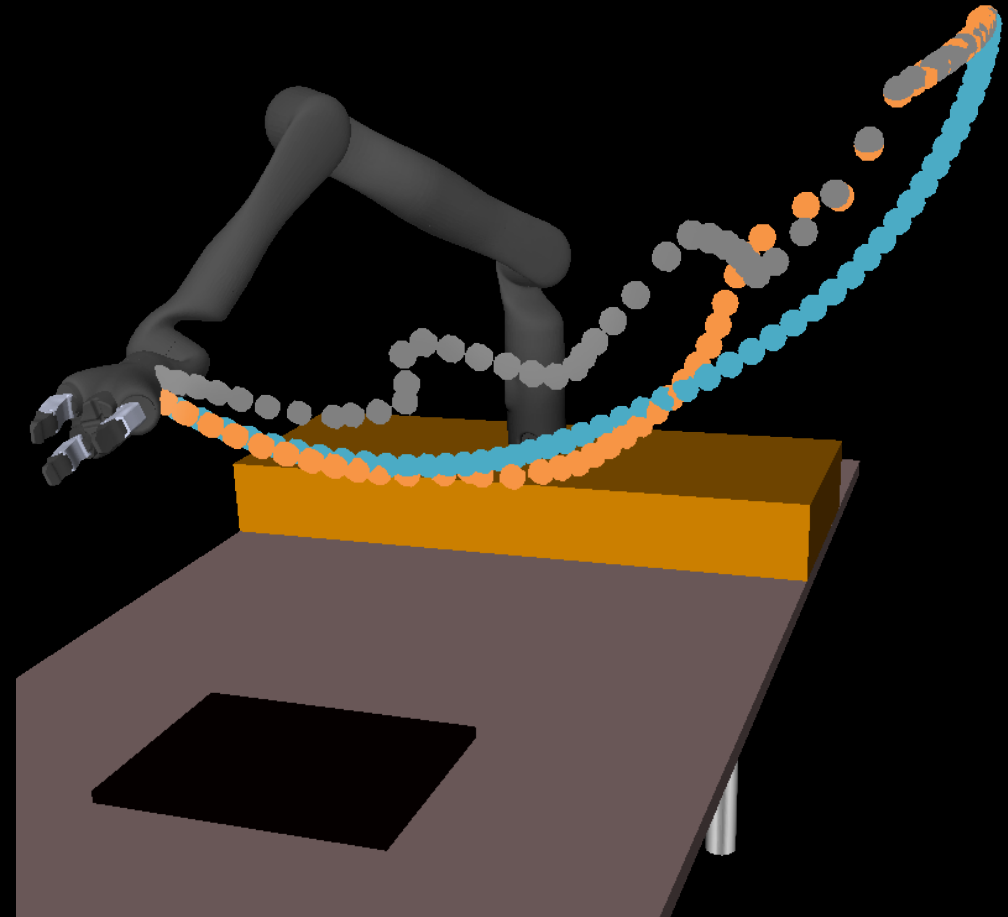
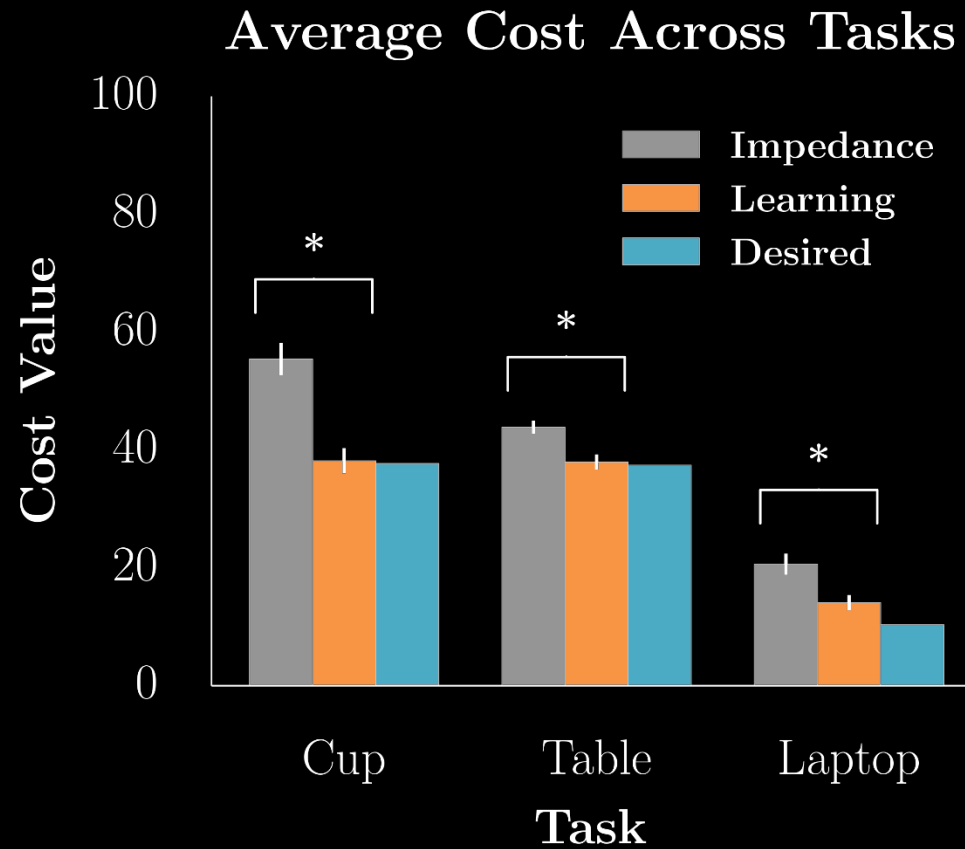
Task 2: stay close to the table



Task 3: don't move over the laptop

Results - Objective

Performed factorial repeated measures ANOVA

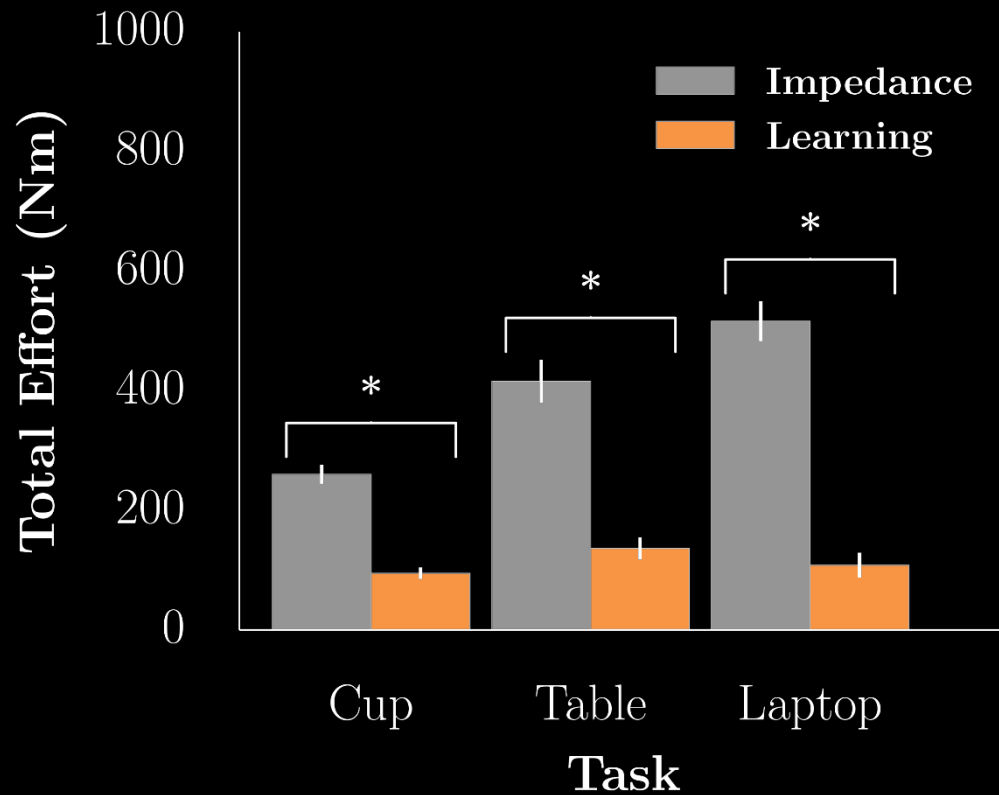


* = $p < 0.0001$

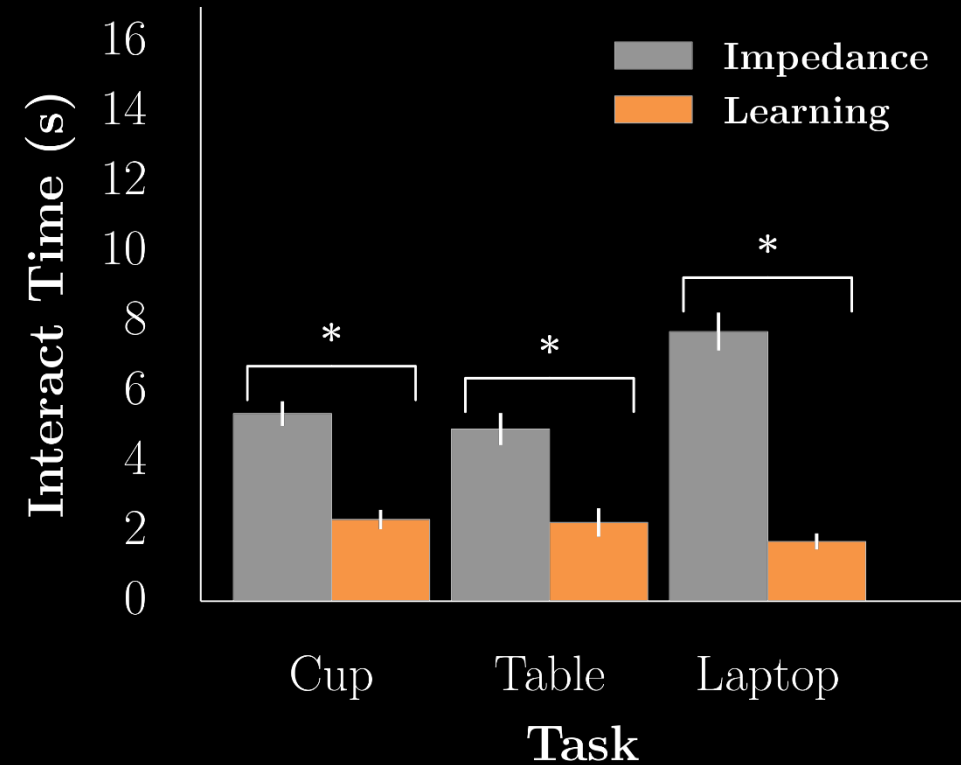
Results - Objective

Performed factorial repeated measures ANOVA

Average Total Human Effort



Average Total Interaction Time



* = $p < 0.0001$

Results - Subjective

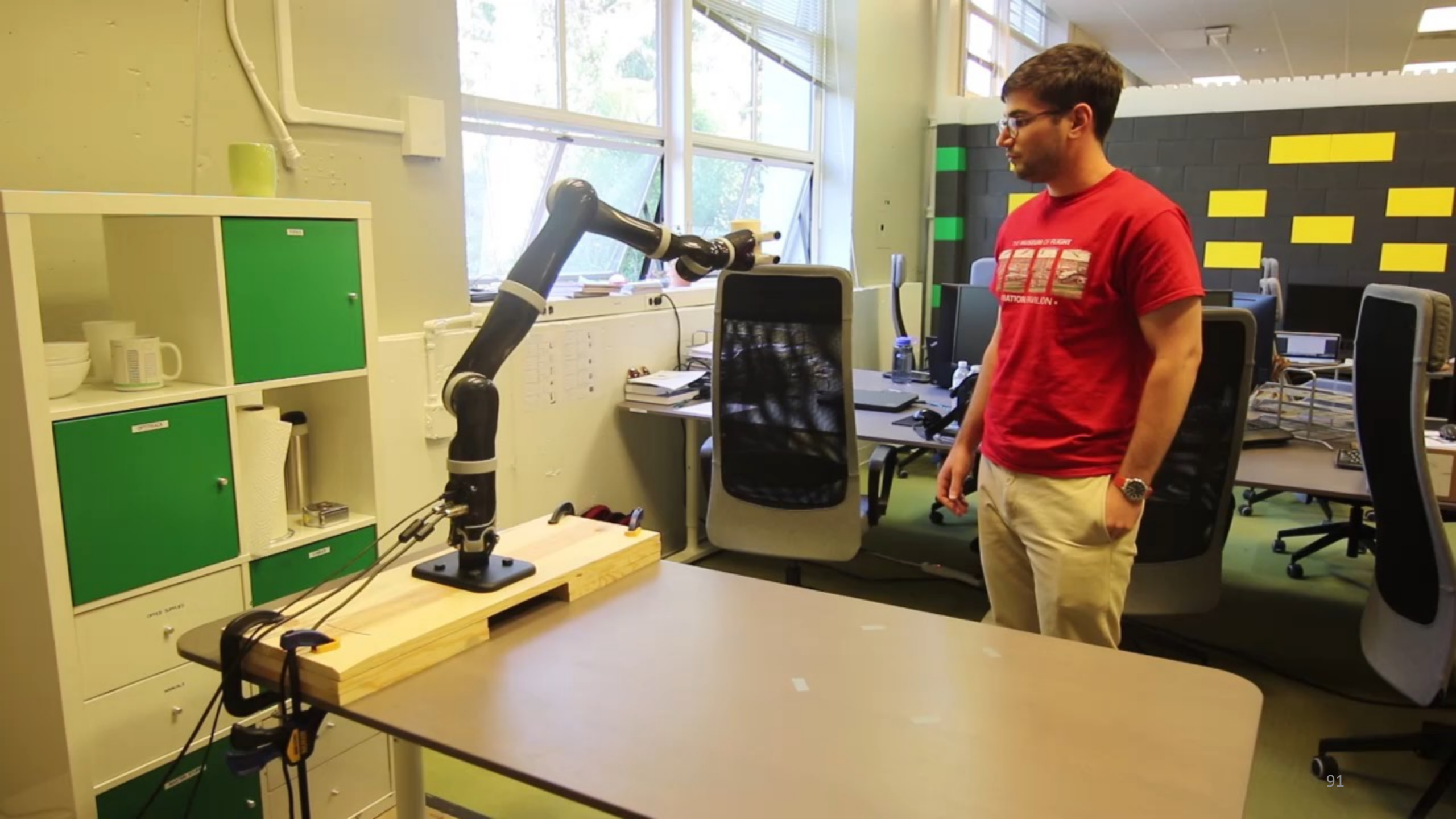
Performed one-way repeated measures ANOVA

| | Questions | Cronbach's α | F(1,9) | p-value |
|---------------|--|---------------------|--------|---------|
| understanding | By the end, the robot understood how I wanted it to do the task. Even by the end, the robot still did not know how I wanted it to do the task. The robot learned from my corrections. The robot did not understand what I was trying to accomplish. | 0.94 | 118.56 | <.0001 |
| effort | I had to keep correcting the robot. The robot required minimal correction. | 0.98 | 85.25 | <.0001 |
| predict | It was easy to anticipate how the robot will respond to my corrections. | 0.8 | 0.06 | 0.82 |
| | The robot's response to my corrections was surprising. | 0.8 | 0.89 | 0.37 |
| collab | The robot worked with me to complete the task. The robot did not collaborate with me to complete the task. | 0.98 | 55.86 | <.0001 |

Results - Subjective

Performed one-way repeated measures ANOVA

| | Questions | Cronbach's α | F(1,9) | p-value |
|---------------|---|---------------------|--------|---------|
| understanding | By the end, the robot understood how I wanted it to do the task. | 0.94 | 118.56 | <.0001 |
| | Even by the end, the robot still did not know how I wanted it to do the task. | | | |
| | The robot learned from my corrections. | | | |
| | The robot did not understand what I was trying to accomplish. | | | |
| effort | I had to keep correcting the robot. | 0.98 | 85.25 | <.0001 |
| | The robot required minimal correction. | | | |
| predict | It was easy to anticipate how the robot will respond to my corrections. | 0.8 | 0.06 | 0.82 |
| | The robot's response to my corrections was surprising. | 0.8 | 0.89 | 0.37 |
| collab | The robot worked with me to complete the task. | 0.98 | 55.86 | <.0001 |
| | The robot did not collaborate with me to complete the task. | | | |



What other kind of human data (or feedback) can we leverage?

Demonstrations

Corrections

Comparisons
(preferences)

“Initial state”
(i.e., preferences implicit in the state of the world)

Proxy reward

Language

Off-switch

... (and more)

Correcting Robot Plans with Natural Language Feedback

Pratyusha Sharma^{‡§}, Balakumar Sundaralingam[‡], Valts Blukis[‡], Chris Paxton[‡],
Tucker Hermans^{‡¶}, Antonio Torralba[§], Jacob Andreas[§], Dieter Fox^{‡†}
[‡] NVIDIA, [§] MIT, [¶] University of Utah, [†] University of Washington

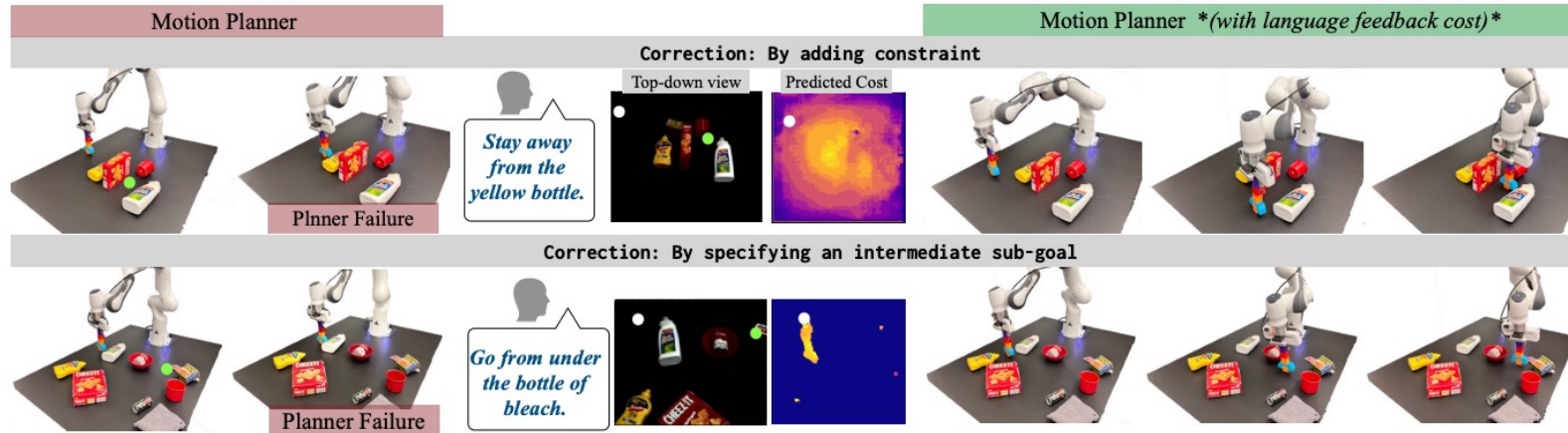


Fig. 1: Robots often fail to do what we want. This can happen for many reasons including mis-specification of goals, failure to anticipate what satisfying plans will do, and because optimization sometimes fails. We show how language can be used to update the underlying cost of a planner to improve task performance. Our approach can use language to specify corrections by a) the addition of constraints or b) specifying intermediate sub-goals for the planner .

Abstract—When humans design cost or goal specifications for robots, they often produce specifications that are ambiguous, under-specified, or beyond planners’ ability to solve. In these cases, *corrections* provide a valuable tool for human-in-the-loop robot control. Corrections might take the form of new goal specifications, new constraints (e.g. to avoid specific objects), or hints for planning algorithms (e.g. to visit specific waypoints). Existing correction methods (e.g. using a joystick or direct manipulation of an end effector) require full teleoperation or real-time interaction. In this paper, we explore *natural language* as an expressive and

This objective function takes the form of a cost function in an optimization-based planning and control framework for manipulation. Our use of language contrasts with previous work where corrective input of robot behavior came from joystick control [36, 33], kinesthetic feedback [27, 19, 6], or spatial labelling of constraints [45, 9]. Kinesthetic and joystick feedback allows for fine-grained control, but typically requires prior expertise and undivided attention from the user, reducing

Physical Corrections



Bajcsy, Andrea, et al. "Learning robot objectives from physical human interaction." CoRL, 2017.

Language Corrections



Go from under the bottle of bleach.

Sharma, Pratyusha, et al. "Correcting robot plans with natural language feedback." RSS, 2022.

trajectory

$$R(\xi; \theta) = \theta^T \phi(\xi)$$

hand-designed features

robot state & observation

NN weights

$$R((q, o), L; \theta) =$$

language

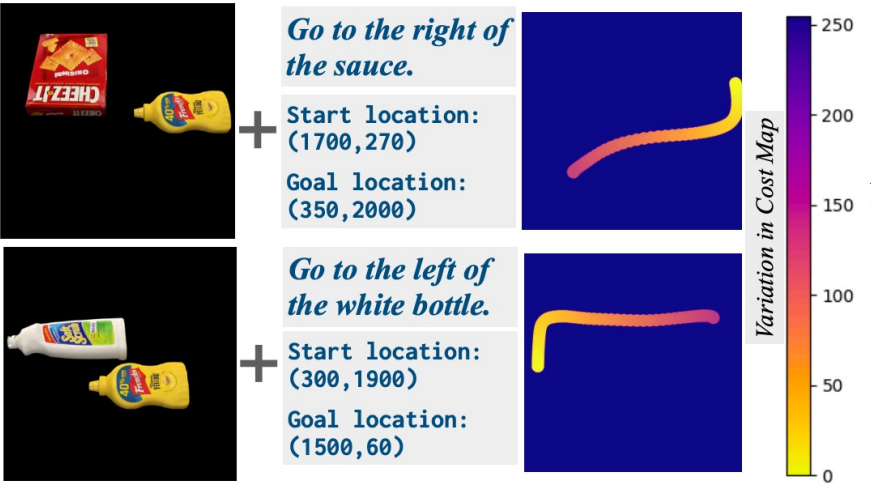


position & velocity reward map

Language Corrections

Offline, train $R(\cdot)$

Training Data

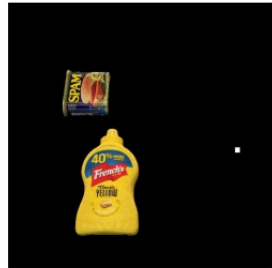


Environment + Start location



Instruction

Go to the bottom of the Cheeze-It box.



Go behind the blue colored container.

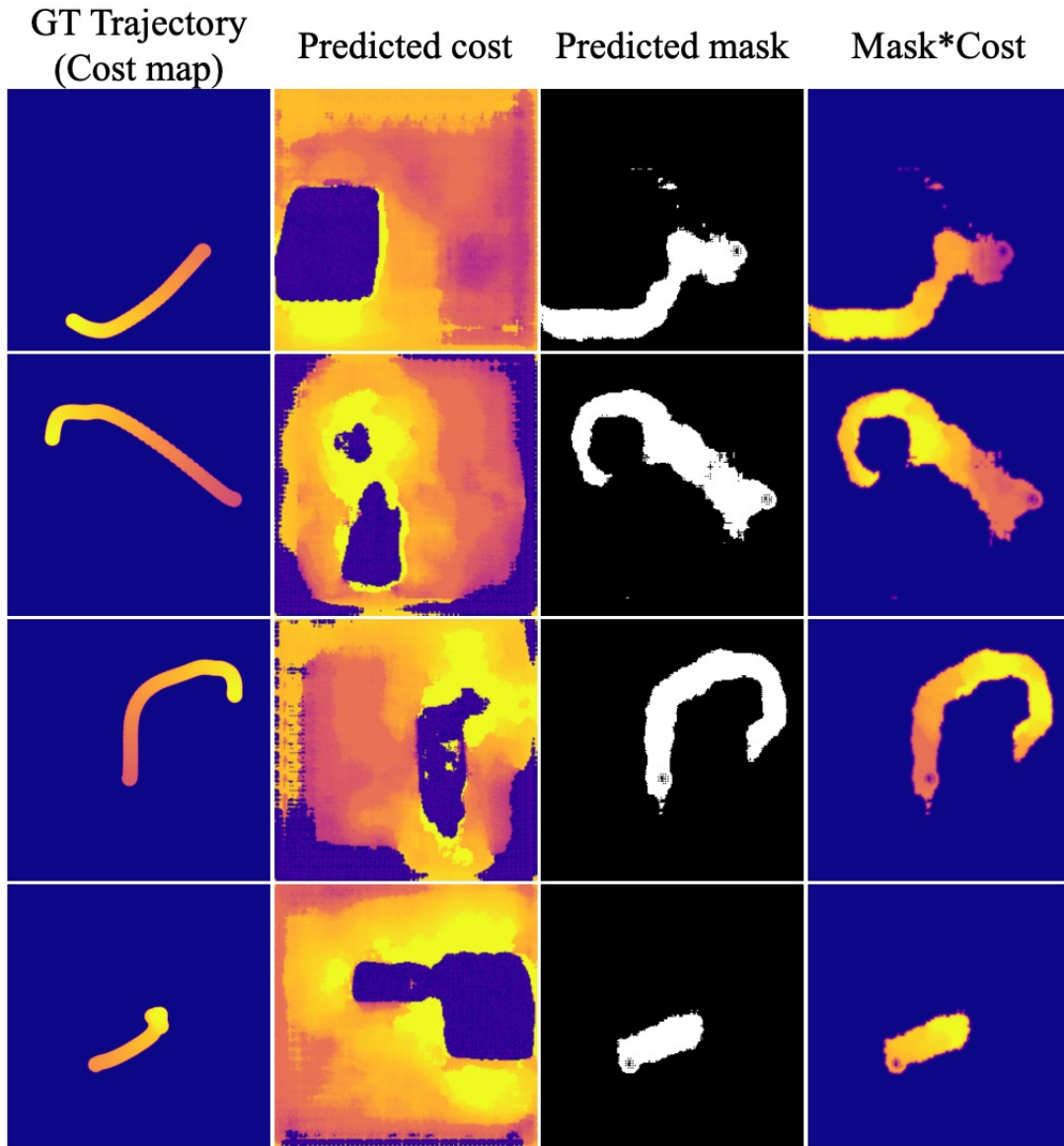


Go to the right of the spam can.



Go in front of the red box.

Online, predict $R(\cdot)$ given specific L



Benefits of Natural Language Feedback(1)

Same correction can be applied to multiple environments in need:

"Hey robot! Go to the left of the bleach first."



Can we *unify learning* from diverse feedback types?

Demonstrations

Comparisons
(preferences)

Proxy reward

Off-switch

Corrections

“Initial state”
(i.e., preferences implicit in the state of the world)

Language

... (and more)