Closed-loop Reasoning about Counterfactuals to Improve Policy Transparency

Michael S. Lee [1] **Henny Admoni** [1] **Reid Simmons** [1]

Abstract

Explanations are a powerful way of increasing the transparency of complex AI policies. Such explanations must not only be informative regarding the policy in question, but must also be tailored to the human explainee. In particular, it is critical to consider the explainee's current beliefs and the counterfactuals (i.e. alternate outcomes) with which they will likely interpret any given explanation. E.g., the explainee will be inclined to wonder "why did event P happen instead of counterfactual Q?" To address this, we first model huwman beliefs using a particle filter to consider the counterfactuals the human will likely use to interpret a potential explanation, which in turn helps select an explanation that is highly informative. Second, we design a closed-loop explanation framework, inspired by the education literature, that continuously updates the particle filter not only based on the explanations provided but also based on feedback from the human regarding their understanding. Finally, we present a user study design for testing the proposed closed-loop explanation framework and its ability to improve human understanding of AI policies.

1. Introduction

Much progress has been made in obtaining complex and capable policies through reinforcement learning (e.g. for learning conversational agents (Christiano et al., 2017), recommender systems (Afsar et al., 2022), and robot policies (Brohan et al., 2023)). Ensuring the transparency (i.e. understandability and predictability (Endsley, 2017)) of these policies in all possible scenarios is key to calibrating the understanding of developers and end-users toward proper usage; however, this remains a challenge (Wells & Bednarz, 2021).

[1]Robotics Institute, Carnegie Mellon University, Pittsburgh, PA USA. Correspondence to: Michael S. Lee <ml5@andrew.cmu.edu>.

ICML (International Conference on Machine Learning) Workshop on Counterfactuals in Minds and Machines, Honolulu, Hawaii, USA.

In this work, we seek to increase transparency through explanations, selected through the *machine teaching* paradigm (Zhu, 2015) that selects the minimal set of examples (e.g. demonstrations) that will help a student (e.g. robot end-user) comprehend a concept (e.g. a policy) given their learning model. Importantly, the robot's explanations must not only be informative with respect to a particular policy but must also be tailored to a particular explainee (i.e. student).

Miller (2019) highlights that human explanations are inherently contrastive with respect to a specific counterfactual case, "presented relative to the explainer's beliefs about the explainee's beliefs." Ehsan et al. (2021) similarly notes that an "explanation is only explanatory if it can be consumed by the recipient." That is, the interaction must be socially grounded so that the explanation is understandable. Thus, our key idea is to model the counterfactuals likely to be considered for a particular explanation while *simultaneously leveraging insights from the education literature on effective teaching* to ensure their understandability.

We preview the importance of both counterfactual and educational considerations through a concept known as the zone of proximal development or "Goldilocks zone" (Hattie & Clarke, 2018; Vygotsky, 1980), which suggests that the examples provided to the learner should not be too easy and not too difficult given their current beliefs. For instance, we observed in our prior work (Lee et al., 2021) that humans struggled to understand highly informative examples right from the outset, as their beliefs were unlikely to give rise to the nuanced counterfactuals necessary for correctly interpreting the example.

To illustrate the efficacy of leveraging counterfactual reasoning and insights from the education literature, consider a robot that aims to make its reward function and subsequent policy more transparent to a human using demonstrations (i.e. examples), tests, and feedback (Fig. 1) according to our proposed closed-loop teaching framework (Fig. 2). The robot's objective is to deliver a package to the destination, whose reward function balances traveling through difficult terrain, like mud, and reducing the overall number of actions it takes (i.e. steps). To convey its reward function, imagine the robot first provides a human with the demonstration in Fig. 1a. Because the robot takes a two-action detour to avoid the mud instead of going through it (a natural coun-

terfactual), the human may infer that the robot associates a negative reward with going through mud.

After providing the first demonstration (i.e. the two-step move around the mud), the robot considers what to demonstrate next to convey more information regarding its reward function. Importantly, it knows that the human likely knows that mud is costly from the first demonstration, but does not know *how* costly. For instance, the human may counterfactually believe that the robot would take a four-action detour when faced with two mud patches (Fig. 1b). However, the robot knows that its ratio of mud to action reward is -3 to -1 and that consequently, it would simply go through the mud in Fig. 1b to maximize its reward. Seeing how its direct path meaningfully differs from the human's likely counterfactual of the sizeable detour, the robot considers this to be an informative next demonstration to provide the human with a lower bound on the cost of mud. Furthermore, we highlight that this demonstration aims for the "Goldilocks zone" as it provides a meaningful yet limited update to the human belief through one additional unit of information that further bounds the cost of the mud.

After two demonstrations have been provided, the *testing effect* (Roediger III & Karpicke, 2006) in the education literature suggests dedicating a portion of the teaching budget on testing to increase student learning. Thus, the robot can provide the human with a diagnostic test that will reveal the accuracy or drift of the robot's current model of the human's beliefs. If the human answers incorrectly, then the robot may provide feedback, a remedial demonstration, and remedial tests until the human demonstrates concept mastery. Importantly, the robot must continue to update its model of the human's beliefs throughout the remedial interactions to ensure that it can consider the right counterfactuals when selecting the next series of demonstrations.

The above interaction demonstrates the importance of 1) a calibrated model of the human's beliefs and providing informative explanations that contrast with the human's likely counterfactuals and 2) the benefits of a closed-loop interaction to ensure that explanations are in fact understood and are within the zone of proximal development.

Our contributions are as follows. First, 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 (i.e. explanation) that could be provided. Second, a closed-loop teaching framework based on insights from the education literature that provides demonstrations, tests, and feedback while continuously updating the model of human beliefs. Third, a plan for a user study to test the proposed closed-loop teaching framework and reasoning over counterfactuals for generating informative and understandable demonstrations.

2. Related Work

Example-based Counterfactual Explanations: Example-based explanations represent a class of methods that have long been studied to aid transparency, e.g. by providing prototypical examples that can summarize a dataset (Bien & Tibshirani, 2011). In line with Miller (2019)'s affirmation that effective explanations are inherently contrastive, Kim et al. (2016) showed that presenting not only prototypes but also criticisms (i.e. representative examples that deviate from the prototypes) was especially helpful in aiding human understanding of a dataset distribution.

As models for decision making get larger and more opaque, there has been significant interest in utilizing counterfactual explanations for greater transparency of decision systems. While the vast majority of these methods explore how classification outcomes may have changed if the input had been different (Verma et al., 2020), other methods explore how changes in the model parameters themselves may lead to different outcomes (Bui et al., 2022). Counterfactual explanations have also been utilized in other domains, the closest to our work being in planning (Stepin et al., 2021). While Sukkerd et al. (2020) and Sreedharan et al. (2018) both consider providing explanations based on counterfactuals, the former focuses solely on the tradeoffs in quality attributes of the plan (e.g., execution time, energy consumption, etc) and does not model the human explainee's beliefs, and the latter provides explanations in the form of propositions and predicates that deviate from the human's counterfactual plan rather than providing examples as explanations.

Policy Summarization: Policy summarization aims to provide a global understanding of a policy to a user through example state-action pairs (Amir et al., 2019), which can aid in transparency. The first approach relies on heuristics such as entropy or differences in Q-values to select states and actions to show (Huang et al., 2018; Amir & Amir, 2018).

We instead build on the second approach based on machine teaching (Zhu et al., 2018). Machine teaching aims to teach a target model (e.g. reward function) to a learner with a given learning model (e.g. inverse reinforcement learning or IRL) using a minimal set of teaching examples (e.g. demonstrations). Methods for conveying a robot's reward function and/or behavior to humans are surveyed by Sanneman et al. (Sanneman & Shah, 2022) and Booth et al. (Booth et al., 2022) and we summarize a few relevant works below.

Brown and Niekum (Brown & Niekum, 2019) proposed the Set Cover Optimal Teaching (SCOT) algorithm for selecting demonstrations that provide the tightest constraints on a target reward function for a pure IRL learner. However, human learning is more multi-faceted and our prior work Lee et al. (2021) tailored SCOT for humans by incorporating human learning techniques such as scaffolding. Our ini-

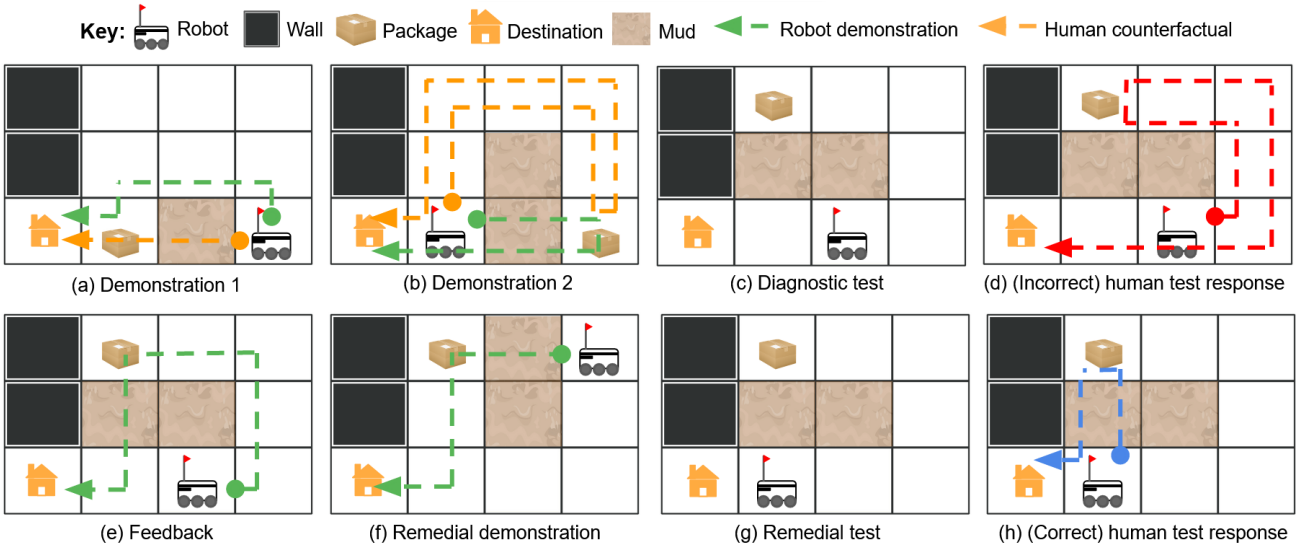

Figure 1. A sample teaching sequence for a batch of knowledge components on the cost of mud. **(a)** A robot's demonstration (green) is shown in contrast to a counterfactual alternative likely considered by a human (orange), which conveys that mud is costly. **(b)** The robot's demonstration lowerbounds the cost of the mud. **(c)** The human is asked to predict the robot's behavior in a test. **(d)** An incorrect response indicates that the demonstration was not understood. **(e)** The human is given the correct response as feedback. **(f)** A remedial demonstration is provided to target the misunderstanding. **(g)** The human is asked to answer the remedial test. **(h)** A correct answer indicates that the human understood the explanation.

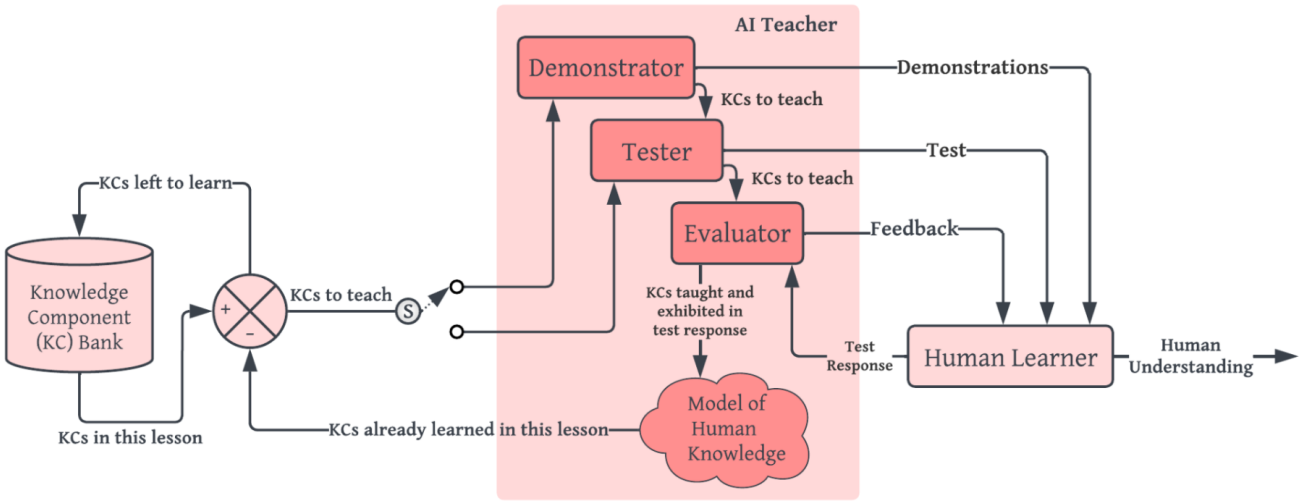

Figure 2. Proposed closed-loop teaching framework. A batch of related knowledge components (KCs) is passed to the AI explainer. The `demonstrator` generates demonstrations that explain the KC, the `tester` provides a test, and the `evaluator` analyzes the test response to see which KCs were understood. If the test was incorrectly answered, the `evaluator` provides the correct answer as feedback, before updating the model of human knowledge. The KCs contained in the human model are compared to the KCs to be understood. If there is a difference and there remain KCs to explain in this batch, then a remedial demonstration and a remedial test is provided. If the remedial test is answered incorrectly, feedback is provided and the switch (labeled 'S') flips such that only tests and feedback are provided until the remaining KCs in this batch are understood. Once all of the KCs in this batch are understood, the switch flip upward again (to also provide demonstrations) and a fresh batch of KCs is pulled from the KC bank.

tial method of scaffolding via IRL did not yield significant learning gains, which we improved upon by incorporating counterfactuals based on the human's beliefs regarding the robot's reward function (Lee et al., 2022). However, this method models the human learner as using exact IRL (Ng & Russell, 2000), which is unable to gracefully handle conflicting information (e.g. knowledge that was assumed to be learned but fails to be demonstrated during testing). Fur-

thermore, we only utilized tests for assessment after having provided demonstrations. We build on this by proposing a Bayesian model of human beliefs in the form of a particle filter and also utilize testing in between demonstrations to iteratively update the human model of beliefs and better ensure understandability.

Finally, we note that Huang et al. (2019) use Bayesian (Ramachandran & Amir, 2007) and MaxEnt (Ziebart et al., 2008) IRL as alternate models of how humans may recover the reward function underlying selected demonstrations. Though these models can incorporate the learner's beliefs (and subsequent counterfactuals) via beliefs initially sampled from a prior over the robot's reward function, such beliefs are not resampled with additional demonstrations, making these IRL methods sensitive to the initial sampling and perhaps leading to slower convergence to the robot's reward function. We instead allow for resampling (Li et al., 2013) within our particle filter model to more efficiently use samples in approximating the posterior distribution of human beliefs.

3. Technical Background

In considering which demonstrations to provide to convey its reward function, the robot assumes that the human uses IRL-like reasoning (Jara-Ettinger, 2019) to infer the reward function underlying the demonstrations, which they can use to deduce the corresponding policy for using planning (Shteingart & Loewenstein, 2014). This section details the technical background necessary for selecting informative demonstrations for a learner using IRL to infer a reward function underlying demonstrations.

Markov decision process: The robot models its world as an instance (indexed by i) of a Markov decision process, MDP_i, comprised of sets of states $\mathcal{S}_i$ and actions $\mathcal{A}$, a transition function T_i, reward function R, discount factor γ, and 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 of their states. An optimal trajectory ξ^* is a sequence of (s_i, a, s_i') tuples obtained by following the robot's optimal policy π^*. Following prior work (Abbeel & Ng, 2004), $R = {\mathbf{w}^*}^\top \phi(s, a, s')$ is represented as a weighted linear combination of reward features. Finally, we assume 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 γ 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 (Figs. 1a and 1b). Thus through IRL, all demonstrations within a domain will support inference over a common $\mathbf{w}^*$. We simplify the notation such that π^* refers to any optimal policy within a domain, and ξ^* refers to a demonstration (dropping the corresponding MDP).

Machine teaching for policies: Our objective to select informative demonstrations for conveying π^* is captured by the machine teaching framework for policies (Lage et al., 2019). We aim to select a set of demonstrations $\mathcal{D}$ of size n that maximizes the similarity ρ between optimal policy π^* and the policy $\hat{\pi}$ recovered using a computational model $\mathcal{M}$ (e.g., IRL) on $\mathcal{D}$

$$\arg\max_{\mathcal{D} \subset \Xi} \rho(\hat{\pi}(\mathcal{D}, \mathcal{M}), \pi^*) \;\text{ s.t. }\; |\mathcal{D}| = n \tag{1}$$

where Ξ is the set of all demonstrations of π^* in a domain. Once $\mathbf{w}^*$ is approximated through IRL, this approach assumes that the learner is able to deduce π^* by planning on the underlying MDP. Thus, the objective reduces to selecting demonstrations that are informative at conveying $\mathbf{w}^*$, which can be measured using behavior equivalence classes.

Behavior equivalence class: The *behavior equivalence class* (BEC) of a demonstration is the region 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 ξ of π is defined as the intersection of half-spaces (Brown & Niekum, 2019) formed by the exact IRL equation (Ng & Russell, 2000)

$$\text{BEC}(\xi|\pi) := \mathbf{w}^\top \left(\mu_\pi^{(s,a)} - \mu_\pi^{(s,b)}\right) \geq 0, \forall (s,a) \in \xi, b \in \mathcal{A}. \tag{2}$$

where $\mu_\pi^{(s,a)} = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \phi\left(s_t\right) \mid \pi, s_0 = s, a_0 = a\right]$ is the vector of reward feature counts accrued from taking the action a in s, then following π after. Any demonstration can be converted into a set of constraints on $\mathbf{w}$ using (2). Importantly, each constraint can be considered as a *knowledge component* (Koedinger et al., 2012) which captures a characteristic of the reward function (e.g. the tradeoffs between the underlying reward features).

Consider again the delivery domain, which has binary reward features $\phi = [$*traversed mud*, *battery recharged*, *action taken*$]$, $\mathbf{w}^* \propto [-3, 3.5, -1]$. In practice, we require $||\mathbf{w}^*||_2 = 1$ to bypass both the scale invariance of IRL and the degenerate all-zero reward function. If no prior knowledge is assumed, the potential reward weights in the human's mind would uniformly span the full surface of the $n-1$ sphere due to the L^2 norm constraint on $\mathbf{w}^*$, where n is the number of domain features. We instead assume that human begins with a prior that action weight is negative (e.g. a bias to take the shortest path). The demonstration in Fig. 3b yields the constraint (or knowledge concept) in Fig. 3c, which indicates that $w_0^* \leq 2w_2^*$ (i.e. mud must be at least twice as costly as an action), since two actions were taken to detour around the mud rather than counterfactually going through it.

4. Methods

The running example of the delivery robot in Section 1 highlights the importance of maintaining an accurate model of human beliefs and likely counterfactuals when deciding on an informative demonstration (e.g. the robot selects the second demonstration of going through the two mud in Fig. 1 because it believes the human's current beliefs would lead them to counterfactually expect the robot to go around the mud).

In this section, we propose a particle filter-based model of human beliefs that is amenable to iterative Bayesian updates and sampling for counterfactual reasoning. We then leverage this model in a closed-loop teaching framework that leverages insights from the education literature to ensure that the demonstrations are understood.

4.1. Particle Filter Human Model

Though we previously modeled the human as an exact IRL learner (Lee et al., 2022), this choice falls short for two reasons. First, people are more likely to perform approximate, rather than exact, inference (Huang et al., 2019). Second, a model of human beliefs solely comprised of half-spaces cannot handle conflicts that arise when the human incorrectly applies a knowledge component during testing that was assumed learned during teaching (as you cannot reconcile two identical half-space constraints that point in opposite directions).

We thus move to a probabilistic human model in the form of a particle filter (Doucet et al., 2009). Each particle represents a potential human belief regarding the robot's reward function, and particle weights are updated in a Bayesian fashion based on constraints conveyed through teaching demonstrations and test responses. *Leveraging both constraints and Bayesian updates gracefully affords both reasoning over discrete KCs (e.g. a lesson comprises a batch of related KCs) and a probabilistic modeling of human understanding that is amenable to iterative updates during teaching and testing.* The particle filter routines outlined the following sections come together in Alg. 1.

4.1.1. Updating Particle Positions and Weights

Assume a set of particles, defined by their positions and associated weights $\{\mathbf{x}_t, \check{\mathbf{w}}_t\}$. Without loss of generality, assume that a demonstration or test response is provided at each time step t. Each demonstration generates multiple constraints by comparing the demonstration against possible counterfactuals and each incorrectly answered test will generate a single constraint by comparing the true test answer against the incorrect answer, both through Eq. 2. Each constraint y_t can be translated into a probability distribution $p(x_t|y_t)$ that can be used to update the weights of each particle.

We propose a custom probability distribution $p(x_t|y_t)$ for each constraint as a combination of the uniform distribution that aligns with the correct half-space of the constraint and a Von-Mises Fisher distribution that aligns with the incorrect half-space (Fig. 4). The uniform distribution captures the notion that any particle lying on the correct half-space is equally valid for that demonstration, whereas the Von-Mises Fisher distribution captures the notion that a particle is exponentially less likely to have generated that demonstration as you move away from the constraint.

Finally, we address common challenges to using particle filters in practice. Sample degeneracy occurs when successive updates to the weights of the particles causes only a few particles to have high weight and fails to model regions of interest in the posterior with sufficient detail (Li et al., 2014). Furthermore, the number of particles (i.e. sample size) should adapt to the complexity of the distribution being modeled (Straka & Ŝimandl, 2009). To address both concerns, we rely on KLD-resampling (Li et al., 2013) to obtain the sample size that bounds the Kullback-Leibler divergence between the sample-based maximum likelihood estimate and the true posterior distribution, and simultaneously rely on systematic resampling 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 only resample when the effective sample size (a measure of sample degeneracy) drops below a predefined threshold and also add Gaussian noise when resampling the particles (Li et al., 2014).

4.1.2. Resetting the Particle Filter

The particle filter may converge then suddenly obtain new information that is not very consistent with the current distribution (see Fig. 6). In this case, the filter will struggle to update as none or very few of the particles weights would be increased to shift the distribution in a meaningful way. We thus implement particle filter resetting, taking inspiration from sensor resetting localization (Lenser & Veloso, 2000; Coltin & Veloso, 2013) that combats the kidnapped robot problem, where the robot has been moved without being told and must reinitialize its localization. Our particle filter resetting triggers when the weights of the particles after accounting for $p(x_t|y_t)$ and before weight normalization (line 11 of Alg. 1) drops below a threshold. We uniformly distribute a set number of particles into the correct half-space (Fig. 6b) and again rely on KLD-resampling (Li et al., 2013) to obtain the number of particles that will bound the Kullback-Leibler divergence between the posterior distribution following the reset and its sample-based maximum

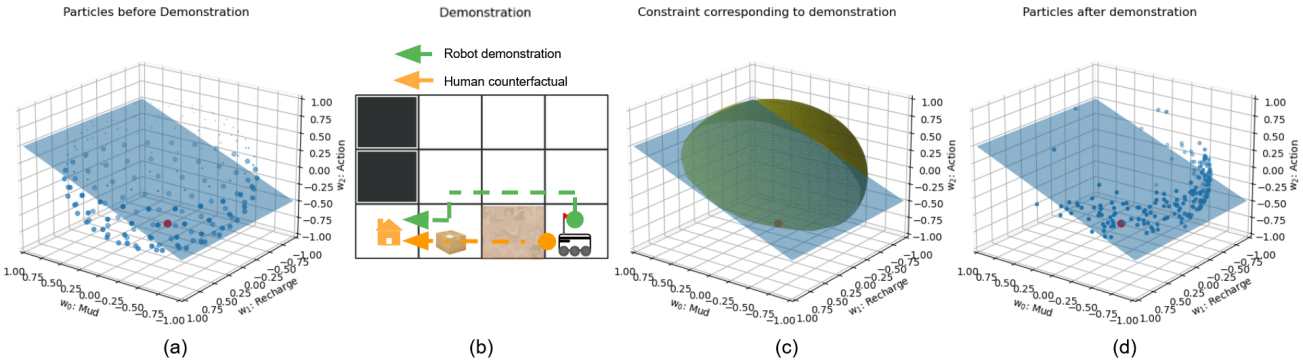

Figure 3. Example sequence on how a demonstration updates a particle filter model of human beliefs. Ground truth reward of the robot is shown as a red dot, and the constraint (or knowledge component) corresponding to the demonstration is shown in all plots for visual reference. **(a)** Particles before demonstration is shown (prior). **(b)** Demonstration shown to human. **(c)** The exact IRL constraint (obtained used Eq. 2) corresponding to the demonstration in (b) that conveys that mud must be at least twice as costly as an action. The likelihood used to update the particle weights will be the custom distribution that combines the uniform and Von-Mises Fisher distributions, whose orientation will be aligned to the exact IRL constraint it approximates (see Fig. 4). **(d)** Particles after demonstration is shown (posterior).

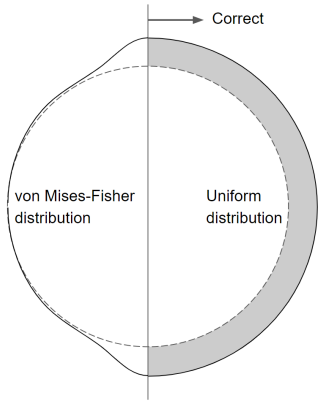

Figure 4. The custom probability density function (pdf) for updating particle weights (see Alg. 1) based on a constraint generated from a demonstration (Eq. 2) shown in 2D. The right side of the constraint consists of the uniform distribution as any particle on the correct side is equally likely. The left side of the constraint consists of the Von-Mises Fisher distribution that represents an exponential fall-off of likelihood.

likelihood estimate. We then sample that number of particles directly from the custom distribution corresponding to $p(x_t|y_t)$ and add it to the particle filter. An example particle filter resetting procedure is shown in Fig. 6.

4.1.3. Sampling Human Beliefs

Given a running particle filter model, we may sample human beliefs in order to do counterfactual reasoning. We first run systematic resampling on the particles to downselect to a candidate set, accounting for the differences in the weights of the particles and favoring those that are higher weighted. We then rely on the 2-approximation algorithm for the k-center problem (Hochbaum & Shmoys, 1985) to greedily select k samples that are spread out such that the maximum distance from any particle in the candidate set to one of the k samples is minimized (see Fig. 5).

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 $\check{w}_t^{(i)} = \check{w}_{t-1}^{(i)} \cdot p(x_t^{(i)}|y_t)$
  end for
  if $\sum_{j=1}^{N} \check{w}_t^{(j)} < \check{w}_{threshold}$ **then**
    // Particle filter has degraded
    Perform a particle filter reset ▷ Section 4.1.2
  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 particles $x_t^{(i)}$ with probabilities $\tilde{w}_t^{(i)}$ using KLD resampling
  end if
end for

4.2. Closed-loop Teaching

With a particle filter model of human beliefs that is amenable to iterative updates via demonstrations and tests, we now formulate a closed-loop teaching framework for conveying a robot's reward function to a human. As we walk through the framework that is visualized in Fig. 2, we highlight the principles from the education literature that guide the design. A rollout of a sample teaching sequence is shown in Fig. 1, which may serve as a visual correspondence to the high-level overview of the sequence is provided in Alg. 2.

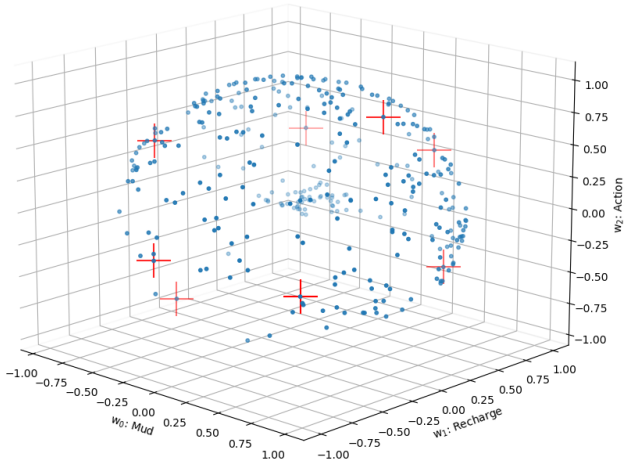

Figure 5. Human counterfactuals are generated by sampling beliefs from the particle filter model. As nearby particles are likely to generate similar counterfactuals, we rely on the 2-approximation algorithm for the k-center problem to sample k beliefs (marked by red crosses) that are spread out.

We first leverage feature and counterfactual scaffolding from our prior work (Lee et al., 2022) to select knowledge components (KCs) that incrementally increase in information across an increasing subset of features (e.g. bounds on the cost of the mud given the step cost, bounds on the reward of recharging given the step cost, then tradeoffs between all three – e.g. is it worth going through two steps and mud to recharge?).

We begin the loop by taking a single batch of related KCs that define a *lesson* (e.g. bounds on the cost of the mud given the step cost) and providing it to the `demonstrator` (see Fig. 2) to select demonstrations that best convey these KCs. Specifically we utilize counterfactual reasoning (Lee et al., 2022) to select demonstrations that are informative with respect to the counterfactuals likely considered by the human[1]. We simultaneously leverage the educational principles of the *zone of proximal development* or the *"Goldilocks" zone* (Hattie & Clarke, 2018; Vygotsky, 1980) to provide a sequence of demonstrations that provide information incrementally, e.g. demonstrations that convey one new constraint at a time (such as an upper bound then a lower bound on mud cost).

After the demonstrations have been provided, the `tester` selects diagnostic tests that will verify whether or not the human has learned the lesson, such that the correct responses will require knowledge of the corresponding KCs. This is motivated by the educational principle of the *testing effect* (Roediger III & Karpicke, 2006), which suggests that learning outcomes are increased when a portion of the teaching budget is devoted to testing the student. These diagnostic tests will also aim toward visual dissimilarity from the teaching demonstrations and be visually complex (Lee et al., 2022) to challenge the learner and test their understanding.

If the `evaluator` notices that the human answered the diagnostic test incorrectly, then it will provide immediate *feedback* to the human on how their answer differed from the correct one, inspired by the findings that immediate feedback on errors lead tp more efficient learning and better learning outcomes (Corbett & Anderson, 2001; Koedinger et al., 2013). The `evaluator` will also update the particle filter model of the human's knowledge (i.e. beliefs) given the diagnostic test response.

If the KCs in this lesson and the KCs exhibited in the test response match, then there are no KCs left to learn, and KCs for the next lesson are provided by the KC bank. If there are still KCs left to teach, then the `demonstrator` will provide a remedial demonstration that conveys that KC with visual simplicity (i.e. without any distracting visual clutter) (Lee et al., 2021). Note that we utilize the particle filter model to consider the counterfactuals the human is likely to consider for each potential demonstration in order to select the one that conveys the missed KC. Then the `tester` will again provide a remedial test with visual complexity, and the `evaluator` will again analyze the test response and update the human model.

If the human gets the remedial test wrong, the switch in Fig. 2 (labeled 'S') flips and the `tester` and `evaluator` will continue to provide only remedial tests and corresponding feedback (but no additional demonstrations) until the human shows understanding of the KC. This is motivated by the *expertise reversal effect* (Kalyuga, 2009), which, when paired with the testing effect, finds that the learners with increased expertise in a material will benefit more from additional testing (in varied contexts) over additional instruction (Koedinger et al., 2012).

Finally, the human's understanding of the explained policy can be evaluated via their performance on a held-out set of tests in which they predict the policy in unseen scenarios.

5. Proposed user study

We are currently creating an online user study that will explore whether our proposed closed-loop teaching method improves a human's understanding of a robot's policy. The study will involve participants watching robot demonstrations in three deterministic domains and predicting the robot's behavior in new test environments. The participants will be explicitly informed of each domain's reward features, but will have to infer the respective reward weights by watching demonstrations.

[1]These initial demonstrations are selected using an idealized model of the human as an exact IRL learner and deviations are corrected using a particle filter model of the human, remedial demonstrations, and tests in our proposed closed-loop teaching.

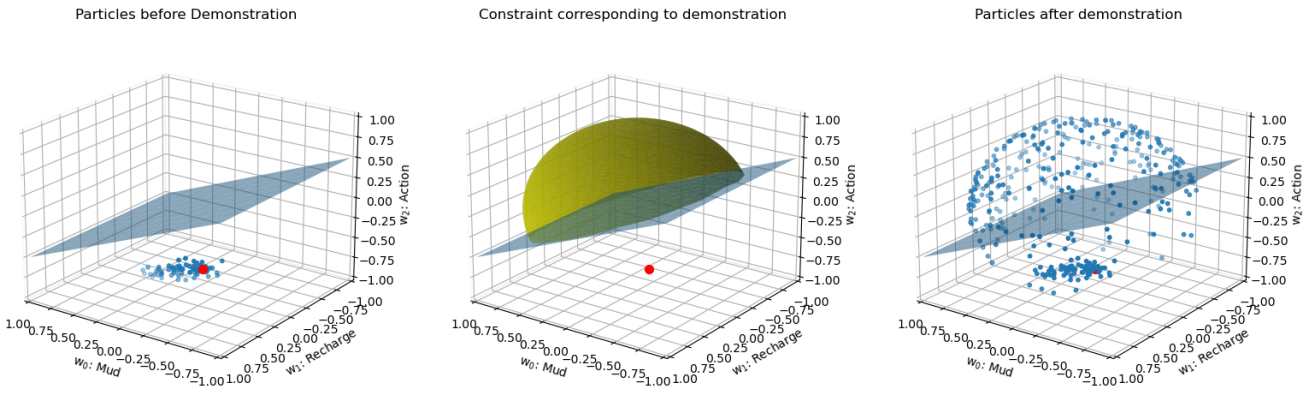

Figure 6. When a test is answered in a way that is heavily inconsistent with the current model of human beliefs, we perform a reset (Section 4.1.2). The constraint consistent with the test response is shown in all panels for reference, with correct side of the constraint visualized via the uniform distribution in the center. Ground truth reward of the robot is shown as a red dot.

Algorithm 2 Closed-loop Teaching Framework

```
Determine batches of knowledge components (KC) using counterfactual scaffolding to form lessons
for For each batch of KCs (i.e. lesson) do
    Provide initial demonstrations
    Provide diagnostic test
    Evaluate diagnostic test response
    if diagnostic test response is incorrect then
        Provide feedback
        Provide remedial demonstrations
        Provide remedial test
        Evaluate remedial test response
        while remedial test response is incorrect do
            Provide remedial test
            Evaluate remedial test response
        end while
    end if
end for
```

The between-subjects variable will be *feedback loop* (open, partial, and full). The open feedback loop will follow our prior work (Lee et al., 2022) in selecting an set of informative demonstrations a priori using counterfactual reasoning that iteratively decrease in BEC area. Partial feedback loop will provide a diagnostic test after each lesson and provide a correction if necessary, while the full feedback loop will also provide a remedial demonstration and remedial tests until correct knowledge of the KC in question is shown by a correct remedial test. The efficacy of the three conditions for teaching the robot's policy will be evaluated by a held-out set of tests at the end of the study (these tests will be pulled directly from our prior work (Lee et al., 2022)).

Our hypotheses are that the *full* feedback loop will lead to the best performance on the held-out tests and also be rated most positively in terms of subjective experience. For more details on the domains, measures, and hypotheses, and general study design, please refer to Appendix A.

6. Conclusion

In this paper, we propose a means of leveraging example-based explanations (i.e. demonstrations) to increase the transparency of complex policies. In alignment with prior work, we stress that effective explanations must consider how the explainee will interpret it given their current beliefs, namely what counterfactuals they are likely to use in extracting information from the explanation. Toward our goal of increased transparency, we presented a particle filter-based model of human beliefs that can be used to select informative explanations. With insights from the education literature, we designed a closed-loop explanation framework that not only provides understandable explanations but also incorporates feedback regarding the human's current understanding back into the robot's model of the human's beliefs. Finally, we presented a user study design for testing our proposed particle filter model and closed-loop explanation framework.

Although we explored counterfactually-informed explanations in the context of robotics, this work is also applicable to explaining policies for sequential decision making more broadly. For example, Ernst et al. (2006) proposed a reinforcement learning-based treatment policy for HIV based on clinical data. Prasad et al. (2019) proposed a low dimensional version of the HIV domain based on a reward function that is a linear combination of three reward features – a direct analog of the delivery domain used in this work.

References

Abbeel, P. and Ng, A. Y. Apprenticeship learning via inverse reinforcement learning. In *ICML*, 2004.

Afsar, M. M., Crump, T., and Far, B. Reinforcement learn-

ing based recommender systems: A survey. *ACM Computing Surveys*, 55(7):1–38, 2022.

Amir, D. and Amir, O. Highlights: Summarizing agent behavior to people. In *AAMAS*, 2018.

Amir, O., Doshi-Velez, F., and Sarne, D. Summarizing agent strategies. *Autonomous Agents and Multi-Agent Systems*, Sep 2019. ISSN 1573-7454.

Bien, J. and Tibshirani, R. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, pp. 2403–2424, 2011.

Booth, S., Sharma, S., Chung, S., Shah, J., and Glassman, E. L. Revisiting human-robot teaching and learning through the lens of human concept learning. In *HRI*, 2022.

Brohan, A., Chebotar, Y., Finn, C., Hausman, K., Herzog, A., Ho, D., Ibarz, J., Irpan, A., Jang, E., Julian, R., et al. Do as i can, not as i say: Grounding language in robotic affordances. In *Conference on Robot Learning*, pp. 287–318. PMLR, 2023.

Brown, D. S. and Niekum, S. Machine teaching for inverse reinforcement learning: Algorithms and applications. In *AAAI*, 2019.

Bui, N., Nguyen, D., and Nguyen, V. A. Counterfactual plans under distributional ambiguity. In *International Conference on Learning Representations*, 2022.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Coltin, B. and Veloso, M. Multi-observation sensor resetting localization with ambiguous landmarks. *Autonomous robots*, 35:221–237, 2013.

Corbett, A. T. and Anderson, J. R. Locus of feedback control in computer-based tutoring: Impact on learning rate, achievement and attitudes. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 245–252, 2001.

Doucet, A., Johansen, A. M., et al. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704):3, 2009.

Ehsan, U., Liao, Q. V., Muller, M., Riedl, M. O., and Weisz, J. D. Expanding explainability: Towards social transparency in ai systems. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–19, 2021.

Endsley, M. R. From here to autonomy: lessons learned from human–automation research. *Human factors*, 59(1): 5–27, 2017.

Ernst, D., Stan, G.-B., Goncalves, J., and Wehenkel, L. Clinical data based optimal sti strategies for hiv: a reinforcement learning approach. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 667–672. IEEE, 2006.

Hattie, J. and Clarke, S. *Visible learning: feedback*. Routledge, 2018.

Hochbaum, D. S. and Shmoys, D. B. A best possible heuristic for the k-center problem. *Mathematics of operations research*, 10(2):180–184, 1985.

Huang, S. H., Bhatia, K., Abbeel, P., and Dragan, A. D. Establishing appropriate trust via critical states. In *IROS*, 2018.

Huang, S. H., Held, D., Abbeel, P., and Dragan, A. D. Enabling robots to communicate their objectives. *Autonomous Robots*, 2019.

Jara-Ettinger, J. Theory of mind as inverse reinforcement learning. *Current Opinion in Behavioral Sciences*, 29: 105–110, 2019.

Kalyuga, S. The expertise reversal effect. In *Managing cognitive load in adaptive multimedia learning*, pp. 58–80. IGI Global, 2009.

Kim, B., Khanna, R., and Koyejo, O. O. Examples are not enough, learn to criticize! criticism for interpretability. *Advances in neural information processing systems*, 29, 2016.

Koedinger, K. R., Corbett, A. T., and Perfetti, C. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science*, 36(5):757–798, 2012.

Koedinger, K. R., Booth, J. L., and Klahr, D. Instructional complexity and the science to constrain it. *Science*, 342 (6161):935–937, 2013.

Lage, I., Lifschitz, D., Doshi-Velez, F., and Amir, O. Exploring computational user models for agent policy summarization. In *International Joint Conference on Artificial Intelligence*, 2019.

Lee, M. S., Admoni, H., and Simmons, R. Machine teaching for human inverse reinforcement learning. *Frontiers in Robotics and AI*, 8:693050, 2021.

Lee, M. S., Admoni, H., and Simmons, R. Reasoning about counterfactuals to improve human inverse reinforcement learning. In *2022 IEEE/RSJ International Conference on*

Intelligent Robots and Systems (IROS), pp. 9140–9147. IEEE, 2022.

Lenser, S. and Veloso, M. Sensor resetting localization for poorly modelled mobile robots. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 2, pp. 1225–1232. IEEE, 2000.

Li, T., Sun, S., and Sattar, T. P. Adapting sample size in particle filters through kld-resampling. *Electronics Letters*, 49(12):740–742, 2013.

Li, T., Sun, S., Sattar, T. P., and Corchado, J. M. Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches. *Expert Systems with applications*, 41(8):3944–3954, 2014.

Miller, T. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

Ng, A. Y. and Russell, S. Algorithms for inverse reinforcement learning. In *International Conf. on Machine Learning*, 2000.

O'Brien, H. L., Cairns, P., and Hall, M. A practical approach to measuring user engagement with the refined user engagement scale (ues) and new ues short form. *International Journal of Human-Computer Studies*, 112: 28–39, 2018.

Prasad, N., Engelhardt, B. E., and Doshi-Velez, F. Defining admissible rewards for high confidence policy evaluation. *arXiv preprint arXiv:1905.13167*, 2019.

Ramachandran, D. and Amir, E. Bayesian inverse reinforcement learning. In *IJCAI*, volume 7, pp. 2586–2591, 2007.

Roediger III, H. L. and Karpicke, J. D. The power of testing memory: Basic research and implications for educational practice. *Perspectives on psychological science*, 1(3): 181–210, 2006.

Sanneman, L. and Shah, J. A. An empirical study of reward explanations with human-robot interaction applications. *IEEE RA-L*, 2022.

Shteingart, H. and Loewenstein, Y. Reinforcement learning and human behavior. *Current Opinion in Neurobiology*, 25:93–98, 2014.

Sreedharan, S., Srivastava, S., and Kambhampati, S. Hierarchical expertise level modeling for user specific contrastive explanations. In *IJCAI*, pp. 4829–4836, 2018.

Stepin, I., Alonso, J. M., Catala, A., and Pereira-Fariña, M. A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access*, 9:11974–12001, 2021.

Straka, O. and Ŝimandl, M. A survey of sample size adaptation techniques for particle filters. *IFAC Proceedings Volumes*, 42(10):1358–1363, 2009.

Sukkerd, R., Simmons, R., and Garlan, D. Tradeoff-focused contrastive explanation for mdp planning. In *RO-MAN*, 2020.

Verma, S., Boonsanong, V., Hoang, M., Hines, K. E., Dickerson, J. P., and Shah, C. Counterfactual explanations and algorithmic recourses for machine learning: a review. *arXiv preprint arXiv:2010.10596*, 2020.

Vygotsky, L. S. *Mind in society: The development of higher psychological processes*. Harvard university press, 1980.

Wells, L. and Bednarz, T. Explainable ai and reinforcement learning—a systematic review of current approaches and trends. *Frontiers in artificial intelligence*, 4:550030, 2021.

Zhu, X. Machine teaching: an inverse problem to machine learning and an approach toward optimal education. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 4083–4087, 2015.

Zhu, X., Singla, A., Zilles, S., and Rafferty, A. N. An overview of machine teaching. *arXiv preprint arXiv:1801.05927*, 2018.

Ziebart, B. D., Maas, A. L., Bagnell, J. A., Dey, A. K., et al. Maximum entropy inverse reinforcement learning. In *AAAI*, 2008.

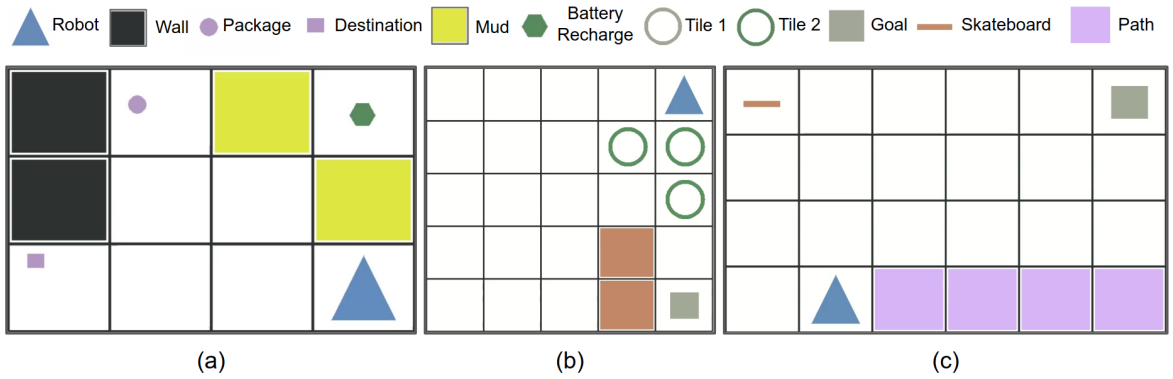

Figure 7. Three domains to be used for the user study, each with a different set of reward weights to infer from demonstrations: **(a)** delivery **(b)** tiles **(c)** skateboard. The semantics of the various objects were hidden using abstract geometric shapes and colors.

A. Additional details on user study design

A.1. Domains

Each of the three domains to be used in the user study will consist of one shared action reward feature (that helps penalize each action), and two unique reward features as follows (see Fig. 7).

Delivery domain. The robot is penalized for moving out of mud and rewarded for recharging.

Tiles domain. The robot is penalized differently for traversing the two differently shaped tiles.

Skateboard domain. The robot is penalized less per action if it has either picked up a skateboard (i.e. riding is less costly than walking) or is traversing through a designated path.

A.2. Expected progression of user study

The user study itself will primarily consist of three trials, with each trial comprising a teaching portion and an assessment portion in a unique domain. During teaching, participants will first be explicitly informed of the reward features of the domain. Then they will infer the corresponding reward weights by watching demonstrations and potentially also taking tests depending on their feedback loop condition and provide subjective ratings on whether each demonstration or test improved their understanding of the robot's policy (M2). After the teaching portion, questions from the User Engagement Scale Short Form (O'Brien et al., 2018) on 'focused attention' and 'perceived usability' will measure how engaged the user was (M3-M4).

During assessment, participants will be tasked with predicting the optimal trajectory in six unseen test environments (a random order of two high, medium, and low difficulty environments each) (M1).

The following measures (M1-M4) will be used to test the hypotheses below (H1-H4).

M1. Optimal response: Participants are assigned a binary score depending on the optimality of their test trajectory.

M2. Improved understanding rating: "Did this [demonstration or test] improve your understanding of game strategy?", answered with a 5-point Likert scale

M3. Focused attention: "I lost myself in this experience./The time I spent learning the game strategy just slipped away./I was absorbed in this experience.", each rated with a 5-point Likert scale

M4. Perceived usability: "I felt frustrated while learning the game strategy./I found learning the game strategy confusing./Learning the game strategy was taxing.", each rated with a 5-point Likert scale

H1: The full feedback loop condition will result in the highest optimal responses and the open feedback loop condition will result in the lowest.

H2: The full feedback loop condition will result in the highest average rating of improved understanding and the open feedback loop condition will result in the lowest.

H3: The full feedback loop condition will result in the highest ratings of 'focused attention' and the open feedback loop condition will result in the lowest.

H4: The full feedback loop condition will result in the highest 'perceived usability' ratings as it is more tailored for the student, and the open feedback loop condition will result in the lowest overall. However, the full feedback loop condition may also be rated to be the most taxing as instruction in the zone of proximal development often leads to high learning gains while requiring mental effort on the part of the student (Lee et al., 2022) (as opposed to instruction that is too easy that will not be taxing but also will not result in any learning).