Learning Robot Objectives from Physical Human Interaction

Andrea Bajcsy*
University of California, Berkeley
abajcsy@berkeley.edu

Dylan P. Losey*
Rice University
dlosey@rice.edu

Marcia K. O'Malley
Rice University
omalleym@rice.edu

Anca D. Dragan
University of California, Berkeley
anca@berkeley.edu

Abstract: When humans and robots work in close proximity, physical interaction is inevitable. Traditionally, robots treat physical interaction as a disturbance, and resume their original behavior after the interaction ends. In contrast, we argue that physical human interaction is *informative*: it is useful information about how the robot *should* be doing its task. We formalize learning from such interactions as a dynamical system in which the task objective has parameters that are part of the *hidden* state, and physical human interactions are observations about these parameters. We derive an online approximation of the robot's optimal policy in this system, and test it in a user study. The results suggest that learning from physical interaction leads to better robot task performance with less human effort.

Keywords: physical human-robot interaction, learning from demonstration

1 Introduction

Imagine a robot performing a manipulation task next to a person, like moving the person's coffee mug from a cabinet to the table (Fig. 1). As the robot is moving, the person might notice that the robot is carrying the mug too high above the table. Knowing that the mug would break if it were to slip and fall from so far up, the person easily intervenes and starts pushing the robot's end-effector down to bring the mug closer to the table. In this work, we focus on how the robot should then *respond* to such physical human-robot interaction (pHRI).

Several reactive control strategies have been developed to deal with pHRI [1, 2, 3]. For instance, when a human applies a force on the robot, it can render a desired impedance or switch to gravity compensation and allow the human to easily move the robot around. In these strategies, the moment the human lets go of the robot, it resumes its original behavior—our robot from earlier would go back to carrying the mug too high, requiring the person to continue intervening until it finished the task (Fig. 1, left).

Although such control strategies guarantee fast reaction to unexpected forces, the robot's return to its original motion stems from a fundamental limitation of traditional pHRI strategies: they miss the fact that human interventions are often *intentional* and occur because the robot is doing something wrong. While the robot's original behavior may have been optimal with respect to the robot's pre-defined objective function, the fact that a human intervention was necessary implies that this objective function was not quite right.

Our insight is that because pHRI is intentional, it is also informative*—it provides observations about the correct robot objective function, and the robot can leverage these observations to learn that correct objective.*

Returning to our example, if the person is applying forces to push the robot's end-effector closer to the table, then the robot should *change* its objective function to reflect this preference, and complete the rest of the current task accordingly, keeping the mug lower (Fig. 1, right). Ultimately, human interactions should not be thought of as disturbances, which perturb the robot from its desired behavior, but rather as corrections, which *teach* the robot its desired behavior.

In this paper, we make the following contributions:

Formalism. We formalize reacting to pHRI as the problem of acting in a dynamical system to optimize an objective function, with two caveats: 1) the objective function has *unknown* parameters

1st Conference on Robot Learning (CoRL 2017), Mountain View, United States.

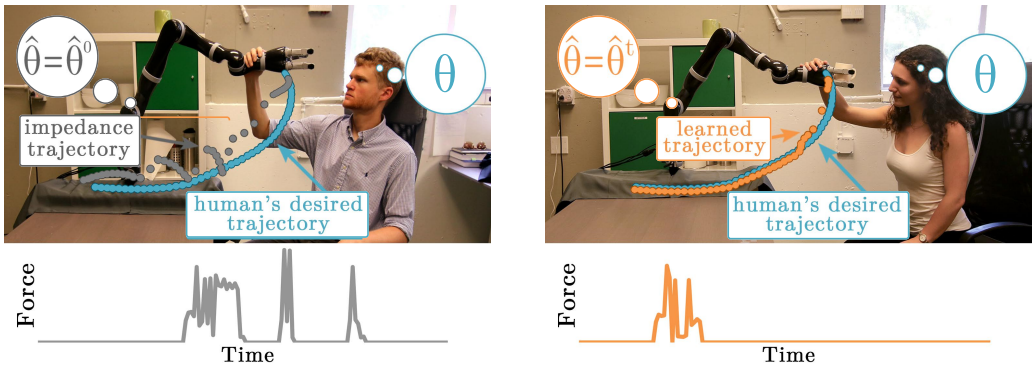


Figure 1: A person interacts with a robot that treats interactions as disturbances (left), and a robot that learns from interactions (right). When humans are treated as disturbances, force plots reveal that people have to continuously interact since the robot returns to its original, incorrect trajectory. In contrast, a robot that learns from interactions requires minimal human feedback to understand how to behave (i.e., go closer to the table).

θ, and 2) human interventions serve as *observations* about these unknown parameters: we model human behavior as approximately optimal with respect to the true objective. As stated, this problem is an instance of a Partially Observable Markov Decision Process (POMDP). Although we cannot solve it in real-time using POMDP solvers, this formalism is crucial to converting the problem of reacting to pHRI into a clearly defined optimization problem. In addition, our formalism enables pHRI approaches to be justified and compared in terms of this optimization criterion.

Online Solution. We introduce a solution that adapts learning from demonstration approaches to our online pHRI setting [4, 5], but derive it as an approximate solution to the problem above. This enables the robot to adapt to pHRI in real-time, as the current task is unfolding. Key to this approximation is simplifying the observation model: rather than interpreting instantaneous forces as noisy-optimal with respect to the *value function* given θ, we interpret them as implicitly inducing a noisy-optimal desired *trajectory*. Reasoning in trajectory space enables an efficient approximate online gradient approach to estimating θ.

User Study. We conduct a user study with the JACO2 7-DoF robotic arm to assess how *online* learning from *physical interactions* during a task affects the robot's objective performance, as well as subjective participant perceptions.

Overall, our work is a first step towards learning robot objectives online from pHRI.

2 Related Work

We propose using pHRI to correct the robot's objective function *while the robot is performing its current task*. Prior research has focused on (a) control strategies for reacting to pHRI *without* updating the robot's objective function, or (b) learning the robot's objectives—from *offline* demonstrations—in a manner that generalizes to *future* tasks, but does not change the behavior during the *current* task. An exception is shared autonomy work, which does correct the robot's objective function online, but only when the objective is parameterized *by the human's desired goal in free-space*.

Control Strategies for Online Reactions to pHRI. A variety of control strategies have been developed to ensure safe and responsive pHRI. They largely fall into three categories [6]: impedance control, collision handling, and shared manipulation control. Impedance control [1] relates deviations from the robot's planned trajectory to interaction torques. The robot renders a virtual stiffness, damping, and/or inertia, allowing the person to push the robot away from its desired trajectory, but the robot always returns to its original trajectory after the interaction ends. Collision handling methods [2] include stopping, switching to gravity compensation, or re-timing the planned trajectory if a collision is detected. Finally, shared manipulation [3] refers to role allocation in situations where the human and the robot are collaborating. These control strategies for pHRI work in real-time, and enable the robot to safely adapt to the human's actions; however, the robot fails to leverage these interventions to update its understanding of the task—left alone, the robot would continue to perform the task *in the same way* as it had planned before any human interactions. By contrast, we focus on enabling robots to adjust how they perform the current task in real time.

Offline Learning of Robot Objective Functions. Inverse Reinforcement Learning (IRL) methods focus explicitly on inferring an unknown objective function, but do it *offline*, after passively observing expert trajectory demonstrations [7]. These approaches can handle noisy demonstrations [8], which become observations about the true objective [9], and can acquire demonstrations through

physical kinesthetic teaching [10]. Most related to our work are approaches which learn from corrections of the robot's trajectory, rather than from demonstrations [4, 5, 11]. Our work, however, has a different goal: while these approaches focus on the robot doing better *the next time* it performs the task, we focus on the robot completing its *current* task correctly. Our solution is analogous to online Maximum Margin Planning [4] and co-active learning [5] for this new setting, but one of our contributions is to derive their update rule as an approximation to our pHRI problem.

Online Learning of Human Goals. While IRL can learn the robot's objective function after one or more demonstrations of a task, online inference is possible when the objective is simply to reach a *goal state*, and the robot moves through free space [12, 13, 14]. We build on this work by considering *general objective parameters*; this requires a more complex (non-analytic and difficult to compute) observation model, along with additional approximations to achieve online performance.

3 Learning Robot Objectives Online from pHRI

3.1 Formalizing Reacting to pHRI

We consider settings where a robot is performing a day-to-day task next to a person, but is not doing it correctly (e.g., is about to spill a glass of water), or not doing it in a way that matches the person's preferences (e.g., is getting too close to the person). Whenever the person physically intervenes and corrects the robot's motion, the robot should react accordingly; however, there are many strategies the robot could use to react. Here, we formalize the problem as a dynamical system with a true objective function that is known by the person but not known by the robot. This formulation interprets the human's physical forces as intentional, and implicitly defines an *optimal* strategy for reacting.

Notation. Let x denote the robot's state (its position and velocity) and u_R the robot's action (the torque it applies at its joints). The human physically interacts with the robot by applying external torque u_H. The robot transitions to a next state defined by its dynamics, $\dot{x} = f(x, u_R + u_H)$, where both the human and robot can influence the robot's motion.

POMDP Formulation. The robot optimizes a reward function $r(x, u_R, u_H; \theta)$, which trades off between correctly completing the task and minimizing human effort

$$r(x, u_R, u_H; \theta) = \theta^T \phi(x, u_R, u_H) - \lambda ||u_H||^2 \tag{1}$$

Following prior IRL work [15, 4, 8], we parameterize the task-related part of this reward function as a linear combination of features ϕ with weights θ. Note that we assume the relevant set of features for each task are given, and we will not explore feature selection within this work.

Here θ encapsulates the true objective, such as moving the glass slowly, or keeping the robot's end-effector farther away from the person. Importantly, this parameter *is not known by the robot*—robots will not always know the right way to perform a task, and certainly not the human-preferred way. If the robot knew θ, this would simply become an MDP formulation, where the states are x, the actions are u_R, the reward is r, and the person would never need to intervene.

Uncertainty over θ, however, turns this into a POMDP formulation, where θ is a hidden part of the state. Importantly, *the human's actions are observations about* θ under some observation model $P(u_H|x, u_R; \theta)$. These observations u_H are atypical in two ways: (a) they affect the robot's reward, as in [13], and (b) they influence the robot's state, but we don't necessarily want to account for that when planning – the robot should not rely on the human to move the robot; rather the robot should consider u_H only for its information value.

Observation Model. We model the human's interventions as corrections which approximately maximize the robot's reward. More specifically, we assume the noisy-rational human selects an action u_H that, when combined with the robot's action u_R, leads to a high Q-value (state-action value) *assuming the robot will behave optimally after the current step* (i.e., assuming the robot knows θ)

$$P(u_H \mid x, u_R; \theta) \propto e^{Q(x, u_R + u_H; \theta)} \tag{2}$$

Our choice of (2) stems from maximum entropy assumptions [8], as well as the Bolzmann distributions used in cognitive science models of human behavior [16].

Aside. We are *not* formulating this as a POMDP to solve it using standard POMDP solvers. Instead, our goal is to clarify the underlying problem formulation and the existence of an optimal strategy.

3.2 Approximate Solution

Since POMDPs cannot be solved tractably for high-dimensional real-world problems, we make several approximations to arrive at an online solution. We first separate estimation from finding the

optimal policy, and approximate the policy by separating planning from control. We then simplify the estimation model, and use maximum a posteriori estimate (MAP) instead the full belief over θ.

QMDP. Similar to [13], we approximate our POMDP using a QMDP by assuming the robot will obtain full observability at the next time step [17]. Let b denote the robot's current belief over θ. The QMDP simplifies into two subproblems: (a) *finding the robot's optimal policy* given b

$$Q(x, u_R, b) = \int b(\theta) Q(x, u_R, \theta) d\theta \tag{3}$$

where $\arg\max_{u_R} Q(x, u_R, b)$ evaluated at every state yields the optimal policy, and (b) *updating our belief over* θ given a new observation. Unlike the actual POMDP solution, here the robot will not try to gather information.

From Belief to Estimator. Rather than planning with the belief b, we plan with only the MAP of $\hat{\theta}$.

From Policies to Trajectories (Action). Computing Q in continuous state, action, and belief spaces is still not tractable. We thus separate *planning* and *control*. At every time step t, we do two things.

First, given our current $\hat{\theta}^t$, we *replan* a trajectory $\xi = x^{0:T} \in \Xi$ that optimizes the task-related reward. Let $\theta^T \Phi(\xi)$ be the cumulative reward, where $\Phi(\xi)$ is the total feature count along trajectory ξ such that $\Phi(\xi) = \sum_{x^t \in \xi} \phi(x^t)$. We use a trajectory optimizer [18] to replan the robot's desired trajectory ξ_R^t

$$\xi_R^t = \arg\max_{\xi} \ \hat{\theta}^t \cdot \Phi(\xi) \tag{4}$$

Second, once ξ_R^t has been planned, we *control* the robot to track this desired trajectory. We use impedance control, which allows people to change the robot's state by exerting torques, and provides compliance for human safety [19, 6, 1]. After feedback linearization [20], the equation of motion under impedance control becomes

$$M_R(\ddot{q}^t - \ddot{q}_R^t) + B_R(\dot{q}^t - \dot{q}_R^t) + K_R(q^t - q_R^t) = u_H^t \tag{5}$$

Here M_R, B_R, and K_R are the desired inertia, damping, and stiffness, $x = (q, \dot{q})$, where q is the robot's joint position, and $q_R \in \xi_R$ denotes the desired joint position. Within our experiments, we implemented a simplified impedance controller without feedback linearization

$$u_R^t = B_R(\dot{q}_R^t - \dot{q}^t) + K_R(q^t - q_R^t) \tag{6}$$

Aside. When the robot is not updating its estimate $\hat{\theta}$, then $\xi_R^t = \xi_R^{t-1}$, and our solution reduces to using impedance control to track an unchanging trajectory [2, 19].

From Policies to Trajectories (Estimation). We still need to address the second QMDP subproblem: updating $\hat{\theta}$ after each new observation. Unfortunately, evaluating the observation model (2) for any given θ is difficult, because it requires computing the Q-value function for that θ. Hence, we will again leverage a simplification from policies to trajectories in order to update our MAP of θ.

Instead of attempting to directly relate u_H to θ, we propose an intermediate step; we interpret each human action u_H via a *intended trajectory*, ξ_H, that the human wants the robot to execute. To compute the intended trajectory ξ_H from ξ_R and u_H, we propagate the deformation caused by u_H along the robot's current trajectory ξ_R

$$\xi_H = \xi_R + \mu A^{-1} U_H \tag{7}$$

where $\mu > 0$ scales the magnitude of the deformation, A defines a norm on the Hilbert space of trajectories and dictates the deformation shape [21], $U_H = u_H$ at the current time, and $U_H = 0$ at all other times. During experiments we here used a norm A based on acceleration [21], but we will explore learning the choice of this norm in future work.

Importantly, our simplification from observing human action u_H to implicitly observing the human's intended trajectory ξ_H means we no longer have to evaluate the Q-value of $u_R + u_H$ given some θ value. Instead, the observation model now depends on the total reward of the implicitly observed trajectory:

$$P(\xi_H \mid \xi_R, \theta) \propto e^{\theta^T \Phi(\xi_H) - \lambda ||u_H||^2} \approx e^{\theta^T \Phi(\xi_H) - \lambda ||\xi_H - \xi_R||^2} \tag{8}$$

This is analogous to (2), but in trajectory space—a distribution over implied trajectories, given θ and the current robot trajectory.

Algorithm 1 Online Learning from pHRI

Initialize: $\hat{\theta}^0, \quad \xi_R^0 \leftarrow TrajOpt(\hat{\theta}^0)$

for $t = 0$ to T **do**

$\xi_H^t \leftarrow \xi_R^t + \mu A^{-1} U_H^t$

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

$\xi_R^{t+1} \leftarrow TrajOpt(\hat{\theta}^{t+1})$

$u_R^t \leftarrow Impedance(\xi_R^t, x^t)$

end for

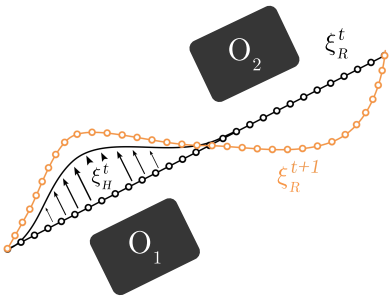


Figure 2: Algorithm (left) and visualization (right) of one iteration of our online learning from pHRI method in an environment with two obstacles O_1, O_2. The originally planned trajectory, ξ_R^t (black dotted line), is deformed by the human's force into the human's preferred trajectory, ξ_H^t (solid black line). Given these two trajectories, we compute an online update of θ and can replan a better trajectory ξ_R^{t+1} (orange dotted line).

3.3 Online Update of the θ Estimate

The probability distribution over θ at time step t is $P(\xi_H^0, .., \xi_H^t | \theta, \xi_R^0, .., \xi_R^t) P(\theta)$. However, since θ is continuous, and the observation model is not Gaussian, we opt not to track the full belief, but rather to track the maximum a posteriori estimate (MAP). Our update rule for this estimate will reduce to *online* Maximum Margin Planning [4] if we treat ξ_H as the demonstration, and to co-adaptive learning [5], if we treat ξ_H as the original trajectory with one waypoint corrected. One of our contributions, however, is to derive this update rule from our MaxEnt observation model in (8).

MAP. Assuming the observations are conditionally independent given θ, the MAP for time $t+1$ is

$$\hat{\theta}^{t+1} = \arg\max_{\theta} P(\xi_H^0, .., \xi_H^t \mid \xi_R^0, .., \xi_R^t, \theta) P(\theta) = \arg\max_{\theta} \sum_{\tau=0}^{t} \log P(\xi_H^\tau \mid \xi_R^\tau, \theta) + \log P(\theta) \tag{9}$$

Inspecting the right side of (9), we need to define both $P(\xi_H | \xi_R, \theta)$ and the prior $P(\theta)$. To approximate $P(\xi_H | \xi_R, \theta)$, we use (8) with Laplace's method to compute the normalizer. Taking a second-order Taylor series expansion of the objective function about ξ_R, the robot's current best guess at the optimal trajectory, we obtain a Gaussian integral that can be evaluated in closed form

$$P(\xi_H \mid \xi_R, \theta) = \frac{e^{\theta^T \Phi(\xi_H) - \lambda ||\xi_H - \xi_R||^2}}{\int e^{\theta^T \Phi(\xi) - \lambda ||\xi - \xi_R||^2} \, d\xi} \approx e^{\theta^T \left(\Phi(\xi_H) - \Phi(\xi_R)\right) - \lambda ||\xi_H - \xi_R||^2} \tag{10}$$

Let $\hat{\theta}^0$ be our initial estimate of θ. We propose the prior

$$P(\theta) = e^{-\frac{1}{2\alpha} ||\theta - \hat{\theta}^0||^2} \tag{11}$$

where α is a positive constant. Substituting (10) and (11) into (9), the MAP reduces to

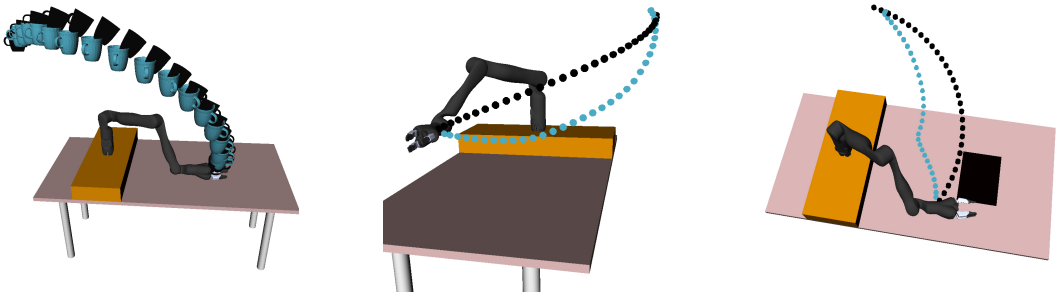
$$\hat{\theta}^{t+1} \approx \arg\max_{\theta} \left\{ \sum_{\tau=0}^{t} \theta^T \big(\Phi(\xi_H^\tau) - \Phi(\xi_R^\tau)\big) - \frac{1}{2\alpha} ||\theta - \hat{\theta}^0||^2 \right\} \tag{12}$$

Notice that the $\lambda ||\xi_H - \xi_R||^2$ terms drop out, because this penalty for human effort does not explicitly depend on θ. Solving the optimization problem (12) by taking the gradient with respect to θ, and then setting the result equal to zero, we finally arrive at

$$\hat{\theta}^{t+1} = \hat{\theta}^0 + \alpha \sum_{\tau=0}^{t} \big(\Phi(\xi_H^\tau) - \Phi(\xi_R^\tau)\big) = \hat{\theta}^t + \alpha\big(\Phi(\xi_H^t) - \Phi(\xi_R^t)\big) \tag{13}$$

Interpretation. This update rule is actually the online gradient [22] of (9) under our Laplace approximation of the observation model. It has an intuitive interpretation: it shifts the weights in the direction of the human's intended feature count. For example, if ξ_H stays farther from the person than ξ_R, the weights in θ associated with distance-to-person features will increase.

Relation to Prior Work. This update rule is analogous to two related works. First, it would be the online version of Maximum Margin Planning (MMP) [4] if the trajectory ξ_H^t were a new demon-



(a) Task 1: Cup orientation (b) Task 2: Distance to table (c) Task 3: Laptop avoidance

Figure 3: Simulations depicting the robot trajectories for each of the three experimental tasks. The black path represents the original trajectory and the blue path represents the human's desired trajectory.

stration. Unlike MMP, our robot does not complete a trajectory, and only then get a full new demonstration; instead, our ξ_H^t is an estimate of the human's *intended* trajectory based on the force applied *during* the robot's execution of the current trajectory ξ_R^t. Second, the update rule would be co-active learning [5] if the trajectory ξ_H^t were ξ_R^t with one waypoint modified, as opposed to a propagation of u_H^t along the rest of ξ_R^t. Unlike co-active learning, however, our robot receives corrections continually, and continually updates the current trajectory in order to complete the *current* task well. Nonetheless, we are excited to see similar update rules emerge from different optimization criteria.

Summary. We formalized reacting to pHRI as a POMDP with the correct objective parameters as a hidden state, and approximated the solution to enable online learning from physical interaction. At every time step during the task where the human interacts with the robot, we first propagate u_H to implicitly observe the corrected trajectory ξ_H (simplification of the observation model), and then update $\hat{\theta}$ via Equation (13) (MAP instead of belief). We replan with the new estimate (approximation of the optimal policy), and use impedance control to track the resulting trajectory (separation of planning from control). We summarize and visualize this process in Fig. 2.

4 User Study

We conducted an IRB-approved user study to investigate the benefits of *in-task* learning. We designed tasks where the robot began with the wrong objective function, and participants phsyically corrected the robot's behavior[1].

4.1 Experiment Design

Independent Variables. We manipulated the *pHRI strategy* with two levels: *learning* and *impedance*. The robot either used our method (Algorithm 1) to react to physical corrections and re-plan a new trajectory during the task; or used impedance control (our method without updating $\hat{\theta}$) to react to physical interactions and then return to the originally planned trajectory.

Dependent Measures. We measured the robot's performance with respect to the true objective, along with several subjective measures. One challenge in designing our experiment was that each person might have a different internal objective for any given task, depending on their experience and preferences. Since we do not have direct access to every person's internal preferences, we defined the true objective ourselves, and conveyed the objectives to participants by demonstrating the desired optimal robot behavior (see an example in Fig. 3(a), where the robot is supposed to keep the cup upright). We instructed participants to get the robot to achieve this desired behavior with minimal human physical intervention.

For each robot attempt at a task, we evaluated the task related and effort related parts of the objective: $\theta^T \Phi(\xi)$ (a cost to be minimized and not a reward to be maximized in our experiment) and $\sum_t ||u_H^t||_1$. We also evaluate the total amount of time spent interacting physically with the robot. For our subjective measures, we designed 4 multi-item scales shown in Table 1: did participants think the robot understood how they wanted to task done, did they feel like they had to exert a lot of effort to correct the robot, was it easy to anticipate the robot's reactions, and how good of a collaborator was the robot.

Hypotheses:

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

[1]For video footage of the experiment, see: https://www.youtube.com/watch?v=1MkI6DH1mcw

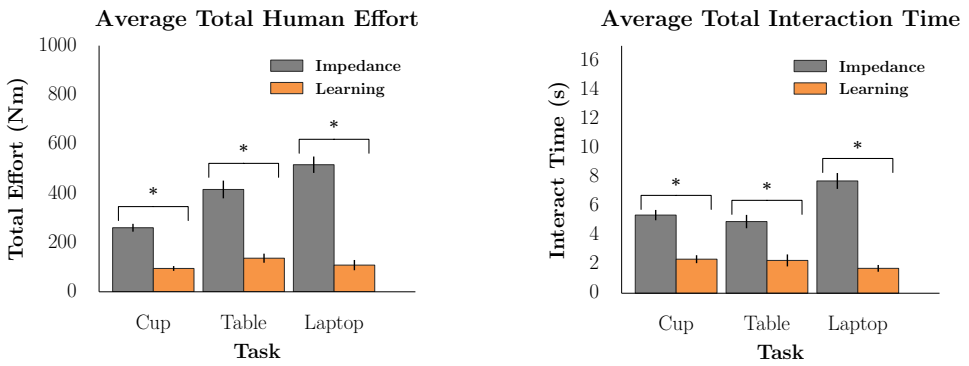


Figure 4: Learning from pHRI decreases human effort and interaction time across all experimental tasks (total trajectory time was 15s). An asterisk (*) means $p < 0.0001$.

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

Tasks. We designed three household manipulation tasks for the robot to perform in a shared workspace (see Fig. 3), plus a familiarization task. As such, the robot's objective function considered two features: velocity and a task-specific feature. For each task, the robot carried a cup from a start to a goal pose with an *initially incorrect objective*, requiring participants to correct its behavior during the task.

During the familiarization task, the robot's original trajectory moved too close to the human. Participants had to physically interact with the robot to get it to keep the cup further away from their body. In Task 1, the robot would not care about tilting the cup mid-task, risking spilling if the cup was too full. Participants had to get the robot to keep the cup upright. In Task 2, the robot would move the cup too high in the air, risking breaking it if it were to slip, and participants had to get the robot to keep it closer to the table. Finally, in Task 3, the robot would move the cup over a laptop to reach it's final goal pose, and participants had to get the robot to keep the cup away from the laptop.

Participants. We used a within-subjects design and counterbalanced the order of the pHRI strategy conditions. In total, we recruited 10 participants (5 male, 5 female, aged 18-34) from the UC Berkeley community, all of whom had technical backgrounds.

Procedure. For each pHRI strategy, participants performed the familiarization task, followed by the three tasks, and then filled out our survey. They attempted each task twice with each strategy for robustness, and we recorded the attempt number for our analysis. Since we artificially set the true objective for participants to measure objective performance, we showed participants both the original and desired robot trajectory before interaction (Fig. 3), so that they understood the objective.

4.2 Results

Objective. We conducted a factorial repeated measures ANOVA with strategy (impedance or learning) and trial number (first attempt or second attempt) as factors, on total participant effort, interaction time, and cumulative true cost[2] (see Figure 4 and Figure 5). Learning resulted in significantly less interaction force ($F(1, 116) = 86.29, p < 0.0001$) and interaction time ($F(1, 116) = 75.52, p < 0.0001$), and significantly better task cost ($F(1, 116) = 21.85, p < 0.0001$). Interestingly, while trial number did not significantly affect participant's performance with either method, attempting the task a second time yielded a marginal improvement for the impedance strategy, but not for the learning strategy. This may suggest that it is easier to get used to the impedance strategy.

Overall, this supports H1, and aligns with the intuition that if humans are truly intentional actors, then using interaction forces as information about the robot's objective function enables robots to better complete their tasks with less human effort compared to traditional pHRI methods.

Subjective. Table 1 shows the results of our participant survey. We tested the reliability of our 4 scales, and found the understanding, effort, and collaboration scales to be reliable, so we grouped them each into a combined score. We ran a one-way repeated measures ANOVA on each resulting score. We found that the robot using our method was perceived as significantly ($p < 0.0001$) more understanding, less difficult to interact with, and more collaborative. However, we found no significant difference between our method and the baseline impedance method in terms of predictability.

[2]For simplicity, we only measured the value of the feature that needed to be modified in the task, and computed the absolute difference from the feature value of the optimal trajectory.

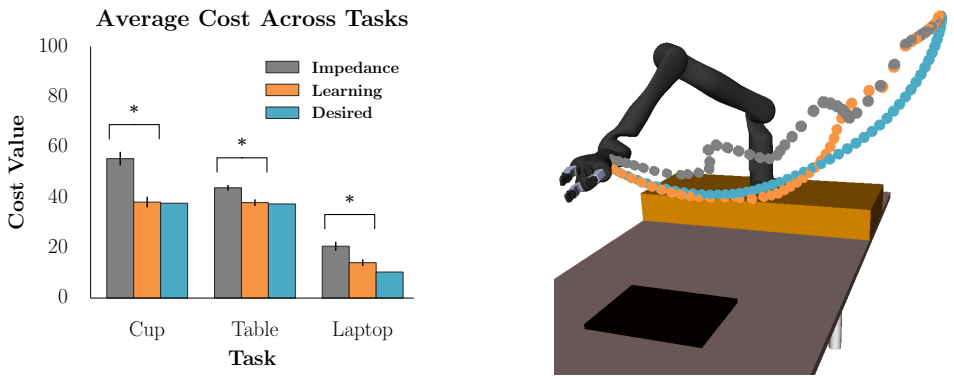


Figure 5: (left) Average cumulative cost for each task as compared to the desired total trajectory cost. An asterisk (*) means $p < 0.0001$. (right) Plot of sample participant data from laptop task: desired trajectory is in blue, trajectory with impedance condition is in gray, and learning condition trajectory is in orange.

Participant comments suggest that while the robot adapted quickly to their corrections when learning (e.g. "The robot seemed to quickly figure out what I cared about and kept doing it on its own"), determining what the robot was doing during learning was less apparent (e.g. "If I pushed it hard enough sometimes it would seem to fall into another mode and then do things correctly").

Therefore, H2 was partially supported: although our learning algorithm was not perceived as more predictable, participants believed that the robot understood their preferences more, took less effort to interact with, and was a more collaborative partner.

	Questions	Cronbach's α	Imped LSM	Learn LSM	F(1,9)	p-value
understanding	By the end, the robot understood how I wanted it to do the task.	0.94	1.70	5.10	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	1.25	5.10	85.25	<**.0001**
	The robot required minimal correction.					
predict	It was easy to anticipate how the robot will respond to my corrections.	0.8	4.90	4.70	0.06	0.82
	The robot's response to my corrections was surprising.	0.8	3.10	3.70	0.89	0.37
collab	The robot worked with me to complete the task.	0.98	1.80	4.80	55.86	<**.0001**
	The robot did not collaborate with me to complete the task.					

Table 1: Results of ANOVA on subjective metrics collected from a 7-point Likert-scale survey.

5 Discussion

Summary. We propose that robots should not treat human interaction forces as disturbances, but rather as *informative* actions. We show that this results in robots capable of *in-task* learning—robots that update their understanding of the task which they are performing and then complete it correctly, instead of relying on people to guide them until the task is done. We test this concept with participants who not only teach the robot to finish its task according to their preferences, but also subjectively appreciate the robot's learning.

Limitations and Future Work. Ours is merely a step in exploring learning robot objectives from pHRI. We opted for an approximation closest to the existing literature, but other possible better online solutions are possible. In our user study, we assumed knowledge of the two relevant reward features. In reality, reward functions will have larger feature sets and human interactions may only give information about a certain *subset* of relevant weights. The robot will thus need to disambiguate what the person is trying to correct, likely requiring active information gathering. Further, developing solutions that can handle dynamical aspects, like preferences about the timing of the motion, would require a different approach to inferring the intended human trajectory, or going back the space of policies altogether. Finally, while we focused on in-task learning, the question of how and when to generalize learned objectives to new task instances remains open.

Acknowledgments

*Andrea Bajcsy and Dylan P. Losey contributed equally to this work.
We would like to thank Kinova Robotics, who quickly and thoroughly responded to our hardware questions. This work was funded in part by an NSF CAREER, the Open Philanthropy Project, the Air Force Office of Scientific Research (AFOSR), and by the NSF GRFP-1450681.

References

[1] N. Hogan. Impedance control: An approach to manipulation; Part II—Implementation. *Journal of Dynamic Systems, Measurement, and Control*, 107(1):8–16, 1985.

[2] S. Haddadin, A. Albu-Schaffer, A. De Luca, and G. Hirzinger. Collision detection and reaction: A contribution to safe physical human-robot interaction. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 3356–3363. IEEE, 2008.

[3] N. Jarrassé, T. Charalambous, and E. Burdet. A framework to describe, analyze and generate interactive motor behaviors. *PLoS ONE*, 7(11):e49945, 2012.

[4] N. D. Ratliff, J. A. Bagnell, and M. A. Zinkevich. Maximum margin planning. In *Machine Learning (ICML), International Conference on*, pages 729–736. ACM, 2006.

[5] A. Jain, S. Sharma, T. Joachims, and A. Saxena. Learning preferences for manipulation tasks from online coactive feedback. *The International Journal of Robotics Research*, 34(10):1296–1313, 2015.

[6] S. Haddadin and E. Croft. Physical human–robot interaction. In *Springer Handbook of Robotics*, pages 1835–1874. Springer, 2016.

[7] A. Y. Ng and S. J. Russell. Algorithms for inverse reinforcement learning. In *Machine Learning (ICML), International Conference on*, pages 663–670. ACM, 2000.

[8] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438, 2008.

[9] D. Ramachandran and E. Amir. Bayesian inverse reinforcement learning. *Urbana*, 51(61801): 1–4, 2007.

[10] M. Kalakrishnan, P. Pastor, L. Righetti, and S. Schaal. Learning objective functions for manipulation. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 1331–1336. IEEE, 2013.

[11] M. Karlsson, A. Robertsson, and R. Johansson. Autonomous interpretation of demonstrations for modification of dynamical movement primitives. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 316–321. IEEE, 2017.

[12] A. D. Dragan and S. S. Srinivasa. A policy-blending formalism for shared control. *The International Journal of Robotics Research*, 32(7):790–805, 2013.

[13] S. Javdani, S. S. Srinivasa, and J. A. Bagnell. Shared autonomy via hindsight optimization. In *Robotics: Science and Systems (RSS)*, 2015.

[14] S. Pellegrinelli, H. Admoni, S. Javdani, and S. Srinivasa. Human-robot shared workspace collaboration via hindsight optimization. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 831–838. IEEE, 2016.

[15] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Machine Learning (ICML), International Conference on*. ACM, 2004.

[16] C. L. Baker, J. B. Tenenbaum, and R. R. Saxe. Goal inference as inverse planning. In *Proceedings of the Cognitive Science Society*, volume 29, 2007.

[17] M. L. Littman, A. R. Cassandra, and L. P. Kaelbling. Learning policies for partially observable environments: Scaling up. In *Machine Learning (ICML), International Conference on*, pages 362–370. ACM, 1995.

[18] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, and P. Abbeel. Motion planning with sequential convex optimization and convex collision checking. *The International Journal of Robotics Research*, 33(9):1251–1270, 2014.

[19] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi. An atlas of physical human–robot interaction. *Mechanism and Machine Theory*, 43(3):253–270, 2008.

[20] M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot modeling and control*, volume 3. Wiley: New York, 2006.

[21] A. D. Dragan, K. Muelling, J. A. Bagnell, and S. S. Srinivasa. Movement primitives via optimization. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 2339–2346. IEEE, 2015.

[22] L. Bottou. Online learning and stochastic approximations. In *On-line Learning in Neural Networks*, volume 17, pages 9–42. Cambridge Univ Press, 1998.