RESEARCH-ARTICLE

# Improving the Transparency of Robot Policies Using Demonstrations and Reward Communication

**MICHAEL S LEE**, Carnegie Mellon University, Pittsburgh, PA, United States

**REID G SIMMONS**, Carnegie Mellon University, Pittsburgh, PA, United States

**HENNY ADMONI**, Carnegie Mellon University, Pittsburgh, PA, United States



**Published:** 20 August 2025
**Online AM:** 10 June 2025
**Accepted:** 21 March 2025
**Revised:** 15 January 2025
**Received:** 22 May 2024

# Improving the Transparency of Robot Policies Using Demonstrations and Reward Communication

MICHAEL S. LEE, REID SIMMONS, and HENNY ADMONI, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA

Demonstrations are a powerful way to teach robot decision-making to humans. Although informative demonstrations may be selected a priori using the machine teaching framework, student learning may deviate from the pre-selected curriculum in situ. This article thus explores augmenting a curriculum of pre-selected demonstrations with a closed-loop teaching framework inspired by principles from the education literature, such as the zone of proximal development and the testing effect. We utilize tests accordingly to close the loop and maintain a novel particle filter model of human beliefs throughout the learning process, allowing us to provide demonstrations that are targeted at the human's current understanding in real time. A user study finds that our proposed closed-loop teaching framework reduces the regret (i.e., the suboptimality) of human test responses by 43% over an open-loop baseline. We also compare our closed-loop teaching framework against another baseline of directly communicating the robot's reward function in a second user study. We find that our closed-loop teaching outperforms direct reward communication by 64%, but we also observe synergies from the use of both teaching forms. Finally, we observe strong interaction effects between the teaching form and the domains considered in both user studies, seeing increased learning outcomes from well-designed demonstration-based teaching in the more challenging domain.

CCS Concepts: • **Computer systems organization** → *Robotics*; • **Mathematics of computing** → *Computing most probable explanation*; • **Human-centered computing** → *User models*; *Empirical studies in HCI*;

Additional Key Words and Phrases: Explainable AI, Transparency, Human-Robot Interaction, Machine Teaching, Inverse Reinforcement Learning

**ACM Reference format:**
Michael S. Lee, Reid Simmons, and Henny Admoni. 2025. Improving the Transparency of Robot Policies Using Demonstrations and Reward Communication. *ACM Trans. Hum.-Robot Interact.* 14, 4, Article 72 (August 2025), 31 pages.
https://doi.org/10.1145/3743150

## 1 Introduction

Much progress has been made in obtaining complex and capable robot policies through **Reinforcement Learning (RL)** (e.g., [2]). Ensuring the transparency (i.e., understandability and predictability [15]) of these policies in all scenarios is key to calibrating the expectations of developers and end-users toward proper usage; however, this remains a challenge [64].

Authors' Contact Information: Michael S. Lee (corresponding author), Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA; e-mail: ml5@alumni.cmu.edu; Reid Simmons, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA; e-mail: rsimmons@andrew.cmu.edu; Henny Admoni, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA; e-mail: henny@cmu.edu.


ACM 2573-9522/2025/8-ART72
https://doi.org/10.1145/3743150

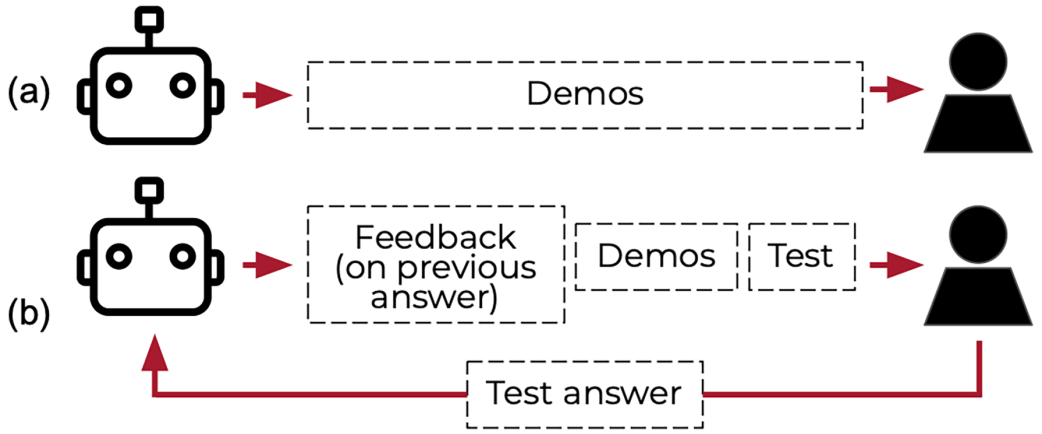


Fig. 1. (a) Previous works aim to improve robot policy transparency via a set of demonstrations selected *a priori*, but student learning may deviate from the expected trajectory. (b) We propose a closed-loop teaching framework using tests and feedback to detect and correct for such deviations *in situ*.

One effective way to increase policy transparency is through demonstrations of the policy, which can be selected through a *machine teaching* [69] paradigm that selects the minimal set of examples (e.g., demonstrations) that will help a student comprehend a concept (e.g., a policy) given their learning model (e.g., **Inverse Reinforcement Learning (IRL)**). Although machine teaching can help select a principled curriculum of demonstrations *a priori*, student learning can deviate from the modeled learning trajectory *in situ*. In our previous work [35], machine teaching-selected demonstrations improved human performance on post hoc tests assessing later-demonstrated concepts but decreased performance on post hoc tests assessing early-demonstrated concepts, suggesting perhaps that the curriculum moved too quickly past the early concepts without in situ testing to provide additional instruction as necessary.

Thus, *our key idea is to complement a curriculum of machine teaching-selected demonstrations with a closed-loop teaching framework inspired by the education literature to provide tailored instruction in real time* (Figure 1). A guiding educational concept is teaching in the ***Zone of Proximal Development* (ZPD)** or "Goldilocks zone" [21, 63], which suggests that the examples provided to the learner should not be too easy nor too difficult, given their current understanding. However, the ZPD often changes at different rates for different students according to their personal learning rate, which must be periodically assessed by testing. We inform the testing cadence with the educational concept of the *testing effect* [56], which predicts an increase in learning outcomes when a portion of the teaching budget is devoted to testing the student (using testing not only as a tool for assessment but also for teaching). And by incorporating tests and feedback in a closed teaching loop, we maintain an up-to-date model of human beliefs and promote demonstrations that are provided at the right level of difficulty in situ.

To illustrate the utility of our closed-loop teaching framework, consider a robot that increases the transparency of its reward function and subsequent policy to a human using demonstrations, tests, and feedback (Figure 2). The robot's objective is to deliver a package to the destination, whose reward function balances traveling through difficult terrain, such as mud, and reducing the number of actions it takes. To convey its reward function, the robot first provides a human with the demonstration in Figure 2(a). Because the robot takes a two-action detour to avoid the mud instead of going through it, the human may infer that the robot associates mud with a negative reward.

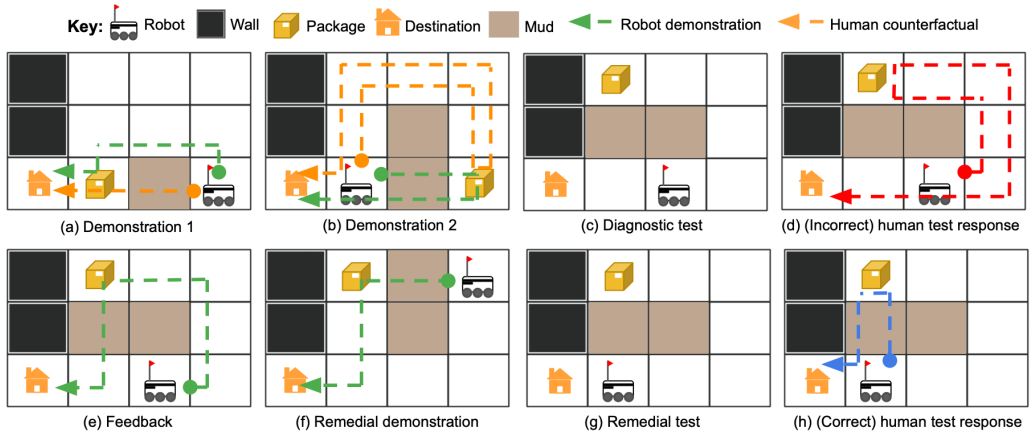


Fig. 2. Sample teaching sequence for a batch of knowledge components (KCs) on mud cost. (a) First demonstration (green) contrasts with a counterfactual alternative likely considered by a human (orange), which conveys that mud is costly. (b) Second demonstration lower-bounds mud cost. (c) Human is asked to predict the robot's behavior in a test. (d) Incorrect response suggests that the demonstration was not understood. (e) Human is given the correct response as feedback. (f) Remedial demonstration is provided to target the misunderstanding. (g) Human is given a remedial test. (h) Correct answer suggests understanding.

The robot considers what to demonstrate next to convey more information regarding its reward function. Importantly, it knows that the human is likely to consider mud as costly from the first demonstration, but does not know *how* costly. For example, the human may counterfactually believe that the robot would take a four-action detour when faced with two mud patches (Figure 2(b)). However, the robot knows that its ratio of reward for mud to action is $-3$ to $-1$ and that, consequently, it would simply go through the mud in Figure 2(b) to maximize its reward. Seeing how its direct path meaningfully differs from the human's likely detouring counterfactual (i.e., an alternative, potentially suboptimal behavior), the robot considers this to be an informative next demonstration to provide that targets the human's ZPD—providing a meaningful yet incremental update to the human belief through an additional unit of information that upper-bounds the cost of mud.

The robot then follows the two demonstrations with a diagnostic test that simultaneously challenges the human to apply their learned knowledge and reveals whether the robot's current model of the human's beliefs needs to be corrected (Figure 2(c)). If the human answers incorrectly (Figure 2(d)), the robot may provide feedback, a remedial demonstration, then a sequence of remedial tests and feedback until the human demonstrates concept mastery, inspired by the testing effect (Figure 2(e)–(h)). Importantly, the robot continues to update its model of the human's beliefs according to test answers and throughout the remedial interactions to consider the right counterfactuals when estimating the informativeness of future demonstrations. The above teaching sequence demonstrates the importance of maintaining a calibrated model of the human's beliefs through closed-loop testing, which can help select demonstrations that are within the human's ZPD in situ.

And while this article focuses primarily on teaching robot decision-making through demonstrations, teaching can take other forms, e.g., directly conveying weights of reward features [57], saliency maps highlighting where the agent is attending to [19], and reward decomposition bars that group future rewards into semantically meaningful categories [6]. Interestingly, Sanneman and Shah [57] found that directly communicating weights of reward features performed the best

objectively and subjectively in their two domains compared to HIGHLIGHTS (a teaching form that communicates an agent's reward function via demonstrations from states with maximal difference between the Q-values for the best and worst actions [3]). We wondered whether direct reward communication would also outperform our closed-loop teaching method in our domains, and also whether there would be synergy in conveying both. We thus ran a second online user study exploring whether direct reward communication could improve the transparency of robot policies in the grid world domains considered in this article.

Our contributions are thus as follows. First, a closed-loop teaching framework that provides demonstrations, tests, and feedback based on insights from the education literature. Second, a particle filter-model of human beliefs that supports iterative updates and a calibrated prediction of the counterfactuals likely considered by the human for each demonstration that could be provided. Third, a user study that finds that our framework reduces the regret of human test responses by 43% over a baseline. And fourth, a second user study that compares our closed-loop teaching framework against directly communicating the robot's reward function, finding that closed-loop teaching outperforms direct reward communication alone by 64% but also observing synergies from leveraging both teaching forms. We observe a strong interaction effect in both user studies, seeing increased learning outcomes from well-designed demonstration-based teaching in more challenging domains. This article builds on our prior work [34–36] and is derived from Chapter 6 of the first author's PhD thesis [33]; we include shared content from the aforementioned sources.

## 2 Related Work

### 2.1 RL and IRL

RL is a framework for learning a policy (i.e., a behavior) that maximizes a given reward function. Classical RL methods such as Q-Learning and SARSA have traditionally been limited to low-dimensional state and action spaces [62]. Recent advances in deep learning have enabled deep RL algorithms such as Deep Q Networks [44], Soft Actor-Critic [20], and Proximal Policy Optimization [58] to scale to high-dimensional domains, including Atari games [44], Go [61], and robot control for manipulation and locomotion [2, 54]. Despite these advancements, a key challenge remains: the resulting policies are often opaque and difficult for humans to understand.

IRL, on the contrary, focuses on inferring the reward function that underlies a policy from observed demonstrations. This framework was introduced by Ng and Russell [45], who provided an approach for extracting constraints on the reward function from demonstrations. Subsequent advancements, such Bayesian IRL [55], maximum entropy IRL [70], deep maximum entropy IRL [65], and adversarial IRL [17], improved robustness and scalability. In this work, we build on previous efforts that use IRL to model human learning from demonstrations [25, 32, 34, 35], to help humans better understand the underlying reward functions of agents and robots (these two terms are interchangeable for the purposes of this work), and their subsequent policies.

### 2.2 Explainable RL

The field of explainable RL focuses on helping humans understand the decision-making of RL agents. Recent surveys [43, 51, 64] highlight a variety of approaches, such as approximating a black-box RL policy via an interpretable model (e.g., a decision tree [60]), using saliency maps to highlight features of a state used for decision-making [19], visualizing minimally different counterfactual states that would have yielded a different action [46], and identification of critical training points (e.g., for estimating Q-values [18]). The most recent survey by Milani et al. [43] divides the work in this field into three categories of methods: *feature importance* methods that highlight the features that influenced the agent's decision-making, *learning process and* ***Markov Decision Process (MDP)***

methods that highlight relevant past experiences or MDP components that lead to the agent's current action, and *policy-level* methods that convey the agent's general long-term behavior.

Explainable RL methods can also differ in the modality used to communicate information, e.g. demonstrations (see works in Section 2.3), natural language [13, 14], direct numerical values (e.g., of reward weights [57], decomposition of action Q-values into semantically meaningful reward types [27]). In this article, we contribute a *policy-level* method that conveys an understanding of an agent's overall behavior to a human through representative demonstrations.

### 2.3 Policy Summarization

Policy summarization aims to provide a global understanding of a policy to a human through example state-action pairs [4], which can aid in transparency. One approach relies on heuristics such as communicating states with a large difference between the best and the worst (or average) Q-values [3, 24], or communicating an agent's second-best trajectory as a counterfactual [5]. We instead build on the second approach based on machine teaching [69], which we highlight below.

Our previous works model human learning from robot demonstrations as resembling IRL and leverage human teaching techniques such as scaffolding [34] and principles from cognitive science such as counterfactual reasoning [35] to provide demonstrations that incrementally provide information on the robot's underlying reward function. However, these methods model the human learner as using exact IRL [45], which is unable to gracefully handle conflicting information (e.g., knowledge assumed to be learned but failed to be demonstrated during testing). Furthermore, they utilize tests for assessment only after having provided demonstrations. We build on this line of work by proposing a Bayesian model of human beliefs in the form of a particle filter and also utilizing intermittent testing to simultaneously maintain an up-to-date model of human beliefs and provide targeted instruction.

Huang et al. [25] also use Bayesian IRL [55] to model human learning from robot demonstrations, but only update the relative probabilities of a static set of reward beliefs with each additional demonstration. We instead allow for resampling [40] of the beliefs within our particle filter to more efficiently approximate the posterior distribution of human beliefs. Furthermore, Huang et al. note an equivalence of Bayesian IRL and maximum entropy IRL under select noise models and explore a variety of exponential likelihood functions for updating a model of human beliefs given an observed robot demonstration, e.g. based on the reward, trajectory, or strategy difference between the observed and expected behavior. While our proposed likelihood function is also based on the difference between the reward of the robot's observed and expected behavior, our formalism more flexibly allows each observed robot demonstration to provide multiple updates to the model of human beliefs based on the number of counterfactual trajectories that the human may consider.

Finally, a line of work by Qian and Unhelkar also explores interactive policy summarization. In [53], they allow humans to request specific demonstrations from an agent and find that a hybrid strategy of agent-selected and human-selected demonstrations yields the best objective and subjective results. Our proposed approach for modeling human beliefs and subsequently selecting informative demonstrations could provide the agent-selected demonstrations in their framework. In close proximity to our work, they also propose personalized policy summarization [52], a method that also utilizes intermittent testing to maintain a model of human beliefs and provide tailored demonstrations. But while personalized policy summarization provides tests in predetermined batch sizes only for assessment, we are inspired by the testing effect to also utilize tests for teaching. When a misunderstanding is identified by a test, we continue to provide tests with corresponding feedback on each received answer in a tight loop until the observed misunderstanding is remedied.

## 3 Technical Background

This section provides the background for selecting informative demonstrations for a (human) learner using IRL-like reasoning to infer a reward function underlying demonstrations. We first introduce the MDP, a common framework for formalizing RL problems and the policies derived from them.

*MDP.* The robot models its world as an instance (indexed by $i$) of an MDP, $MDP_i$, composed of sets of states $\mathcal{S}_i$ and actions $\mathcal{A}$, a transition function $T_i$, a reward function $R$, a discount factor $\gamma$, and the initial state distribution $S_i^0$. We refer to a group of related MDP instances as a *domain* (described below) and $\mathcal{S} : \bigcup_i \mathcal{S}_i$ is the union over all their states. An optimal trajectory $\xi^*$ is a sequence of $(s_i, a, s_i')$ tuples that follow the optimal policy of the robot $\pi_i^*$. In line with prior work [1], reward $R$ is represented as a weighted linear combination of reward features $\phi$: $R = \mathbf{w}^{*\top}\phi(s, a, s')$. Finally, we assume that the human is aware of the full MDP apart from weights $\mathbf{w}^*$.

A domain is a group of MDPs that share $R, \mathcal{A}$, and $\gamma$ but differ in $T_i, \mathcal{S}_i$, and $S_i^0$. For example, all MDPs in the delivery domain share the same $R$, even though they may contain different mud patches (Figure 2(a) and (b)). Thus through IRL, all demonstrations within a domain will support inference over a common $\mathbf{w}^*$. We simplify the notation such that $\pi^*$ refers to any optimal policy within a domain, and $\xi^*$ refers to a demonstration (dropping the corresponding MDP).

*Machine Teaching for Policies.* Our objective in selecting an informative curriculum of demonstrations to convey $\pi^*$ is captured by the machine teaching framework for policies [32]. We aim to select a set of demonstrations that helps a human, who is assumed to use IRL-like reasoning [26], approximate $\mathbf{w}^*$, and then perhaps use planning [59] to recover $\pi^*$. Thus, the objective reduces to selecting demonstrations that are informative in conveying $\mathbf{w}^*$, which can be measured using **Behavior Equivalence Classes (BECs)**.

*BEC.* The *BEC* of a demonstration is the set of reward functions under which the demonstration is still optimal.

For a reward function that is a weighted linear combination of features, the BEC of a demonstration $\xi^*$ of $\pi^*$ is defined as the half-space [35] formed by the exact IRL equation [45]

$$\mathrm{BEC}(\xi^*|\pi^*, \pi_{\mathbf{w}}) := \mathbf{w}^{*\top}\left(\mu^s_{\pi^*} - \mu^s_{\pi_{\mathbf{w}}}\right) \geq 0, s = \xi^*(0), \tag{1}$$

where $\mu^s_\pi = \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t\phi\left(s_t\right) \mid \pi, s_0 = s\right]$ is the vector of reward feature counts accrued from starting in $s$ and following $\pi$ after ($\pi_{\mathbf{w}}$ is the optimal policy under reward weight $\mathbf{w}$) and $\xi^*(0)$ is the first state of $\xi^*$. Any demonstration can be converted into a constraint on $\mathbf{w}^*$ using Equation (1) and a candidate belief $\mathbf{w}$. Importantly, each constraint can be considered a ***Knowledge Component*** **(KC)** [30] that captures a characteristic of the reward function (e.g., a tradeoff between the underlying reward feature weights).

Consider again the delivery domain, which has binary reward features $\phi$ = [*traversed mud*, *battery recharged*, *action taken*], $\mathbf{w}^* \propto [-3, 3.5, -1]$.[1] We assume that the human begins with a prior that the weight of the "action taken" feature is negative (e.g., a bias toward the shortest path, Figure 3(a)). The demonstration in Figure 3(b) yields the constraint (or KC) in Figure 3(c), which indicates that $w_0^* \leq 2w_2^*$ (i.e., mud is at least twice as costly as an action), since two actions were taken to detour around the mud rather than counterfactually going through it (the optimal trajectory for a candidate belief that considers mud to be slightly negative, neutral, or slightly positive).

[1]In practice, we require $||\mathbf{w}^*||_2 = 1$ to bypass both the scale invariance of IRL and the degenerate all-zero reward function without loss of generality.

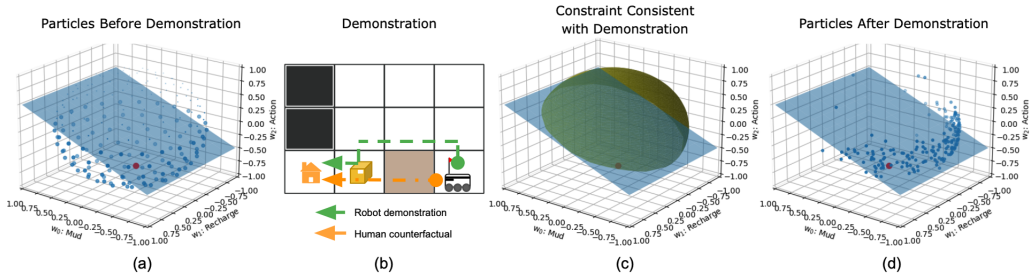


Fig. 3. Example sequence on how a demonstration updates a particle filter model of human beliefs. The robot reward function is shown as a red dot, and the constraint consistent with the demonstration is shown in all plots for reference. (a) Particles before demonstration (prior). (b) Demonstration shown to the human, alongside a counterfactual that considers mud to be slightly negative, slightly positive, or neutral. (c) The constraint (Equation (1)) consistent with the demonstration that conveys that mud must be at least twice as costly as an action, visualized with the uniform distribution portion of the custom distribution (Figure 4) used to update particle weights. (d) Particles after demonstration (posterior).

## 4 Methods

The example of the delivery robot in Section 1 highlights the importance of maintaining an up-to-date model of human beliefs and likely counterfactuals when selecting a demonstration. In this section, we propose a particle filter-based model of human beliefs amenable to iterative Bayesian updates and sampling for counterfactual reasoning, where each particle represents a potential human belief regarding the robot's reward function. We then leverage this model in a closed-loop teaching framework that uses insights from the education literature to select demonstrations that target gaps in human understanding identified through testing.

### 4.1 Particle Filter Human Model

A particle filter is a sequential Monte Carlo method that can flexibly model the progression of arbitrary posterior distributions (e.g., non-Gaussian, multimodal) given new observations and a likelihood function [12]. Given its feasibility for the domains considered in this article and its prior use in modeling various human states and behaviors, such as body tracking [7, 10], sentence comprehension [38], and bandit-like gameplay [66], we model the human's beliefs over a robot's reward function as a set of particles, defined by their positions and associated weights $\{\mathbf{x}_t, \check{\mathbf{w}}_t\}$. Each particle represents a possible reward function that the human could believe the robot to have, and the associated particle weight captures the strength of that belief.

*4.1.1 Updating Particle Positions and Weights.* Without loss of generality, assume that a demonstration or test response is provided at each timestep $t$. Each demonstration generates multiple constraints by comparing the demonstration against possible counterfactual trajectories, and each incorrectly answered test will generate a single constraint by comparing the true test answer against the incorrect answer, both through Equation (1). Each constraint generated via a demonstration or a test response is a half-space constraint, with one side being *consistent* with the demonstration or test response and the other side being *inconsistent*.

Each constraint $y_t$ can then be translated into a probability distribution $p(\mathbf{x_t}|y_t)$ that can be used to update the weights of each particle (Figure 3). We propose a custom probability distribution $p(\mathbf{x_t}|y_t)$ that translates each constraint into a combination of a uniform distribution that aligns with the consistent half-space of the constraint and the von Mises–Fisher distribution (a generalization of the Gaussian distribution on a sphere [11]) whose mean direction aligns with the inconsistent

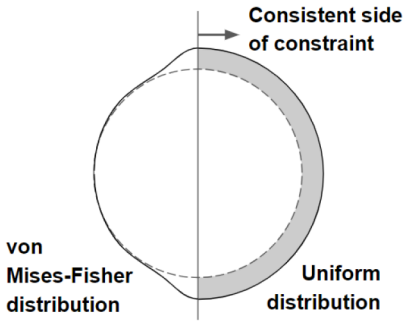


Fig. 4. A cross-section of the spherical pdf used to update particle weights given a constraint generated from a demonstration (Equation (1)).

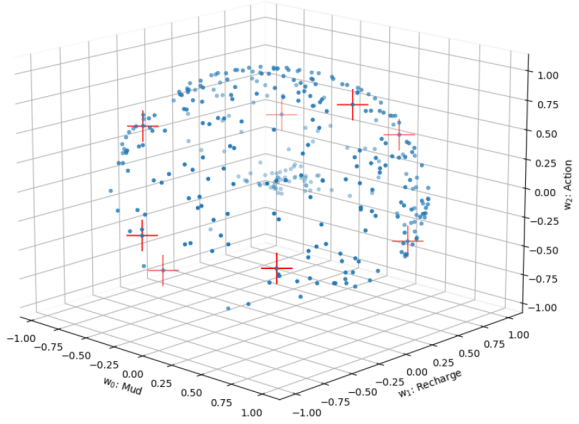


Fig. 5. Human counterfactuals are generated by sampling beliefs from the particle filter model. As nearby particles are likely to generate similar counterfactuals, we rely on the 2-approximation algorithm for the k-center problem to sample $k$ beliefs (marked by red crosses) that are spread out.

half-space (Figure 4). The uniform distribution asserts that any particle lying on the consistent half-space is equally valid for that demonstration, whereas the von Mises–Fisher distribution asserts that a particle is exponentially less likely to have generated that demonstration as you move away from the consistent side of the constraint. Please find the **Probability Density Function (pdf)** of the custom distribution in Appendix A.1 and the routine for updating the particle filter given new demonstrations or tests in Algorithm 1. And to maintain the conciseness of the main script, please find practical tips on how to resample the particle filter to combat sample degeneracy and impoverishment (line 13 of Algorithm 1), as well as how to reset the particle filter if it receives heavily conflicting information (line 8 of Algorithm 1) in Appendix A.2.

*4.1.2 Sampling Human Beliefs.* Given a running particle filter model, we may sample human beliefs in order to do counterfactual reasoning over how the human may interpret each demonstration that could be shown. We first run systematic resampling [39] on a copy of the particles to downselect to a candidate set, favoring those that are higher weighted. We then rely on the 2-approximation algorithm [23] to greedily select $k$ distributed samples such that the maximum distance from any particle in the candidate set to one of the $k$ samples is minimized (Figure 5). The algorithm iteratively picks the particle with the largest distance to the already selected samples as

**Algorithm 1:** Particle Filter for Modeling Human Beliefs

Initialize particles $x_0^{(i)} \sim p(x_0)$ for $i = 1, \ldots, N$
**for** $t = 1, \ldots, T$ **do**
    // Update filter given new demonstration or test at $t$
    **for** $i = 1, \ldots, N$ **do**
        Compute weight $\check{w}_t^{(i)} = \check{w}_{t-1}^{(i)} \cdot p(\mathbf{x_t}^{(i)} | y_t)$
    **end for**
    **if** $\sum_{j=1}^{N} \check{w}_t^{(j)} < \check{w}_{threshold}$ **then**
        Perform a particle filter reset
    **end if**
    Normalize weights $\tilde{w}_t^{(i)} = \frac{\check{w}_t^{(i)}}{\sum_{j=1}^{N} \check{w}_t^{(j)}}$
    Compute effective sample size $n_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (\tilde{w}_t^{(i)})^2}$
    **if** $n_{\text{eff}} < N_{\text{threshold}}$ **then**
        Resample $x_t^{(i)}$ with probabilities $\tilde{w}_t^{(i)}$ using KLD resampling
    **end if**
**end for**

the next sample; this heuristic ensures that the maximum distance from any particle to any of the selected samples is never worse than twice the optimal solution. As nearby particles are likely to generate similar counterfactuals, we sample beliefs that are approximately spread out.

For our experiments, we set $k$ to 25. To support real-time counterfactual reasoning, we also sampled 2500 beliefs from the surface of the 2-sphere (the space of possible human beliefs regarding the robot's reward function in our domains) for which we pre-computed the optimal policies. Each particle in the particle filter was then mapped to the closest precomputed belief during experiments toward efficient selection of additional demonstrations and tests.

### 4.2 Closed-Loop Teaching

With a particle-filter model of human beliefs that is amenable to iterative updates, we now formulate a closed-loop teaching framework for conveying a robot's reward function to a human using demonstrations and tests. As we walk through the framework conceptualized in Figure 6, we highlight the principles from the education literature that guide the design. A sample rollout of a teaching sequence is shown in Figure 2, which serves as a visual correspondence to the algorithmic characterization of the framework provided in Algorithm 2.

We first leverage feature and counterfactual scaffolding from our prior work [35] to select KCs (see Equation (1)) that incrementally increase in information across an increasing subset of features (e.g., mud vs. action cost, recharging vs action cost, then tradeoffs between all three). This set of KCs guides the machine teaching selection of the *curriculum* of demonstrations that can be used to teach the robot reward function to a human.

We begin the teaching loop by taking a single batch of related KCs that define a *lesson* (e.g., bounds on mud cost) and providing it to the `demonstrator` (Figure 6) to select demonstrations from the curriculum that convey these KCs. Specifically, we utilize counterfactual reasoning [35] to select demonstrations that are informative with respect to the counterfactuals likely considered by the human. We simultaneously leverage the educational principles of the *ZPD* [63] to provide a sequence of demonstrations that provide information incrementally, i.e., demonstrations that convey one new constraint at a time (such as first providing a lower-bound on the mud cost, then later an upper-bound). And when given a choice between two equally informative demonstrations

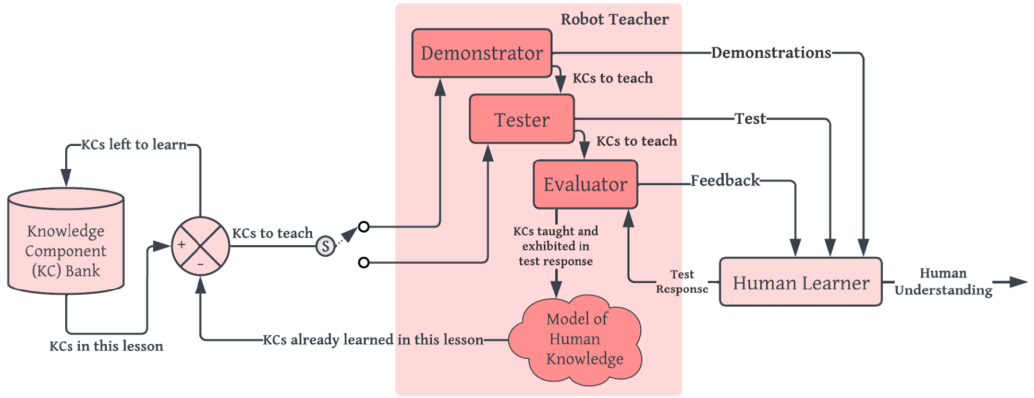


Fig. 6. Proposed closed-loop teaching framework. Knowledge components (KCs) are passed to the robot teacher as a lesson. The `demonstrator` generates demonstrations that convey the KCs, the `tester` provides test(s), and the `evaluator` analyzes the test response(s), provides feedback on its correctness, and updates the model of human knowledge. If the human fails to learn a KC through two rounds of demonstrations and tests, the switch (labeled "S") flips such that only tests and feedback are provided until an understanding of the remaining KCs is demonstrated through correct responses.

**Algorithm 2:** Closed-Loop Teaching Framework

```
Group related knowledge components (KC) into batches using counterfactual scaffolding
for each batch of KCs (i.e. lesson) do
    Provide initial demonstrations and diagnostic tests
    Evaluate diagnostic test responses
    if diagnostic test responses are incorrect then
        Provide corrective feedback, remedial demo, and a remedial test
        Evaluate remedial test response
        while remedial test response is incorrect do
            Provide corrective feedback and provide new remedial test
            Evaluate remedial test response
        end while
    end if
end for
```

that could be shown next to convey the desired KC, we optimize for visual similarity and visual simplicity as suggested by our previous work [34], selecting the one that looks most similar to the previously shown demonstration (e.g., location of mud patches) and has the fewest visual clutter (e.g., number of mud patches).

After the demonstrations have been provided, the `tester` selects *diagnostic tests* that will verify whether the human has learned the KCs in the lesson. These diagnostic tests optimize for visual dissimilarity to the teaching demonstrations and visual complexity (i.e., increasing distracting visual clutter) [35] to challenge the learner.

For each diagnostic test answered incorrectly, the `evaluator` will provide immediate *feedback* to the learner, highlighting how their answer differs from the correct one. This approach is inspired by research indicating that immediate feedback on errors improves learning outcomes [29]. In addition, a remedial demonstration that visually simplifies [34] the missed KC will be provided to reinforce the concept being taught, along with a remedial test featuring greater visual complexity to challenge the learner in demonstrating the missed KC. The selection of the remedial demonstration

or test is achieved through greedy sequential optimization, focusing on minimizing the distance between the constraint of the missed KC and a constraint conveyed by a candidate demonstration or test: we first minimize the number of mismatched feature counts between the two constraints, then minimize the Manhattan distance between the constraints. It is important to note that the missed KC is determined by comparing the human's test answer with the optimal test answer; while it may not correspond directly to one of the KCs originally included in the lesson, it best addresses the learner's current misunderstanding.

If the human also gets the remedial test wrong, the switch in Figure 6 (labeled "S") flips, and the `tester` and `evaluator` will continue to provide only visually dissimilar and complex remedial tests with corresponding feedback (but no additional demonstrations) until the human shows understanding of each iteration's missed KC. This is motivated by the testing effect [56], which supports the use of tests not only for assessment but also for teaching and increasing learning outcomes. Note that for each demonstration provided or test response received throughout this learning process, we update the particle filter model of the human's beliefs. And we utilize the particle filter model to consider the counterfactuals the human is likely to consider for each potential remedial demonstration or remedial test in order to select the one that will best convey or test the missed KC for the human. Once all of the missed KCs for this lesson have been demonstrated via correct remedial test responses, a fresh batch of KCs (i.e., a new lesson) is pulled from the KC bank and the switch flips upward to provide demonstrations again.

Alternatively, if all diagnostic tests in this lesson had been correctly answered initially, a fresh batch of KCs would have been pulled from the KC bank to begin the next lesson directly without remedial instruction.

When all lessons have been taught, the human's subsequent knowledge can be evaluated on a held-out set of tests in which they predict the robot's policy in previously unseen environments.

## 5 User Study on Open-Loop vs. Closed-Loop Teaching

We conducted an online user study[2] exploring whether our proposed closed-loop teaching method improves the transparency of a robot's policy to a human. The study involved participants learning about the robot policy in two domains through a combination of demonstrations, tests, and feedback and predicting the robot's behavior in new test environments.

### 5.1 Study Design

We followed a mixed study design. The between-subjects variable was *feedback loop* with the following three conditions. *Open* feedback loop followed our prior work [35] in utilizing counterfactual reasoning to select a set of informative demonstrations a priori that monotonically decreased in cumulative BEC area (i.e., a model of human beliefs), one KC at a time. *Partial* feedback loop additionally provided a diagnostic test after each lesson and provided feedback as necessary, while the *full* feedback loop additionally provided a remedial demonstration and remedial tests until the KC in question was correctly applied in a remedial test. For a fair comparison, each condition showed the same median number of demonstrations and tests (11 for delivery and 22 for skateboard).

The within-subject variable was *domain*, which consisted of the following two conditions. In the *delivery* domain, the robot is penalized for moving out of mud and rewarded for recharging. In the *skateboard* domain, the robot is rewarded each time it moves with the skateboard (e.g., riding is efficient) or traverses through a designated path (Figure 7). Thus, each domain consists of two

[2]Code for the methods, domains, and relevant hyper-parameters used in this study can be found at https://github.com/SUCCESS-MURI/closed_loop_teaching_study.

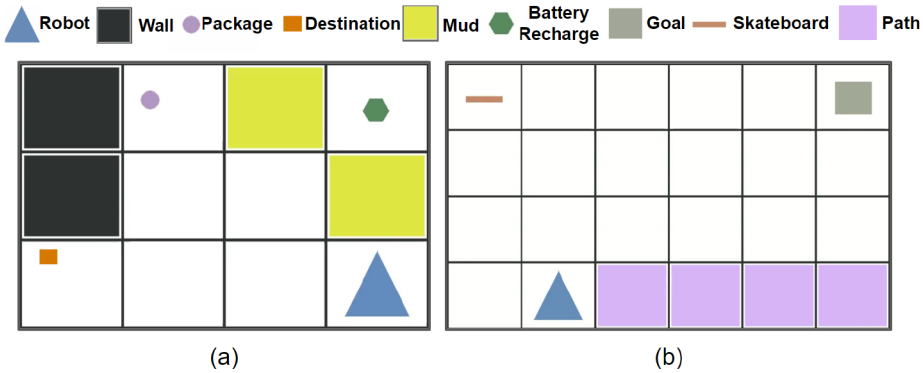


Fig. 7. Two domains designed for user study, (a) delivery, (b) skateboard. The semantics of the objects were hidden using arbitrary shapes and colors.

unique reward features and one shared feature that penalizes each action. The *skateboard* domain was designed to be more challenging than the *delivery* domain, as the value of the skateboard depends both on the distance to the skateboard and subsequent distance to the goal. Considering the possibility of interaction effects between the between-subjects variable of feedback loop and domains of varying difficulties, we subsequently also provide hypotheses that relate to domain difficulty at the end of this section. The order of the domains shown to the user was counterbalanced in the study.

The user study consisted of two trials, with each trial comprising a teaching portion and a testing portion in one domain. During teaching, participants were first explicitly informed of the reward features of the domain. Then, they inferred the corresponding reward weights by watching demonstrations and perhaps undergoing diagnostic tests, corrective feedback, and additional remedial instruction depending on their assigned feedback loop condition. For every interaction, participants indicated whether it improved their understanding of the robot's policy via a Likert scale. At the end of the teaching session, participants were asked to rate their level of focused attention, the perceived usability of their assigned teaching condition, and their understanding of the robot's policy via Likert scales. During testing, participants were tasked with predicting the robot's optimal trajectory in six unseen test environments in random order, which were selected according to prior work [35] to comprise two low, medium, and high difficulty environments each.

We tested the following hypotheses (H1–H4) using the measures (M1–M4) below. The Likert scales corresponding to M2 and M4 were provided after the teaching portion but before the testing portion, and Likert scales corresponding to M3 were provided after each demonstration and test in the teaching portion.

*H1*: (a) The test responses will be best for *full* feedback loop, then *partial*, then *open*. (b) *Delivery* will result in better test responses over *skateboard*.

*H2*: (a) Focused attention and perceived usability will be highest for *full* feedback loop, then *partial*, then *open*. (b) *Delivery* will result in higher focused attention and perceived usability over *skateboard*.

*H3*: (a) Improvement ratings will be highest for *full* feedback loop, then *partial*, then *open*. (b) *Delivery* will result in higher improvement ratings over *skateboard*.

*H4*: (a) Understanding ratings will be highest for *full* feedback loop, then *partial*, then *open*. (b) *Delivery* will result in higher understanding ratings over *skateboard*.

*M1. Test Response*: The reward of the human's test response, measuring the human's ability to predict the robot's policy.

*M2. Focused Attention and Perceived Usability*: We adapted the User Engagement Scale short form [48] to ask six questions targeting focused attention and perceived usability, each answered with a 5-point Likert scale. Please find the corresponding questions in Appendix A.3.

*M3. Improvement*: "Did this interaction improve your understanding of the game strategy [i.e. robot policy]?", answered with a 5-point Likert scale.

*M4. Understanding*: "Do you feel that you now understand the game strategy?", answered with a 5-point Likert scale.

### 5.2 Results

We collected data from 206 participants using Prolific [49]. The participants were approximately 70% male, 28% female, 1% non-binary, and 1% preferred not to disclose, and the ages ranged from 18 to 67 ($M = 32.49$, SD = 11.15). The recruitment process and study were approved by the Carnegie Mellon University Institutional Review Board. In the *full* feedback loop condition, we removed data from one participant who did not miss any diagnostic tests during teaching (thus did not see any remedial instruction in either domain) and an outlier participant whose total number of interactions exceeded 3 standard deviations of the mean number of interactions in this condition (since repeated failures of similar remedial tests suggested lack of attention). This left 68 participants in each between-subjects condition.

We present the results below with the caveat that two bugs in the user study code were discovered post hoc. First, only the positions of the particles, and not their weights, were considered when sampling human beliefs from the particle filter (Section 4.1.2). Second, remedial demonstrations and remedial tests that did not minimize the distance to a missed KC were sporadically selected (Section 4.2). Correcting both bugs post hoc reveals that while approximately 2.59% of the interactions in the *full* feedback loop condition could have been different, the vast majority of interactions in this condition would have remained unchanged. Furthermore, since these bugs sporadically produced off-target remedial instruction, we hypothesize that these results represent a lower bound on the efficacy of the *full* feedback loop condition.

*H1*: We considered analyzing test responses in two ways: binary scores measuring the optimality of human test responses, and regret measuring the degree of suboptimality of human test responses (i.e., the difference between rewards of human and optimal test responses). The former analysis was coarse and did not yield any significant results, so we opted for the latter, which provides a finer resolution. We also considered normalizing the regret by the optimal test response reward but decided against it to prevent identical mistakes from being penalized differently based on different trajectory lengths and optimal rewards (please find further elaboration in Section 5.3). A two-way mixed ANOVA indicated a significant effect of feedback loop on regret ($F(2, 201) = 3.65, p = 0.028$).[3] Tukey analyses revealed that *full* ($M = 0.24$) had 43% lower regret over *open* ($M = 0.42$, $p = 0.027$), with *partial* sitting in between with no significant difference to either ($M = 0.29$, Figure 8(a)). The ANOVA also indicated a significant effect of domain on regret ($F(1, 201) = 50.75, p \leq .001$), where a $t$-test revealed a significant difference between the regret between *delivery* ($M = 0.18$) and *skateboard* ($M = 0.45$), $t(406) = -5.792, p < 0.001$.

The ANOVA also indicated an interaction effect ($F(2, 201) = 3.45, p = 0.03$) between feedback loop and domain. In the *skateboard* domain, Tukey analyses revealed that *full* ($M = 0.33$) had significantly lower regret over *open* ($M = 0.62$, $p = 0.014$),

[3] Although one participant had only 11/12 test responses recorded, we note that this does not significantly impact the reported results as responses were averaged for each participant and 2,447 total test responses were recorded.

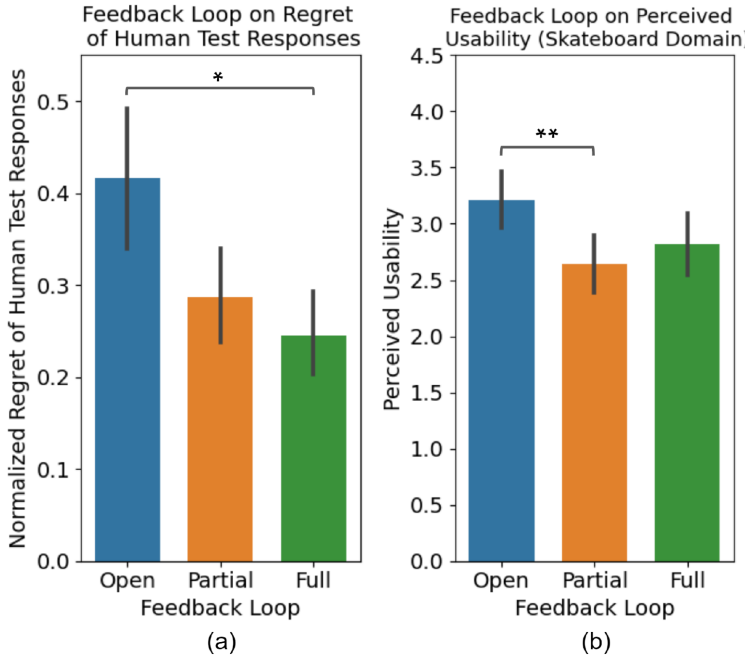


Fig. 8. (a) Full closed-loop teaching yields lower regret for human tests responses than *open* across domains (lower is better). (b) *Partial* yields lower ratings on perceived usability (higher is better) than *open* in the skateboard domain. Error bars indicate 95% confidence intervals.

*H1a is partially supported.* Although the regret for *partial* sat in between *full* and *open* as expected (being an intermediary between those two levels), it was not significantly different from either. However, *full* did indeed significantly outperform *open*. The interaction effect reveals that the difference between *full* and *open* on regret is driven by results in the *skateboard* domain. *H1b is supported. Delivery* resulted in a significantly lower regret over *skateboard*, as expected.

*H2*: We ran a Cronbach's alpha to verify the reliability of the corresponding Likert scales for measuring focused attention and perceived usability. For focused attention, we observed that the value rose from $\alpha = 0.58$ to $\alpha = 0.65$ without the second item (which asked for a response to the question "The time I spent learning the game strategy passed by quickly." on a 5-point scale), and we remove this item from the analysis accordingly. For perceived usability, we keep all items for the analysis below as removing any of them did not increase the $\alpha = 0.86$ that was obtained using all items.

A two-way mixed ANOVA did not find a significant effect of feedback loop ($F(2, 201) = 1.56, p = 0.21$), nor domain ($F(1, 201) = 0.38, p = 0.54$) on focused attention, nor an interaction effect between feedback loop and domain on focused attention ($F(2, 201) = 1.90, p = 0.15$). A two-way mixed ANOVA found a significant effect of domain on perceived usability ($F(1, 201) = 85.77, p < 0.001$). A $t$-test revealed a significant difference in the perceived usability ratings of *delivery* ($M = 3.57$) and *skateboard* ($M = 2.89$), $t(406) = 6.562, p < 0.001$. Finally, a two-way mixed ANOVA also found an interaction effect between feedback loop and domain on perceived usability ($F(2, 201) = 6.17, p = 0.003$), where Tukey revealed a significant difference between *partial* ($M = 2.64$) and *open* ($M = 3.21$) for *skateboard* ($p = 0.006$, Figure 8(b)). A main effect of feedback loop on perceived usability was not found ($F(2, 201) = 2.06, p = 0.13$).

Table 1. Correctness of the Signs of Reward Weight Estimates from Participants

| | Delivery Domain | | Skateboard Domain | |
|---|---|---|---|---|
| | Correct | Incorrect | Correct | Incorrect |
| *Open* loop | 52% | 48% | 55% | 45% |
| *Partial* closed loop | 54% | 46% | 52% | 48% |
| *Full* closed loop | 59% | 41% | 54% | 46% |

*H2a is not supported.* Although no main effects were found for feedback loop on focused attention or perceived usability, the interaction effects with the *skateboard* domain reveal that *partial* feedback loop is less usable than *open* loop. *H2b is partially supported.* The trend of the domain differences continues with *delivery* yielding significantly higher ratings of perceived usability over *skateboard*, although no difference was found between the domains for focused attention.

*H3:* As participants gave an *improvement* rating for each interaction (e.g., demonstration, feedback), a mean is more descriptive than a median for each participant and for each domain and we use parametric analyses accordingly. A two-way mixed ANOVA indicated a significant effect of domain on improvement ($F(1, 201) = 32.17, p < 0.001$). A $t$-test revealed that the teaching in *delivery* ($M = 3.38$) was rated to yield higher improvement than in *skateboard* ($M = 3.12$), $t(406) = 3.001, p = 0.003$). The ANOVA did not indicate a significant effect of feedback loop ($F(2, 201) = 1.54, p = 0.22$) nor a significant interaction effect ($F(2, 201) = 1.23, p = 0.29$) between feedback loop and domain.

*H3a is not supported.* Feedback loop did not impact ratings of improvement. *H3b is supported.* The ratings suggest that participants learned more overall about the *delivery* domain than the *skateboard* domain.

*H4:* The Kruskal–Wallis H test did not reveal a statistically significant effect of feedback loop on ratings of understanding ($p = 0.41$). However, the Wilcoxon signed-rank test showed a statistically significant difference in ratings of understanding between *delivery* and *skateboard* domains ($Z = -6.474, p < 0.001$). Although the median ratings on understanding of both domains were 4, the mean for *delivery* was 3.90 and the mean for *skateboard* was 3.34.

*H4a is not supported.* Feedback loop did not impact ratings of understanding. *H4b is supported.* The ratings support a difference in the difficulty of the two domains.

Finally, as an exploratory measure, we asked participants at the end of each domain in the user study (having gone through the respective teaching and testing portions) to provide their best estimate as to the weights of the domain's reward features. We evaluated whether the signs of each of the estimated weights were correct as a coarse, first-pass analysis, which can be found in Table 1 as percentages. We note that estimated weights for up to 2 individuals (out of 68) were not recorded for each condition due to technical difficulties.

### 5.3 Discussion

The primary hypothesis of the user study, H1a, was partially supported with *full* closed-loop teaching leading to a significantly lower regret in human test responses over *open* loop teaching. As *partial* closed-loop was explicitly designed to incorporate only a subset of *full*'s framework (i.e., diagnostic tests and feedback, but not additional remedial demonstrations or tests), it predictably led to regret that sat in between *full* and *open* without significant difference to either. Importantly, the three aforementioned conditions each provided the same median number of interactions (where each demonstration or test counts as one interaction), highlighting that the content and the interaction

type matter in instruction. *Full* closed-loop teaching was designed to detect misunderstandings in human's beliefs using diagnostic tests, then address the misunderstanding with tailored remedial demonstrations and tests until the human exhibits understanding through a correct test response. *Open* loop teaching does not provide real-time tailoring of instruction, and *partial* only provides a diagnosis of potential misunderstanding and shallow remediation through quick feedback.

Not too surprisingly, the results indicated a clear difference between the two domains across all measures except focused attention (as the domains were designed to vary in difficulty). Interestingly, there were interaction effects driven by domain. The results show that the significant improvement in objective learning outcomes from *full* closed-loop teaching over *open* comes primarily from the *skateboard* domain, suggesting perhaps that the benefit of the proposed fully closed-loop teaching scheme is greater for more challenging domains.

Despite the improvement in objective learning outcomes, *full* is not simultaneously able to significantly improve usability over *open*. Similar to the observation made in our prior work [35], we again see hints of the dual nature of effective learning that requires mental effort to continuously update one's knowledge (note that the perceived usability questions in this study address a similar construct to mental effort). Indeed, one person in the *full* condition provided the following response to the open-ended question at the conclusion of the study, "Do you have any general comments or feedback on the study? Is there anything you wish [the robot] would've done to help you understand the game strategies better?"

> "I found it a little confusing. Each time I thought I understood the best strategy I was proved wrong. Nothing more [the robot] could have done except give more examples. More examples and more practice might have helped."

*Full* closed-loop teaching employs the counterfactual scaffolding technique of [35] to explicitly select demonstrations for the initial curriculum that the human does not expect to provide maximum information. Although we detect when the human has failed to successfully incorporate knowledge from counterfactual scaffolding demonstrations and remedy with remedial demonstrations and tests, these initial demonstrations can understandably be challenging to grasp. A closed-loop teaching scheme is thus critical for keeping the human learner in the ZPD with intermittent testing, feedback, and targeted instruction.

Interestingly, we also saw another interaction effect where *partial* loop teaching is rated significantly less usable than *open* in the skateboard domain. Several people in *partial* noted that they wanted more demonstrations to clear up confusion, e.g., saying "the strategy on the first game somewhat confused me. Maybe if there were more demonstrations it would be easier to understand its strategy." We hypothesize that it can be frustrating to have diagnostic tests highlight gaps in understanding without providing further instruction (as in the case of *full*) or not highlight potential gaps in understanding at all and provide additional instruction instead (as in the case of *open*).

We also considered analyzing H1 using normalized regret as previously mentioned in Section 5.2. In debating whether to analyze participant test responses using regret or normalized regret, we observed a key tradeoff between the two metrics that is highlighted in Figure 9. While normalizing regret by the reward of the optimal trajectory allows for a fairer comparison between tests of different domains (each with its own unique reward function), it also necessarily scales the reward of each individual error according to the reward of the entire trajectory. For example, while one may argue that the suboptimal test responses that go through mud in Figure 9(a) and (b) are qualitatively the same and should be penalized identically (indeed the regret for both trajectories is 0.64), the normalized regrets are different. The normalized regret for Figure 9(a) is 0.60, while the normalized regret for Figure 9(b) is only 0.43, as mistakenly going through mud comprises a smaller portion

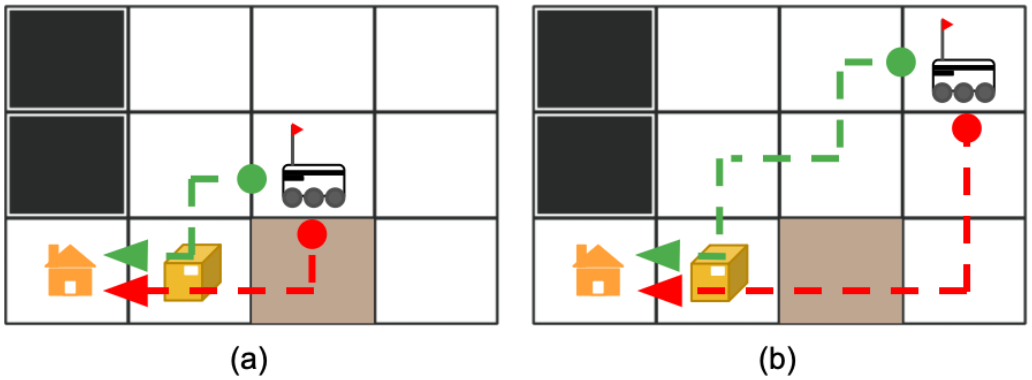


Fig. 9. Two scenarios exemplifying the difference between regret and normalized regret, where optimal and suboptimal trajectories are shown in green and red, respectively. The regret in both scenarios is 0.64, but normalized regret is 0.60 in (a) and 0.43 in (b).

of the longer overall trajectory in Figure 9(b). We note that there are merits to each measure and advise selecting one over the other depending on context. For example, a 10-minute detour in a 5-hour trip to a conference is arguably negligible, but the same detour for a daily commute from the hotel to the conference that should only take 10 minutes is arguably worse and better captured by normalized regret (as regret would be the same). In this article, we are instead interested in measuring *how much someone has learned* and thus each mistake should arguably be penalized the same, regardless of whether it is made once in a shorter trajectory or once in a longer trajectory. We thus opt for regret as our measure. Interestingly, none of the significant findings change for H1 when moving from one form of regret to the other—no new results are added nor taken away. This may be because the sizes of our domains were similar (the delivery and skateboard domains consisted of 10 and 24 grid squares, respectively) and resulted in reward feature counts of the same magnitude. Furthermore, the reward feature weights were $l_2$-normalized such that each weight lay between 0 and 1. For domains of vastly different reward feature counts and reward weights may subsequently lead to vastly different regret and we suggest normalization for fairer comparison across domains.

Finally, the results of asking the participants to guess the weights of the reward features in each domain surprised us (Table 1). Although there were always more, or at least as many, correct answers as incorrect answers, the number of incorrect answers was higher than expected given people's ability to predict the robot's policy in tests. This suggests that the humans likely did not perform IRL as we algorithmically modeled in this article. Furthermore, the proportion of correct answers increases from *open*, to *partial*, to *full* in order of decreasing regret for *delivery*, but not so for *skateboard*. As we observed in our previous work [34], a more difficult and complex domain may have encouraged participants to utilize a different imitation-based learning style than IRL-based learning style, which we further discuss in Section 7.

## 6 User Study on Demonstrations vs. Direct Reward Communication

While this article has focused so far on teaching robot policies in the form of demonstrations, the teaching can take other forms. Interestingly, Sanneman and Shah [57] found that communicating weights of reward features directly performed the best objectively and subjectively in their two domains (waypoints and grid world) compared with HIGHLIGHTS, a policy summarization

technique that communicates the reward function via demonstrations from states with maximal difference between the Q-values for the best and worst actions [3].

We wondered whether direct reward communication would also outperform our closed-loop teaching method in our domains, and also whether there would be synergy in conveying both. We thus ran a second online user study exploring whether direct reward communication would improve the transparency of robot policies in the grid world domains considered in this article.

### 6.1 Study Design

Most of the details of this user study are carried over from the previous user study in Section 5. The within-subject variable was again *domain*, which consisted of the same two conditions as the user study on feedback loop: delivery and skateboard.

The between-subjects variable was *teaching form* with the following three conditions:

— *Direct reward* followed the methodology of [57] and directly provided the numerical reward weights to the participant in a bar graph along with the numerical values.
— *Full* implemented the full closed-loop teaching framework as described earlier in this article as a baseline.
— *Joint* provided both direct reward information via bar graphs and numerical values, as well as the full closed-loop teaching framework.

The user study consisted of two trials, with each trial comprising a teaching portion and a testing portion in one domain. During teaching, participants were first explicitly informed of the reward features of the domain through an informational page. In the *direct reward* or *joint* conditions, the participants were also provided the corresponding reward weights in bar graph form as well as explicit numerical values on this informational page. For these two conditions, the numerical values of the reward weights were also provided on every subsequent page (e.g., alongside demonstrations and tests) to remove the confound of memory. Participants in the *direct reward* condition then moved straight from the informational page on reward weights and features (which comprised the teaching portion) to a page of Likert items that queried their level of focused attention, the perceived usability of their assigned teaching condition, and their subsequent understanding of the robot's policy to close out their teaching portion. Participants in the *full* and *joint* conditions were instead provided demonstrations and perhaps diagnostic tests, corrective feedback, and additional remedial instruction as necessary following the informational page. For every interaction, participants in these two conditions also indicated whether the interaction improved their understanding of the policy using a Likert item. Participants in the *full* and *joint* conditions also closed out their teaching portions by responding to Likert items that queried their level of focused attention, the perceived usability of their assigned teaching condition, and their subsequent understanding of the robot's policy. As noted above, *direct reward* had the shortest teaching portion which we expected to lead to the highest usability ratings, but we expected *joint* to foster greater focused attention through the provision of reward weight and demonstration information that reinforced one another.

Following the teaching portion, participants in all conditions proceeded to the testing portion where they predicted the robot's optimal trajectory in six unseen test environments in random order, which were selected according to prior work [35] to comprise two low, medium, and high difficulty environments each.

We tested the following hypotheses (H1–H4) using the measures (M1–M4) below (all measures are shared with the previous user study in Section 5 but are repeated here for convenience). The Likert scales corresponding to M2 and M4 were provided after the teaching portion but before the testing portion, and the Likert scales corresponding to M3 were provided after each demonstration and test in the teaching portion.

*H1*: (a) The test responses will be best for *joint*, then *full*, then *direct reward*. (b) *Delivery* will result in better test responses over *skateboard*.

*H2*: (a) Focused attention will be highest for *joint*, then *direct reward*, then *full*. Perceived usability will be highest for *direct reward*, then *joint*, then *full*. (b) *Delivery* will result in higher focused attention and perceived usability over *skateboard*.

*H3*: (a) Improvement ratings will be highest for *joint*, then *full* (no improvement ratings were queried for *direct reward*). (b) *Delivery* will result in higher improvement ratings over *skateboard*.

*H4*: (a) Understanding ratings will be highest for *joint*, then *full*, then *direct reward*. (b) *Delivery* will result in higher understanding ratings over *skateboard*.

*M1. Test Response*: The reward of the human's test response, measuring the human's ability to predict the policy.

*M2. Focused Attention and Perceived Usability*: We adapted the User Engagement Scale short form [48] to ask three questions targeting focused attention, each answered with a 5-point Likert scale. Please find the corresponding questions in Appendix A.3.

*M3. Improvement*: "Did this interaction improve your understanding of the game strategy [robot policy]?", answered with a 5-point Likert scale.

*M4. Understanding*: "Do you feel that you now understand the game strategy?", answered with a 5-point Likert scale.

### 6.2 Results

We collected data from 204 participants using Prolific [49]. The participants were approximately 72% male, 26% female, 1% non-binary, and 1% preferred not to disclose, and the ages ranged from 18 to 67 ($M = 31.54$, SD = 9.68). The recruitment process and study were approved by the Carnegie Mellon University Institutional Review Board. Sixty-eight participants were randomly assigned to each of the three between-subjects conditions, and the order of the domains in the study was counterbalanced.

We again present the results below with the caveat that two bugs in the user study code were discovered post hoc. First, only the positions of the particles, and not their weights, were considered when sampling human beliefs from the particle filter (Section 4.1.2). Second, remedial demonstrations and remedial tests that did not minimize the distance to a missed KC were sporadically selected (Section 4.2). Correcting both bugs post hoc reveals that while approximately 2.11% of the interactions in the *joint* teaching form condition could have been different, the vast majority of interactions in this condition would have remained unchanged. Furthermore, since these bugs sporadically produced off-target remedial instruction, we hypothesize that these results represent a lower bound on the efficacy of the *joint* teaching form condition.

*H1*: Consistent with the previous user study, we analyze participant test responses using regret (i.e., the difference between rewards of human and optimal test responses). A two-way mixed ANOVA indicated a significant effect of feedback loop on regret ($F(2, 201) = 23.72, p < 0.001$). Tukey analyses revealed that both *joint* ($M = 0.22$) and *full* ($M = 0.24$) had significantly lower regret compared to *direct reward* ($M = 0.66$), with both at $p < 0.001$. The ANOVA also indicated a significant effect of domain on regret ($F(1, 201) = 51.62, p < 0.001$), where a $t$-test revealed a significant difference between the regret between *delivery* ($M = 0.18$) and *skateboard* ($M = 0.57$), $t(406) = -6.378, p < 0.001$.

Finally, the ANOVA also indicated an interaction effect ($F(2, 201) = 14.65, p < 0.001$) between teaching form and domain. In the *delivery* domain, Tukey revealed that *joint* ($M = 0.12$) led to significantly lower regret compared to *direct reward* ($M = 0.25$), at $p = 0.005$, while *full* ($M = 0.16$) trended toward significantly lower regret than *direct reward* at $p = 0.08$. In the *skateboard* domain,

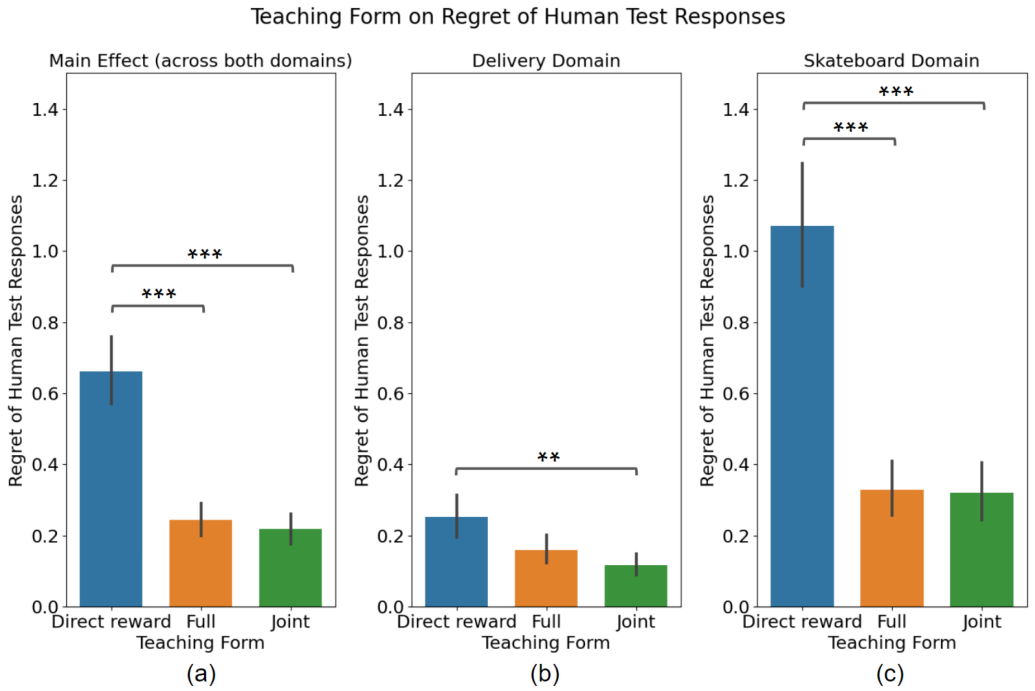


Fig. 10. (a) *Direct reward* leads to significantly higher regret in human test responses compared to *full* and *joint*. (b–c) The gap between the regret from direct reward and the other teaching forms is notably bigger in the skateboard domain than the delivery domain, where the skateboard was objectively and subjectively deemed by participants to be more challenging.

Table 2. Mean Regret of Human Test Responses across the Five Conditions of the Two User Studies (Lower is Better)

| | *Open* loop | *Partial* | *Full* closed loop | ***Joint*** | *Direct reward* |
|---|---|---|---|---|---|
| Delivery | 0.210 | 0.162 | 0.160 | **0.118** | 0.254 |
| Skateboard | 0.624 | 0.412 | 0.328 | **0.320** | 1.070 |

Tukey analyses revealed that both *joint* ($M = 0.32$) and *full* ($M = 0.33$) had significantly lower regret compared to *direct reward* ($M = 1.07$), with both at $p < 0.001$ (Figure 10).

*H1a is partially supported.* While *joint* and *full* each led to significantly lower regret compared to *direct reward*, *joint* did not lead to significantly lower regret with respect to *full* as expected. An exploration of the interaction effect revealed that the differences between *direct reward* and either *joint* or *full* are larger in the *skateboard* domain, again suggesting an interesting influence of domain that will be discussed in more detail in the next section. *H1b is supported. Delivery* resulted in a significantly lower regret over *skateboard*, as expected.

For completeness, Table 2 compares the mean regret of human test responses across the five conditions comprising the two user studies conducted in this article. Of note are *direct reward* leading to the worst performance in both domains, and full and joint performing the best (a statistically significant difference was not found between these two conditions). We provide similar tables that compare the results of other measures in the two user studies in subsequent analyses. In Tables 2–6, the best outcomes across the five conditions of the two user studies are bolded for reference.

Table 3. Mean Focused Attention Rating across the Five Conditions of the Two User Studies (Higher is Better)

| | *Open* loop | *Partial* | *Full* closed loop | ***Joint*** | *Direct reward* |
|---|---|---|---|---|---|
| Delivery | 4.279 | 4.412 | 4.272 | **4.522** | 4.169 |
| Skateboard | 4.309 | 4.360 | 4.169 | **4.397** | 4.147 |

Table 4. Mean Perceived Usability Rating across the Five Conditions of the Two User Studies (Higher is Better)

| | *Open* loop | *Partial* | *Full* closed loop | ***Joint*** | *Direct reward* |
|---|---|---|---|---|---|
| Delivery | 3.525 | 3.485 | 3.686 | 3.569 | **3.819** |
| Skateboard | 3.211 | 2.637 | 2.819 | 2.873 | **3.691** |

Table 5. Mean Improvement Rating across the Five Conditions of the Two User Studies (Higher is Better)

| | *Open* loop | *Partial* | *Full* closed loop | ***Joint*** | *Direct reward* |
|---|---|---|---|---|---|
| Delivery | 3.430 | 3.269 | 3.440 | **3.848** | N/A |
| Skateboard | 3.270 | 2.953 | 3.125 | **3.729** | N/A |

Table 6. Mean Understanding Rating across the Five Conditions of the Two User Studies (Higher is Better)

| | *Open* loop | *Partial* | *Full* closed loop | ***Joint*** | *Direct reward* |
|---|---|---|---|---|---|
| Delivery | 3.809 | 3.882 | 4.015 | **4.353** | 4.147 |
| Skateboard | 3.589 | 3.147 | 3.294 | 4.029 | **4.279** |

*H2*: We ran a Cronbach's alpha to verify the reliability of the corresponding Likert scales for measuring focused attention and perceived usability. For focused attention, we observed that the value again rose from $\alpha = 0.61$ to $\alpha = 0.67$ without the second item (which asked for a response to the question "The time I spent learning the game strategy passed by quickly." on a 5-point scale) and we remove this item from the analysis accordingly. For perceived usability, we keep all items for the analysis below as removing any of them did not significantly increase the $\alpha = 0.85$ that was obtained using all items.

A two-way mixed ANOVA found a significant effect of feedback loop ($F(2, 201) = 5.63, p = 0.004$) on focused attention. Tukey analyses revealed that *joint* ($M = 4.46$) led to significantly higher ratings over *full* ($M = 4.22$) and *direct reward* ($M = 4.16$), at $p = 0.033$ and $p = 0.005$, respectively. While the ANOVA reported a significant effect of domain on focused attention ($F(1, 201) = 5.11, p = 0.02$), a post hoc $t$-test revealed that the difference between focused attention ratings in *delivery* ($M = 4.32$) and *skateboard* ($M = 4.24$) was not significant, $t(406) = 1.349, p = 0.18$. The ANOVA did not find an interaction effect between teaching form and domain on focused attention ($F(2, 201) = 0.72$, $p = 0.49$). For completeness, Table 3 compares the mean focused attention rating across the five conditions comprising the two user studies conducted in this paper.

A two-way mixed ANOVA also found a significant main effect of teaching form on perceived usability ($F(2, 201) = 8.30, p < 0.001$), where Tukey revealed that *direct reward* ($M = 3.76$) led to significantly higher ratings over *joint* ($M = 3.22$) and *full* ($M = 3.25$), at $p = 0.001$ and $p =$

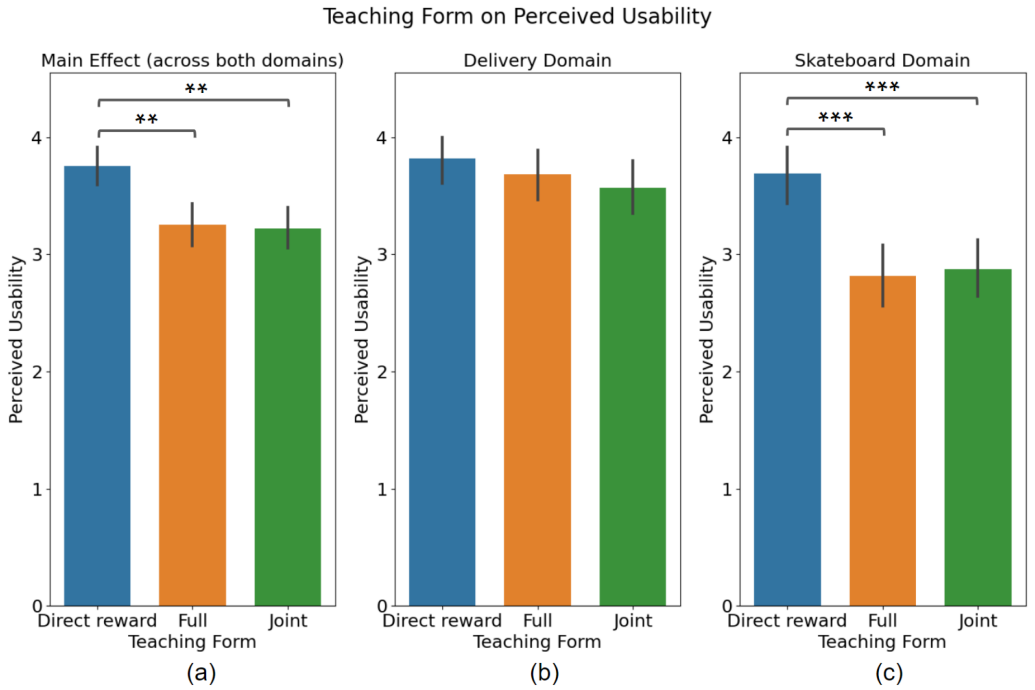


Fig. 11. (a) *Direct reward* leads to significantly higher ratings of perceived usability compared to *full* and *joint*. (b–c) The main effect is mostly driven by the skateboard domain.

0.002, respectively. The ANOVA also revealed a significant effect of domain on perceived usability ($F(1, 201) = 78.51, p < 0.001$), and a post hoc $t$-test revealed that a significant difference between ratings in *delivery* ($M = 3.70$) and *skateboard* ($M = 3.13$), $t(406) = 5.641, p < 0.001$. Finally, the ANOVA also found an interaction effect between teaching form and domain on perceived usability ($F(2, 201) = 12.36, p < 0.001$), where Tukey revealed that *direct reward* ($M = 3.70$) led to significantly higher ratings over *joint* ($M = 2.87$) and *full* ($M = 2.82$), at $p < 0.001$ for both, only for *skateboard* (no significant differences were found for the *delivery* domain—Figure 11). For completeness, Table 4 compares the mean perceived usability rating across the five conditions comprising the two user studies conducted in this paper.

*H2a is partially supported. Joint* resulted in significantly higher focused attention ratings over *full* and *direct reward* as expected. However, there was no difference in the focused attention ratings between *full* and *direct reward*. *Direct reward* resulted in significantly higher perceived usability ratings over *joint* and *full* as expected, but there was no difference in perceived usability ratings between *joint* and *full*. Interestingly, post hoc analyses of the interaction effect between domain and usability find that the significant main effects are entirely driven by *skateboard*. *H2b is partially supported.* The trend of domain differences continues with *delivery* yielding significantly higher ratings of perceived usability over *skateboard*, although no difference was found between the domains for focused attention.

*H3*: As participants gave an *improvement* rating for each interaction in *joint* and *full* (e.g., demonstration, feedback), a mean is more descriptive than a median for each participant and for each domain and we again use parametric analyses accordingly.[4] A two-way mixed ANOVA indicated a

[4]Due to technical difficulties, the improvement ratings of 2 out of 68 participants in the *joint* condition were not recorded.

significant effect of teaching form on improvement ($F(1, 132) = 11.85, p = 0.001$). A $t$-test revealed that *joint* ($M = 3.77$) yielded significantly higher ratings on improvement over full ($M = 3.23$), $t(134) = 3.613, p = 0.001$. The ANOVA also indicated a significant effect of domain on improvement ($F(1, 132) = 18.23, p < 0.001$). A $t$-test revealed that the teaching in *delivery* ($M = 3.64$) was rated to yield higher improvement than in *skateboard* ($M = 3.43$), $t(270) = 1.900, p = 0.058$). The ANOVA did not indicate a significant interaction effect between teaching form and domain ($F(1, 134) = 3.36, p = 0.06$). For completness, Table 5 compares the mean improvement rating across the five conditions comprising the two user studies conducted in this paper.

*H3a is supported.* As expected, *joint* lead to higher ratings on improvement over *full*. *H3b is supported.* The ratings also suggest that participants learned more overall about the *delivery* domain than the *skateboard* domain.

*H4*: The Kruskal–Wallis H test revealed that *full* ($M = 3.65$) yielded significantly lower ratings of understanding compared to *joint* ($M = 4.19$) as well as *direct reward* ($M = 4.21$), at $p < 0.001$ for both. The Wilcoxon signed-rank test also showed a statistically significant change in ratings of understanding between *delivery* and *skateboard* domains ($Z = -4.83, p < 0.001$). Although the median ratings on understanding for both domains were 4, the mean for *delivery* was 4.17 and the mean for *skateboard* was 3.87. For completness, Table 6 compares the mean understanding rating across the five conditions comprising the two user studies conducted in this paper.

*H4a partially supported.* While ratings on understanding were higher for *joint* over *full* as expected, ratings on understanding were also higher for *direct reward* over *full*. *H4b is supported.* The ratings on understanding were higher in *delivery* than *skateboard* as expected.

### 6.3 Discussion

We first observe that the best reward communication method is likely domain-dependent, and we specifically hypothesize that conveying numerical reward weights alone is increasingly insufficient as a teaching form as domain complexity increases. Not only does *direct reward* lead to significantly higher regret than *joint* and *full*, the gap is larger in *skateboard* over *delivery*—where we consider the former domain more complex than the latter. In our study, *delivery* and *skateboard* each had three reward features, but both objective and subjective results strongly indicated that the latter domain was more challenging for participants. First, *delivery* often supports more "local" planning around individual mud patches and batteries, whereas *skateboard* requires more "global" planning that considers the distance to the skateboard and the subsequent distance to the goal to determine whether it is worth detouring to pick up the skateboard on the way to the goal. In this, we would argue that the *skateboard* domain has an implicit dependence between the action and skateboard reward features that must be carefully considered in advance before selecting between a path that detours to pick up a skateboard along the way and a path that does not. Furthermore, the grid size of the *delivery* domain was smaller than *skateboard* and the reward weights were more coarse (the reward weights for *delivery* were −3, 3.5, and 1 for moving out of mud, picking up the battery, and for each action, respectively, whereas the reward weights for *skateboard* were 0.825, 0.4875, and −1 for moving with the skateboard, moving on the path, and for each action, respectively). This allowed for more subtle tradeoffs to be made in the skateboard domain such that the difference in reward between a trajectory that detoured to pick up the skateboard first, a trajectory that detoured to go on the path instead, and a trajectory that went straight toward the goal could differ by only minute amounts.

Given our domain-dependent results, we argue that domain characterization is an open and important topic that can help us infer which results may be generalized to other domains. One such characterization is domain complexity, which is difficult to define. Sanneman and Shah [57] offer a definition for the related concept of reward complexity as the number of features that comprise the

reward function. Although the number of reward features is a reasonable starting point for domain complexity, our observations in the delivery and skateboard domains suggest that one must also consider the degree of interaction between features and the subtleties of the tradeoffs that result from the reward features. And though we do not test this in our user study, another consideration when considering domain complexity could be the degree of familiarity in addition to the size of the state space. As Qian and Unhelkar [53] note, their navigation domain had a much smaller state space of 400 over other domains that had a state space of 3,200 and 80,000, but it was the most challenging for their participants to grasp due to some of the navigation robot's less intuitive movements.

Second, we observe a potential synergy between different teaching forms where they can help reinforce each other's information. Although we did not find any difference in regret, *joint* has significantly higher ratings on improvement and focused attention than *full*. Interestingly, a few qualitative quotes from the *direct reward* or *full* conditions suggest that participants wanted the information that was outside the purview of their condition. In response to the open-ended question at the conclusion of the study, "Do you have any general comments or feedback on the study? Is there anything you wish [the robot] would've done to help you understand the game strategies better?", two participants in the *direct reward* condition replied:

> "a demonstration instead of written rules might have helped a bit more," and
>
> "Maybe an example puzzle with optimal moves demonstrated,"

indicating a desire for demonstrations as well. And one participant in the *full* condition replied:

> "If [the robot] told me the implication of moving into yellow or purple boxes, it would have helped me a lot,"

indicating a desire for direct information regarding the effect of various reward features (e.g., perhaps in the form of numerical weights). And people who received both numerical weights and demonstrations in the *joint* condition, replied:

> "This demonstration reinforced to me the importance of obtaining the orange rectangle as moving with it results in a + 0.825% energy change," and
>
> "I already knew to avoid the yellow square, and would have moved the same way as demonstrated,"

which reveal the dual possibility for different teaching forms to be helpfully reinforcing or unhelpfully redundant. To the latter point, one must be mindful of cognitive overload when providing too much information at once, which can lead to a worse understanding of model decision-making [50].

Third, we have defined transparency as understandability and predictability, borrowing from the work of Endsley [16], a leading expert in human situational awareness involving intelligent agents. One can easily imagine how understandability and predictability can be correlated to one another: high understandability could improve predictability through forward simulation, and high predictability could improve understanding through the generation of data that could support model building. However, we observe that high self-reports of understanding do not always translate to corresponding performance. While *direct reward* led to significantly higher levels of reported understanding[5] over *full*, as well as significantly higher ratings on usability over *full*, *direct*

[5]We note that we queried participants for their perceived understanding right after the conclusion of the teaching portion of the user study, and before the testing portion. We hypothesize that perceptions of understanding may have changed when queried after the testing portion.

*reward* also led to significantly worse objective performance. Our results raise the possibility that people may believe that their knowledge is sufficient and may terminate learning early (especially since effective learning often requires significant mental effort as previous results in this article and our prior work [35] have shown), even when tests would likely reveal significant gaps in their knowledge. All in all, our results point to a need for a closed-loop, robot-driven teaching that provides tests and additional instruction as needed to discover and reconcile gaps in the human's understanding. And though our results support robot-driven teaching, Qian and Unhelkar [53] found that a hybrid strategy where participants could choose between agent-selected and user-requested examples outperformed only agent-selected examples and was also subjectively preferred. However, we note that they fixed the teaching budget, and an interesting direction for future work may be in exploring how to balance agent-driven and user-driven learning given a flexible teaching budget (e.g., the human may be feeling unmotivated and wish to terminate learning after a few insufficient examples).

Finally, understandability is a multifaceted concept that can be difficult to measure in practice. While the accuracy of a person's prediction of a robot's behavior is arguably the most common submeasure of understandability (e.g., [6, 25, 31, 57]), other measures include coding responses to an open-ended question regarding robot decision-making (e.g., [6, 57]), agent preference elicitation and feature subselection [57], and verification of agent response and counterfactual reasoning [31]. Our user studies that tested participants' abilities to predict robot behavior and our single Likert-scale item querying gross understanding are incomplete measures, and we also leave how one may query and measure a human's understanding of robot decision-making more comprehensively for future work. We consider other limitations and opportunities for follow-on work in the next section.

## 7 Limitations and Future Work

In this work, we focused on teaching a low-dimensional reward of a robot that specifically took the form of a weighted linear combination of reward features. For more high-dimensional reward functions, recent work has begun leveraging such abstractions, or often referred to as concepts, to increase the interpretability of policies learned through RL [8, 57, 68]. However, these methods require the human to hand-specify the concepts. Automatically distilling high-dimensional reward features into low-dimensional and semantically meaningful concepts and selecting demonstrations that convey both the concepts and the weighting will be an important direction moving forward. Furthermore, we constrained ourselves to grid worlds of limited size and diversity (e.g., the number and locations of possible mud and path patches in the delivery and skateboard domains were decided a priori) that could support exhaustive enumeration. In moving to continuous domains that may not afford an exhaustive enumeration of all possible demonstrations, we may potentially take inspiration from work like goal recognition design [28], which aims to find a domain instance that forces a robot to reveal its objective as early as possible, to formulate the real-time enumeration of demonstrations as a search problem.

In addition to IRL, **Imitation Learning (IL)** is also a commonly accepted model of human learning [9, 22, 32], which models humans as learning the optimal behavior directly from demonstrations (as opposed to through an intermediate reward function like IRL). There are a number of possible algorithms that support both styles [47], and it is not always obvious which style or algorithm would best model human learning in a given situation. It is also possible that people switch between IRL and IL-style reasoning (e.g., depending on the familiarity of the domain [32], which can even change as a function of the number of demonstrations seen [34]), or perhaps there is yet another style of learning from demonstrations that humans employ. The findings of Lage et al. [32] additionally suggest that human learning of the robot's policy can increase if the

robot correctly models the human learning style (e.g., IRL vs IL) when generating demonstrations. Determining when humans employ IRL or IL, and identifying other styles of human learning from demonstrations will be interesting future endeavors.

Finally, we largely restricted ourselves to increasing the transparency of robot policies through demonstrations in this article. However, this is just one form that policy and reward teaching can take. We saw in the follow-up study in Section 6 that direct reward communication integrated nicely with demonstrations to yield high objective and subjective outcomes nearly across the board. This highlights the potential synergies that can arise from employing complementary explanation techniques; e.g., global *policy-level* techniques that convey an understanding of a robot's overall behavior through representative examples can be combined with local *feature importance* techniques that highlight the contextual factors that influence a robot's single decision [43]. Additionally, language is another common modality for teaching that shares strengths and weaknesses that are complementary to that of demonstrations (e.g., for explaining agent decision-making [13, 14]). While language has the ability to convey complex, generalizable concepts more effectively than demonstrations, language is heavily dependent on shared abstraction between parties (e.g., what a rook is in the statement "In chess, rooks move along rows and columns."), can suffer from ambiguity, and may struggle to convey certain physical concepts such as spatial movement, color, and so on. While demonstrations are inherently grounded, they require the learner to infer the underlying rules or concepts, some of which may be difficult to demonstrate exhaustively (e.g., it would be inefficient to demonstrate all the possible ways that the rook can move on a chess board). Recent work has begun exploring leveraging the complementary strengths of language and demonstrations for humans to teach robots [42, 67], which we posit will also be effective conversely for robots to teach their policies and reward functions to humans.

## 8 Conclusion

As robots increasingly abound in society, it is important that their decision-making is *transparent*, e.g., such that the actions taken by robots are predictable and understandable to humans. Transparency is critical for not only developers in reviewing and ensuring proper robot function but also for end users in having calibrated expectations—preventing undertrust and disuse, or overtrust and misuse. Machine teaching provides a principled framework for selecting demonstrations *a priori* that increases the transparency of robot policies to humans; however, individuals may differ in their learning trajectories *in situ*. We thus augment a curriculum of preselected demonstrations with a novel closed-loop teaching framework inspired by key concepts from the education literature to provide tailored instruction. A user study finds that our teaching framework consisting of demonstrations, tests, feedback, and remedial instruction reduces the regret in human test responses by 43% over a baseline.

Furthermore, demonstrations are only one means of improving the transparency of robot policies and, inspired by results by Sanneman and Shah [57], we also saw how directly conveying the robot's underlying reward weights fared in our domains, both as a standalone method as well as in conjunction with our closed-loop teaching via demonstrations. In contrast to their findings, we found that directly conveying the robot's reward weights yielded significantly worse human test responses, although it led to reports of high understanding and usability as a teaching form. However, providing both reward weights and demonstrations provided synergy that allowed for high objective and subjective outcomes nearly across the board, highlighting that different teaching forms can provide complementary information that can augment one another. Echoing the broader consensus in the explainable AI literature that there is no one-size-fits-all explainability method, we leave the exploration of the synergy of various methods in the diversity of possible domains as an exciting direction for future work.

## References


[1] Pieter Abbeel and Andrew Y. Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the 21st International Conference on Machine Learning*, 1–8.

[2] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. 2023. Do as I can, not as I say: Grounding language in robotic affordances. In *Conference on Robot Learning*. PMLR, 287–318.

[3] Dan Amir and Ofra Amir. 2018. Highlights: Summarizing agent behavior to people. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 1168–1176.

[4] Ofra Amir, Finale Doshi-Velez, and David Sarne. 2019. Summarizing agent strategies. *Autonomous Agents and Multi-Agent Systems* 33, 5 (Sep. 2019), 628–644.

[5] Yotam Amitai, Yael Septon, and Ofra Amir. 2024. Explaining reinforcement learning agents through counterfactual action outcomes. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, 10003–10011.

[6] Andrew Anderson, Jonathan Dodge, Amrita Sadarangani, Zoe Juozapaitis, Evan Newman, Jed Irvine, Souti Chattopadhyay, Alan Fern, and Margaret Burnett. 2019. Explaining reinforcement learning to mere mortals: An empirical study. arXiv:1903.09708. Retrieved from https://arxiv.org/abs/1903.09708

[7] I-Cheng Chang and Shih-Yao Lin. 2010. 3D human motion tracking based on a progressive particle filter. *Pattern Recognition* 43, 10 (2010), 3621–3635.

[8] Devleena Das, Sonia Chernova, and Been Kim. 2023. State2explanation: Concept-based explanations to benefit agent learning and user understanding. In *Conference on Neural Information Processing Systems*, 67156–67182.

[9] Nathaniel D. Daw, Yael Niv, and Peter Dayan. 2005. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience* 8, 12 (2005), 1704–1711.

[10] Jesús Martínez Del Rincón, Dimitrios Makris, Carlos Orrite Uruñuela, and Jean-Christophe Nebel. 2011. Tracking human position and lower body parts using Kalman and particle filters constrained by human biomechanics. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41, 1 (2010), 26–37.

[11] Inderjit S. Dhillon and Suvrit Sra. 2003. *Modeling Data Using Directional Distributions*. Technical Report. Citeseer.

[12] Arnaud Doucet and Adam M. Johansen. 2009. A tutorial on particle filtering and smoothing: Fifteen years later. In *Handbook of Nonlinear Filtering*. Dan Crisan and Boris Rozovskii (Eds.), Oxford University Press.

[13] Upol Ehsan, Brent Harrison, Larry Chan, and Mark O. Riedl. 2018. Rationalization: A neural machine translation approach to generating natural language explanations. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 81–87.

[14] Upol Ehsan, Pradyumna Tambwekar, Larry Chan, Brent Harrison, and Mark O. Riedl. 2019. Automated rationale generation: A technique for explainable AI and its effects on human perceptions. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, 263–274.

[15] Mica R. Endsley. 2017. From here to autonomy: Lessons learned from human–automation research. *Human Factors* 59, 1 (2017), 5–27.

[16] Mica R. Endsley. 2023. Supporting human-AI teams: Transparency, explainability, and situation awareness. *Computers in Human Behavior* 140 (2023), 107574.

[17] Justin Fu, Katie Luo, and Sergey Levine. 2018. Learning robust rewards with adversarial inverse reinforcement learning. In *International Conference on Learning Representations*, 15 pages.

[18] Omer Gottesman, Joseph Futoma, Yao Liu, Sonali Parbhoo, Leo Celi, Emma Brunskill, and Finale Doshi-Velez. 2020. Interpretable off-policy evaluation in reinforcement learning by highlighting influential transitions. In *International Conference on Machine Learning*, 3658–3667.

[19] Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. 2018. Visualizing and understanding atari agents. In *International Conference on Machine Learning*. PMLR, 1792–1801.

[20] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*. PMLR, 1861–1870.

[21] John Hattie and Shirley Clarke. 2018. *Visible Learning: Feedback*. Routledge.

[22] Mark K. Ho and Thomas L. Griffiths. 2022. Cognitive science as a source of forward and inverse models of human decisions for robotics and control. *Annual Review of Control, Robotics, and Autonomous Systems* 5 (2022), 33–53.

[23] Dorit S. Hochbaum and David B. Shmoys. 1985. A best possible heuristic for the k-center problem. *Mathematics of Operations Research* 10, 2 (1985), 180–184.

[24] Sandy H. Huang, Kush Bhatia, Pieter Abbeel, and Anca D. Dragan. 2018. Establishing appropriate trust via critical states. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3929–3936.

[25] Sandy H. Huang, David Held, Pieter Abbeel, and Anca D. Dragan. 2019. Enabling robots to communicate their objectives. *Autonomous Robots* 43, 2 (2019), 309–326.

[26] Julian Jara-Ettinger. 2019. Theory of mind as inverse reinforcement learning. *Current Opinion in Behavioral Sciences*

29 (2019), 105–110.
[27] Zoe Juozapaitis, Anurag Koul, Alan Fern, Martin Erwig, and Finale Doshi-Velez. 2019. Explainable reinforcement learning via reward decomposition. In *IJCAI/ECAI Workshop on Explainable Artificial Intelligence*, 7 pages.
[28] Sarah Keren, Avigdor Gal, and Erez Karpas. 2014. Goal recognition design. In *24th International Conference on Automated Planning and Scheduling*, 154–162.
[29] Kenneth R. Koedinger, Julie L. Booth, and David Klahr. 2013. Instructional complexity and the science to constrain it. *Science* 342, 6161 (2013), 935–937.
[30] Kenneth R. Koedinger, Albert T. Corbett, and Charles Perfetti. 2012. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science* 36, 5 (2012), 757–798.
[31] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Samuel J. Gershman, Been Kim, and Finale Doshi-Velez. 2018. An Evaluation of the Human-Interpretability of Explanation. In *Conference on Neural Information Processing Systems (NeurIPS) Workshop on Correcting and Critiquing Trends in Machine Learning*, 24 pages.
[32] Isaac Lage, Daphna Lifschitz, Finale Doshi-Velez, and Ofra Amir. 2019. Exploring computational user models for agent policy summarization. In *International Joint Conference on Artificial Intelligence*, 1401–1407.
[33] Michael S. Lee. 2024. *Improving the Transparency of Agent Decision Making to Humans Using Demonstrations*. Ph.D. Dissertation. Carnegie Mellon University, Pittsburgh, PA.
[34] Michael S. Lee, Henny Admoni, and Reid Simmons. 2021. Machine teaching for human inverse reinforcement learning. *Frontiers in Robotics and AI* 8 (2021), 693050.
[35] Michael S. Lee, Henny Admoni, and Reid Simmons. 2022. Reasoning about counterfactuals to improve human inverse reinforcement learning. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 9140–9147.
[36] Michael S. Lee, Henny Admoni, and Reid Simmons. 2023. Closed-loop reasoning about counterfactuals to improve policy transparency. In *International Conference on Machine Learning (ICML) Workshop on Counterfactuals in Minds and Machines*, 12 pages.
[37] Scott Lenser and Manuela Veloso. 2000. Sensor resetting localization for poorly modelled mobile robots. In *International Conference on Robotics and Automation*, 1225–1232.
[38] Roger Levy, Florencia Reali, and Thomas Griffiths. 2008. Modeling the effects of memory on human online sentence processing with particle filters. *Advances in Neural Information Processing Systems* 21 (2008), 937–944.
[39] Tiancheng Li, Miodrag Bolic, and Petar M. Djuric. 2015. Resampling methods for particle filtering: Classification, implementation, and strategies. *IEEE Signal Processing Magazine* 32, 3 (2015), 70–86.
[40] Tiancheng Li, Shudong Sun, and Tariq Pervez Sattar. 2013. Adapting sample size in particle filters through KLD-resampling. *Electronics Letters* 49, 12 (2013), 740–742.
[41] Tiancheng Li, Shudong Sun, Tariq Pervez Sattar, and Juan Manuel Corchado. 2014. Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches. *Expert Systems with Applications* 41, 8 (2014), 3944–3954.
[42] Toby Jia-Jun Li, Marissa Radensky, Justin Jia, Kirielle Singarajah, Tom M. Mitchell, and Brad A. Myers. 2019. Pumice: A multi-modal agent that learns concepts and conditionals from natural language and demonstrations. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, 577–589.
[43] Stephanie Milani, Nicholay Topin, Manuela Veloso, and Fei Fang. 2023. Explainable reinforcement learning: A survey and comparative review. *ACM Computing Surveys* 56, 7 (2023), 1–36.
[44] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. arXiv:1312.5602. Retrieved from https://arxiv.org/abs/1312.5602
[45] Andrew Y. Ng and Stuart Russell. 2000. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning*, 2–9.
[46] Matthew L. Olson, Roli Khanna, Lawrence Neal, Fuxin Li, and Weng-Keen Wong. 2021. Counterfactual state explanations for reinforcement learning agents via generative deep learning. *Artificial Intelligence* 295 (2021), 103455.
[47] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J. Andrew Bagnell, Pieter Abbeel, Jan Peters. 2018. An algorithmic perspective on imitation learning. *Foundations and Trends in Robotics* 7, 1-2 (2018), 1–179.
[48] Heather L. O'Brien, Paul Cairns, and Mark Hall. 2018. A practical approach to measuring user engagement with the refined user engagement scale (UES) and new UES short form. *International Journal of Human-Computer* 112 (2018), 28–39.
[49] Stefan Palan and Christian Schitter. 2018. Prolific. ac—A subject pool for online experiments. *Journal of Behavioral and Experimental Finance* 17 (2018), 22–27.
[50] Forough Poursabzi-Sangdeh, Daniel G. Goldstein, Jake M. Hofman, Jennifer Wortman Wortman Vaughan, and Hanna Wallach. 2021. Manipulating and measuring model interpretability. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 1–52.

[51] Erika Puiutta and Eric M. S. P. Veith. 2020. Explainable reinforcement learning: A survey. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*. Springer, 77–95.
[52] Peizhu Qian, Harrison Huang, and Vaibhav Unhelkar. 2024. PPS: Personalized policy summarization for explaining sequential behavior of autonomous agents. In *Proceedings of the 7th AAAI/ACM Conference on AI, Ethics, and Society*, 1167–1179.
[53] Peizhu Qian and Vaibhav Unhelkar. 2022. Evaluating the role of interactivity on improving transparency in autonomous agents. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, 1083–1091.
[54] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. 2017. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. arXiv:1709.10087. Retrieved from https://arxiv.org/abs/1709.10087
[55] Deepak Ramachandran and Eyal Amir. 2007. Bayesian inverse reinforcement learning. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, 2586–2591.
[56] Henry L. Roediger III and Jeffrey D. Karpicke. 2006. The power of testing memory: Basic research and implications for educational practice. *Perspectives on Psychological Science* 1, 3 (2006), 181–210.
[57] Lindsay Sanneman and Julie A. Shah. 2022. An empirical study of reward explanations with human-robot interaction applications. *IEEE Robotics and Automation Letters* 7, 4 (2022), 8956–8963.
[58] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. arXiv:1707.06347. Retrieved from https://arxiv.org/abs/1707.06347
[59] Hanan Shteingart and Yonatan Loewenstein. 2014. Reinforcement learning and human behavior. *Current Opinion in Neurobiology* 25 (2014), 93–98.
[60] Andrew Silva, Matthew Gombolay, Taylor Killian, Ivan Jimenez, and Sung-Hyun Son. 2020. Optimization methods for interpretable differentiable decision trees applied to reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, 1855–1865.
[61] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the game of go without human knowledge. *Nature* 550, 7676 (2017), 354–359.
[62] Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction*. MIT Press, 552 pages.
[63] Lev Semenovich Vygotsky. 1980. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press, 174 pages.
[64] Lindsay Wells and Tomasz Bednarz. 2021. Explainable ai and reinforcement learning—a systematic review of current approaches and trends. *Frontiers in AI* 4 (2021), 550030.
[65] Markus Wulfmeier, Peter Ondruska, and Ingmar Posner. 2015. Maximum entropy deep inverse reinforcement learning. arXiv:1507.04888. Retrieved from https://arxiv.org/abs/1507.04888
[66] Michael S. K. Yi, Mark Steyvers, and Michael Lee. 2009. Modeling human performance in restless bandits with particle filters. *The Journal of Problem Solving* 2, 2 (2009), 5.
[67] Albert Yu and Raymond J. Mooney. 2022. Using both demonstrations and language instructions to efficiently learn robotic tasks. arXiv:2210.04476. Retrieved from https://arxiv.org/abs/2210.04476
[68] Renos Zabounidis, Joseph Campbell, Simon Stepputtis, Dana Hughes, and Katia P. Sycara. 2023. Concept learning for interpretable multi-agent reinforcement learning. In *Conference on Robot Learning*. PMLR, 1828–1837.
[69] Xiaojin Zhu. 2015. Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 4083–4087.
[70] Brian D. Ziebart, Andrew L. Maas, J. Andrew Bagnell, and Anind K. Dey. 2008. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd National Conference on Artificial Intelligence*, 1433–1438.

# A Appendix

## A.1 Custom Distribution for Updating Weights in Particle Filter

We propose a custom probability distribution for updating the weight of each particle given constraints from a demonstration or test (please refer back to Section 4.1.1 for more information). As a summary, the custom distribution is composed of a uniform distribution that aligns with the consistent half-space of the constraint and the von Mises–Fisher distribution (a generalization of the Gaussian distribution on a sphere [11]) whose mean direction aligns with the inconsistent half-space (Figure 4). The uniform distribution asserts that any particle lying on the consistent half-space is equally valid for that demonstration, whereas the von Mises–Fisher distribution asserts

that a particle is exponentially less likely to have generated that demonstration as you move away from the consistent side of the constraint.

The resulting pdf of the custom distribution is

$$f_c(x, \mu, \kappa) = \begin{cases} \frac{1}{c_1 4\pi}, & \mu^\top x \geq 0 \\ \frac{c_2 \kappa e^{\kappa \mu^\top x}}{c_1 2\pi(e^\kappa - e^{-\kappa})}, & \mu^\top x < 0, \end{cases} \tag{A1}$$

with a normalizing constant $c_1$ that ensures that the pdf sums to 1

$$c_1 = \frac{1}{\int_0^\pi \int_{\frac{\pi}{2}}^{\frac{3\pi}{2}} \frac{c_2 \kappa e^{\kappa cos(\theta) \cdot sin(\phi)} sin(\phi)}{2\pi(e^\kappa - e^{-\kappa})} d\theta d\phi + 0.5}, \tag{A2}$$

and a scaling constant $c_2$ that matches the probability of the von Mises–Fisher distribution ($f_v$) to that of the uniform distribution at the meeting point of the two distributions

$$c_2 = \frac{1}{4\pi f_v(y, \mu, \kappa)}, \forall y \ s.t. \ \mu^\top y = 0. \tag{A3}$$

Although the custom distribution naturally generalizes to higher dimensions, the particles in our two domains each have three reward features and are constrained to the 2-sphere. The pdf conveyed in Equations (A1)–(A3) is thus specified for the 2-sphere.

In addition to its mean direction, the von Mises–Fisher distribution is described by its concentration parameter $\kappa$, which, as the name implies, captures how concentrated the distribution is around its mean. In our experiments, we set $\kappa$ to be 2, which we empirically observed as providing the desired signal-to-noise ratio during the particle weight updates ($\kappa = 0$ corresponds to the uniform distribution and the distribution becomes more peaked around the mean, and less noisy, as $\kappa$ increases).

### A.2 Particle Filter Resampling and Resetting

We address common challenges to using particle filters in practice. Sample degeneracy occurs when successive updates to the weights of the particles cause only a few particles to have high weight and the particle filter fails to model regions of interest in the posterior with sufficient detail [41]. Furthermore, the number of particles (i.e., sample size) should adapt to the complexity of the distribution being modeled. To address both concerns, we rely on KLD-resampling [40] to obtain the sample size that bounds the Kullback–Leibler (KL) divergence between the sample-based maximum likelihood estimate and the true posterior distribution, and simultaneously rely on systematic resampling [39] to concentrate the sampling near regions of high probability. Finally, measures to combat sample degeneracy can actually cause sample impoverishment, where the particle filter is too concentrated and not amenable to future shifts in the posterior. Thus, we resample only when the effective sample size (a measure of sample degeneracy) drops below a predefined threshold and also add Gaussian noise when resampling the particles [41]. This limited resampling balances the risk of running into sample degeneracy or sample impoverishment, which are at opposite extremes.

Finally, the particle filter may converge, then suddenly obtain new information that is heavily inconsistent with the current distribution. In this case, the filter will struggle to update, as none or very few of the particle weights would be increased to shift the distribution in a meaningful way. We thus implement particle filter resetting, taking inspiration from sensor resetting localization [37] that combats the kidnapped robot problem, where the robot has been moved without being told and must reinitialize its localization. A reset triggers when the weights of the particles, after accounting for $p(\mathbf{x_t}|y_t)$ and before weight normalization, drop below a threshold (Algorithm 1). We uniformly distribute a set number of particles into the consistent half-space and again rely on

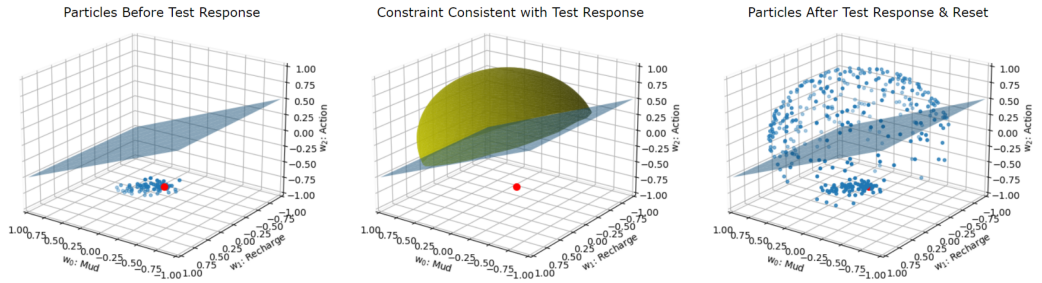


Fig. A1. When a test response is heavily inconsistent with the current model of human beliefs, we perform a reset. The constraint consistent with the test response is shown in all panels, with the consistent side shown with the uniform distribution as a yellow dome in the center panel. The robot reward function is shown as a red dot.

KLD-resampling [40] to obtain the number of particles that will bound the KL divergence between the posterior distribution following the reset and its sample-based maximum likelihood estimate. We then sample that the number of particles directly from the custom distribution corresponding to $p(\mathbf{x_t}|y_t)$ and add it to the particle filter (Figure A1).

### A.3 User Engagement Questions

We adapted the User Engagement Scale short form [48] to ask six questions targeting focused attention:

—"I was fully engaged with learning the game strategy."
—"The time I spent learning the game strategy passed by quickly."
—"I was absorbed in this experience."

and measure perceived usability:

—"I felt frustrated while learning the game strategy."
—"I found learning the game strategy confusing."
—"Learning the game strategy was taxing."

each answered with a 5-point Likert scale in the two studies described in the article.

Received 22 May 2024; revised 15 January 2025; accepted 21 March 2025