# Goals vs. Rewards: A Preliminary Comparative Study of Objective Specification Mechanisms

**Septia Rani, Serena Booth, Sarath Sreedharan**

**Keywords:** objective specification, goals, rewards.

## Summary

This paper studies two popular objective specification mechanisms for sequential decision-making problems: goals and rewards. We investigate how easy it is for non-AI experts to use these different specification mechanisms effectively. Namely, we investigate how effectively people can use these mechanisms to (a) correctly direct an AI system or robot to generate some desired behavior and (b) predict the behavior encoded in a given objective specification. We perform a user study to assess these questions. In addition, we present a formalization of the problems of objective specification and behavior prediction, and we characterize *underspecification* and *overspecification*. While participants have a strong preference for using goals as an objective specification mechanism, we find a surprising result: even non-expert users are equally capable of specifying and interpreting reward functions.

## Contribution(s)

1. The paper assesses how well non-expert users can effectively make use of goal and reward specification mechanisms. In particular, we study whether they (a) can use these mechanisms to generate specifications that result in some intended target behavior and (b) can predict behavior that could result from the given specification.
   **Context:** We are unaware of any works that perform such human-centric comparisons. The closest works we know of focus purely on how successful engineers are in hand-crafting reward functions (cf. (Knox et al., 2023; Booth et al., 2023)).
2. We provide a formal definition of the specification and prediction task to support comparisons between reward functions and goals. We also provide a formal characterization of the conditions under which an objective can be said to be overspecified or underspecified.
   **Context:** While there are existing works that have tried to model objective misspecification (e.g., Mechergui & Sreedharan (2024)), underspecification (e.g., Shah et al. (2022)), and misspecification (e.g., Amodei et al. (2016)), these definitions have not been formalized to cover and compare multiple specification modalities.
3. Our results present evidence that the non-expert users' ability to correctly specify and interpret reward functions is comparable to their ability to provide goal specifications. However, we see a clear difference in their preferences between the two metrics: they overwhelmingly prefer the goal mechanism.
   **Context:** We are unaware of any prior works that point to parity in user ability to leverage the two objective specification mechanisms. This result may imply that developing novel interfaces for reward functions could help users of RL techniques to utilize reward functions more effectively. Mechanisms like reward machines are one such promising mechanism (Icarte et al., 2022).

# Goals vs. Rewards: A Preliminary Comparative Study of Objective Specification Mechanisms

**Septia Rani**[1], **Serena Booth**[2,3], **Sarath Sreedharan**[1]
{septia.rani,sarath.sreedharan}@colostate.edu,
serena_booth@brown.edu

[1]**Department of Computer Science, Colorado State University**
[2]**Department of Computer Science, Brown University**
[3]**The Simons Institute for the Theory of Computing, University of California, Berkeley**

## Abstract

This paper studies two popular objective specification mechanisms for sequential decision-making: goals and rewards. We investigate how easy it is for people without AI expertise to use these different specification mechanisms effectively. Specifically, through this paper, we investigate how effectively these mechanisms could be used to (a) correctly direct an AI system or robot to generate some desired behavior and (b) predict the behavior encoded in a given objective specification. We first present a formalization of the problems of objective specification and behavior prediction, and we characterize the problems of underspecification and overspecification. We then perform a user study to assess how well participants are able to use rewards and goals as specification mechanisms, and their propensity for overspecification and underspecification with these mechanisms. While participants have a strong preference for using goals as an objective specification mechanism, we find a surprising result: even non-expert users are equally capable of specifying and interpreting reward functions as of using goals.

## 1 Introduction

We examine the two common specification mechanisms for sequential decision-making: goals and rewards. We assess how well non-AI experts can work with these different specification mechanisms. Goals and rewards have different expected upsides. Goals allow people to provide a partial specification of their desired end states. This mechanism is commonly used in classical planning (Cox, 2016) and has also received a lot of attention from recent work in using Large Language Models (LLMs) (Brown et al., 2020) for robot planning (cf. (Brohan et al., 2023)). Rewards, on the other hand, are the underlying objective specification mechanisms used by reinforcement learning (RL) methods (Sutton & Barto, 2018) and Markov Decision Processes (MDPs). Rewards are a means for encoding goals: the reward hypothesis asserts that “all of what we mean by goals and purposes can be well thought of as maximization of the expected value of the cumulative sum of a received scalar signal [reward]” (Sutton & Barto, 2018). The reward format allows one to associate a scalar signal with reaching some state or performing some action in a given state.

The research community has developed a rigorous understanding of these specification mechanisms' expressiveness and representational limitations (cf. (Abel et al., 2021)). Despite this understanding, the ease with which users can express their underlying objectives in these forms has not, to our knowledge, been explicitly studied. While the development of LLMs has received attention as potentially intuitive interfaces to AI systems, they do not entirely address the question of how to best construct specifications for AI systems, either. After all, LLMs would need to translate the user utterances into the underlying objective specification (whether goals, rewards, or some other form), and it is unclear if these utterances would contain sufficient information for the translation.

In this paper, we conduct a user study to examine the ease of use, strengths, and weaknesses of the two specification mechanisms when used by non-AI experts. In the user study, we expose participants to these objective specification mechanisms in intuitive tasks using simple interfaces and measure (a) how well the users are able to use the specific mechanism correctly and (b) how well they can understand an objective specified using each mechanism. While there have been some efforts at measuring the difficulty in specifying rewards (Booth et al., 2023), to the best of our knowledge, our work represents the first effort to perform such a comparative analysis of the two specification mechanisms among non-AI experts. To ensure our user studies are performed from a firm and formal grounding, we also provide a concrete characterization of the tasks related to objective specification and behavior prediction given an objective. Additionally, we provide a characterization for when a given objective could be said to be overspecified or underspecified.

The two primary takeaways from our study results are as follows. First, non-expert users are not as bad at specifying reward functions as is generally assumed, and in fact, their ability to do so is comparable to their ability to correctly specify goals. This is a surprising result, as goals are a seemingly more intuitive mechanism and are more commonly represented in everyday communications. Second, despite their ability to use rewards as specifications, users generally perceive goal specification to be more intuitive and easier to specify. We believe that the results from this study could help us design objective specification interfaces that are more intuitive and easier to use for everyday users.

The paper is structured as follows: Section 2 discusses the related works. Section 3 provides a brief discussion of goals and rewards as an objective specification mechanism and potential trade-offs. Section 4 describes the formal definition of specification and prediction. We describe the specific hypotheses we focus on in Section 5. Section 6 discusses the methods, including the study design. Section 7 presents the results and discussions. Finally, the conclusion is described in Section 8.

## 2 Related Work

The notion that goals are a natural way people think about their objectives has a long history. One could see similar ideas being discussed, Aristotle's notions of phronēsis (Taylor, 2019) to means-end analysis (Simon, 2019). Apart from evidence that people may leverage some notions of goals in their own reasoning, there have been fewer studies performed in determining if goals are, in fact, the best mechanisms for people to actually specify their objectives. Some works within this space include proposals to compare how effectively people can specify their objectives in procedural terms, i.e., in terms of actions or sequence of actions, as opposed to the end goal (Tran, 2024).

In the reward space, reinforcement learning often assumes the existence of a divined reward function that encodes the task. In practice, though, correctly specifying reward functions is nontrivial: the challenge of doing so correctly has catalyzed the take-off of the AI safety research community (Amodei et al., 2016; Russell, 2022). Further, reward functions are typically designed by engineers through trial-and-error design processes (Knox et al., 2023), which are subject to oversights and inaccuracies, even when crafted by reinforcement learning experts (Booth et al., 2023).

Because of the challenges of using either goals or rewards as specifications, efforts in human-computer interaction, broadly construed, have sought to use intuitive signals in place of these explicit specification modalities. These alternatives span feedback (Knox & Stone, 2009; MacGlashan et al., 2017), corrections (Losey & O'Malley, 2018; Bajcsy et al., 2018), advice (Thomaz & Breazeal, 2008; Amershi et al., 2014), demonstrations (Ravichandar et al., 2020), dynamical system modulation matrices (Figueroa et al., 2020), and, most famously, preferences (Christiano et al., 2017; Ziegler et al., 2019; Biyik & Sadigh, 2018). While these intuitive mechanisms unlock non-expert users' ability to program machines, their interpretation is subject to failures and misinterpretation since the human providing the specification has less control over how the system interprets their specification. For example, a line of research has questioned the inductive bias used in reinforcement learning from human preference (Knox et al., 2022).

## 3 Background

We will start by providing a brief sketch of the two specification mechanisms under consideration: goals and rewards. Since we primarily focus on sequential decision-making settings, for each problem, we will separate out the task domain from the objective specification. In each case, the task domain will provide the details on the dynamics of the task and the starting state of the environment.

To start with, goals as an objective specification mechanism is most commonly used in deterministic factored planning settings, also referred to as "*classical planning*" settings (Geffner & Bonet, 2013). In general, a classical planning problem can be represented by a tuple of the form $\mathcal{P}^c = \langle \mathcal{D}^c, \mathcal{G}^c \rangle$, where $\mathcal{D}^c$ is the task domain and $\mathcal{G}^c$ is the goal specification. Here we use the superscript '$c$' to denote the fact that the model and the components are part of a classical planning model. The task domain is further defined as $\mathcal{D}^c = \langle F^c, A^c, I^c \rangle$, where $F^c$ is a set of proposition variables or facts used to define the state space, $A^c$ is the set of actions and $I^c$ is the initial state. Each action $a \in A^c$, is further defined by a tuple of the form $a = \langle pre(a), add(a), del(a) \rangle$. Here $pre(a) \subseteq$ is the preconditions that need to be satisfied for the action $a$ to be executable, $add(a)$ and $del(a)$ are add and delete effects, respectively. The result of executing an action $a$ in state $s$, is captured by the transition function $\Gamma^c$, and is given as:

$$\Gamma^c(s,a) = \begin{cases} (s \setminus del(a)) \cup add(a) \text{ if } pre(a) \subseteq s \\ \textit{Undefined} \text{ otherwise} \end{cases}$$

We will also overload the notation and use $\Gamma^c$ to denote the execution of action sequences. A solution to a classical planning problem takes the form of an action sequence whose execution in the initial state results in a state that satisfies the goal specification. Such an action sequence is referred to as a plan. More formally, an action sequence $\pi = \langle a_1, ..., a_k \rangle$ is a plan if $\Gamma^c(I^c, \pi) \supseteq \mathcal{G}^c$. In the simplest formalism, an optimal plan corresponds to the shortest possible plan, i.e., this plan contains the least number of steps[1].

Reward functions are defined in the context of a Markov Decision Process or MDP (Puterman, 1990). Here, an MDP will be defined using a tuple of the form $\mathcal{P}^m = \langle \mathcal{D}^m, \mathcal{R}^m \rangle$. As with the previous planning formalism, $\mathcal{D}^m$ stands for the domain, but our objective is now given by a reward function $\mathcal{R}^m$. Here, we use the superscript '$m$' to denote the fact that this is modeling an MDP. In this case, the domain is given by a tuple of the form $\langle F^m, A^m, I^m, T^m, \gamma \rangle$, now as before $F^m$ stands for the state variable and $I^m$ the initial state. Here, $A^m$ only lists the action labels, and the dynamics of the action are determined completely by the transition probability function $T^m$. Finally, $\gamma \in [0, 1)$ represents the discount factor that determines how the agent maximizes cumulative discounted future rewards or returns. Here, we will also have a slightly different state space. Specifically, we will define it as $S^m = 2^F \cup \{\bot\}$. Here, we add the new state $\bot$ as a stand-in for the end state. Now, the transition function will be given as

$$T^m : S^m \times A^m \times S^m \rightarrow \{0, 1\}$$

Here, the mapping is only to probabilities 0 and 1 since we focus on problems with deterministic transition probabilities. To support the transition into end states, we will also introduce an exit action $\mathcal{E} \in A^m$, that will deterministically transition into the end state $\bot$.

We will define the reward function as $\mathcal{R}^m : F \times A \rightarrow \mathbb{R}$, i.e., a mapping from a state variable and action pair to a number.The reward associated with a state, action pair is given as

$$\mathcal{R}^m(s,a) = \begin{cases} \sum_{f \in S} \mathcal{R}^m(f,a) \text{ if } s \neq \bot \\ 0 \text{ otherwise} \end{cases}$$

[1] There are more expressive formalisms that allow one to associate non-unit costs with actions.

A solution to an MDP problem takes the form of a policy $\pi : S^m \rightarrow A$, i.e., a function that maps states to actions. A policy is said to be optimal if it maximizes the total expected discounted reward received under the given policy.

At this point, it is worth noting that for every classical planning MDP task domain $\mathcal{D}^c$, we can build a corresponding task domain $\mathcal{D}^c_m = \langle F^c_m, A^c_m, I^c, T^c_m, \gamma \rangle$, where $F^c_m = F^c \cup \{\perp\}$, $A^c_m$ one action label for each action in $A^c$ plus a label for $\mathcal{E}$, $I^c$ is the initial state (and same as before), the transition $T^c_m$ returns one only if it is a valid transition per $\Gamma^c$. For the application of actions in states where the preconditions are not met, we will assign a probability of '1' to transition to $\perp$, and $\perp$ is treated as an absorber state.

We will use the notion of trace as a shared notion of behavior that can be used in both settings. A trace $\tau$ for a policy or plan consists of a sequence of state-action pairs that results from the execution of a policy or plan in the initial state. We will also use the notation $\mathcal{P} = \langle \mathcal{D}, \mathcal{O} \rangle$ as a generalized scheme of model representation that can stand in for both classical planning problems and MDP. Depending on the context, $\mathcal{O}$ could either be a reward or a goal.

## 4 Specification and Prediction

With the basic notations in place, we can precisely define the exact questions under examination. In particular, we are interested in the user's ability to specify an objective that can lead to some desired behavior or be able to predict behavior that could result from optimizing for a given objective function. These two problems correspond to the primary ways users specify objectives. We start with the specification problem, where a user must identify an objective resulting in a target behavior.

**Definition 1** *For a given domain model $\mathcal{D}$ and a target trace $\tau$, the specification problem corresponds to finding an objective $\mathcal{O}$, such that $\tau$ is a trace for an optimal solution for the problem $\mathcal{P} = \langle \mathcal{D}, \mathcal{O} \rangle$.*

If the optimal solution for a given objective specification (i.e., a goal or reward) leads to a trace $\tau$, then we will refer to that objective specification as being a correctly specified objective for $\tau$, else it is referred to as a misspecified objective.

Moving from the more general to specific settings, we start seeing differences in the properties of the specification forms. For example, one can show that even when a goal specification cannot be found for a given trace, it might be possible to find a reward function in a corresponding MDP.

**Proposition 1** *For a classical planning domain $\mathcal{D}^c$, let $\tau = \langle I, a_0, ..., s_k \rangle$, be a trace such that for every consecutive state-action-state tuple $s_i, a_i, s_{i+1}$ we have $\Gamma^c(s_i, a_i) = s_{i+1}$, and the trace contains no repeating states, then even if there exists no goal for which $\tau$ is a trace for an optimal plan, there still exists a reward function for the corresponding MDP domain $\mathcal{D}^c_m$ for which $\tau$ is a trace for an optimal policy.*

The above proposition can be proven by showing that there exist traces that satisfy the property for which no goal exists and by showing the existence of a reward for which the trace is part of an optimal policy. First, consider a trace that includes an avoidable subsequence. In other words, let $s_i$ and $s_j$ be part of $\tau$ such that their positions in sequences are separated by more than two positions, i.e., there are at least two actions between $s_i$ and $s_j$. Now let's assume there exists an action $a$, such that $\Gamma^c(s_i, a) = s_j$. Then, by definition, this trace cannot be part of an optimal plan since you can get a shorter trace that results in the same state by removing the original actions between $s_i$ and $s_j$. As for the second part, consider a reward function that assigns zero to every state. Under this reward function, all policies have the same value and are optimal. Given the fact that all transition in the trace corresponds to valid ones in the original domain model, there exists at least one policy for which this is a valid trace.

This example shows how the reward function provides a clear advantage in terms of expressivity. However, this advantage goes even further: the knowledge about the goal will allow us to reconstruct a reward function for the corresponding model directly. Specifically, one can create a reward function that assigns a positive reward to all the goal fluents for the exit action, or more formally,

**Proposition 2** *For a trace $\tau$ and a classical planning domain $\mathcal{D}^c$, let $\mathcal{G}^c$ be a correctly specified goal, then $\mathcal{R}^c_m$ must be a correctly specified reward for $m(\mathcal{D}^c)$, when*

$$\mathcal{R}^c_m(f, a) = \begin{cases} r^+ \textit{ if } f \in \mathcal{G}^c \textit{ and } a = \mathcal{E} \\ 0 \ \textit{ otherwise} \end{cases}$$

The validity of the above proposition is straightforward. The agent only receives a positive reward for performing the exit action from states that satisfy the goal specification. The presence of a discount factor means that this would need to be achieved in as few steps as possible.

Now, it is also worth noting that not all correctly specified objectives are equal. In particular, we can identify two categories. In one case, the user may not have provided enough details; we will call such cases examples of *underspecification*. In the latter case, the user would have provided more details than needed or examples of *overspecification*. The implications of these two design flaws are wildly different. While overspecification might reduce the set of optimal policies and prevent the AI system from coming up with creative solutions, underspecification could result in unexpected behavior or specification gaming. We can more formally define these two categories as follows:

**Definition 2** *For a domain model $\mathcal{D}$ and a target trace $\tau$, a given specification $\mathcal{O}$ is said to be underspecified if there are other traces $\tau' \neq \tau$ that could result from other optimal solutions for $\mathcal{P} = \langle \mathcal{D}, \mathcal{O} \rangle$.*

In the above definition, underspecification is purely defined by the fact that there are other traces and solutions possible (given the deterministic settings we consider, there is a one-to-one mapping between solutions and traces). On the other hand, defining overspecification requires us to use a notion of specification size, i.e., $|\mathcal{O}|$, where for goals, the size is given by the number of fluents in the specification, and for rewards, the number of fluent action pairs with non-zero values. Now, we can define overspecification to be cases where specifications of smaller size exist that are not underspecified.

**Definition 3** *For a domain model $\mathcal{D}$ and a target trace $\tau$, a given specification $\mathcal{O}$ is said to be overspecified if (a) $\mathcal{O}$ is not underspecified and (b) there exists another correct specification $\mathcal{O}'$, such that $\mathcal{O}'$ is not an under specification and $|\mathcal{O}'| < |\mathcal{O}|$.*

This brings us to the end of the section discussing the first task, namely, objective specification. The second task corresponds to the user's ability to make inferences based on the given objective. Here, we consider the simple case of whether a user can tell if a trace is possible under a given specification.

**Definition 4** *For a given problem $\mathcal{P} = \langle \mathcal{D}, \mathcal{O} \rangle$ and a trace $\tau$, the prediction problem corresponds to identifying whether $\tau$ is a trace for an optimal solution for the problem $\mathcal{P}$.*

## 5 Hypotheses

Our study is primarily designed to measure how the choice of specification mechanism can affect the user's ability to specify objectives and predict agent behavior. The primary hypotheses we plan to test here are as follows, since goals register as an intuitively easier form of specification:

- H1-a: Participants are more likely to provide accurate goals than accurate reward specifications.
- H1-b: Participants are more likely to correctly interpret goals than reward specifications.

The next question we consider concerns the participants' workload, specifically the cognitive load imposed and the time required for each of the two specification mechanisms.

- H2-a: Reward specifications will result in a higher workload than goal specifications and will require a longer time to finish.
- H2-b: Trying to interpret reward functions will result in a higher workload than goal specifications and will require a longer time to finish.

Now, we also wanted to use this as an opportunity to understand ways in which the user specification may differ from the minimal specification, which brings us to the hypothesis:

- H3: Participants are more likely to underspecify objectives than to overspecify them.

We will test the above hypothesis for both reward and goal specification cases.

To assess the H2, we measure the participants' workload for each objective specification mechanism and task in the survey using the NASA Task Load Index (TLX). NASA TLX has six dimensions: mental demand, physical demand, temporal demand, performance, effort, and frustration level (Hart, 1986). Each dimension is measured using a Likert rating scale, ranging from 0 to 20. 0 indicates the lowest possible level of demand or workload for that dimension (e.g., the task was not demanding, no effort was required, or no frustration was experienced). Conversely, 20 indicates the highest possible level of demand or workload for that dimension (e.g., the task was extremely demanding, required maximum effort, or extreme frustration).

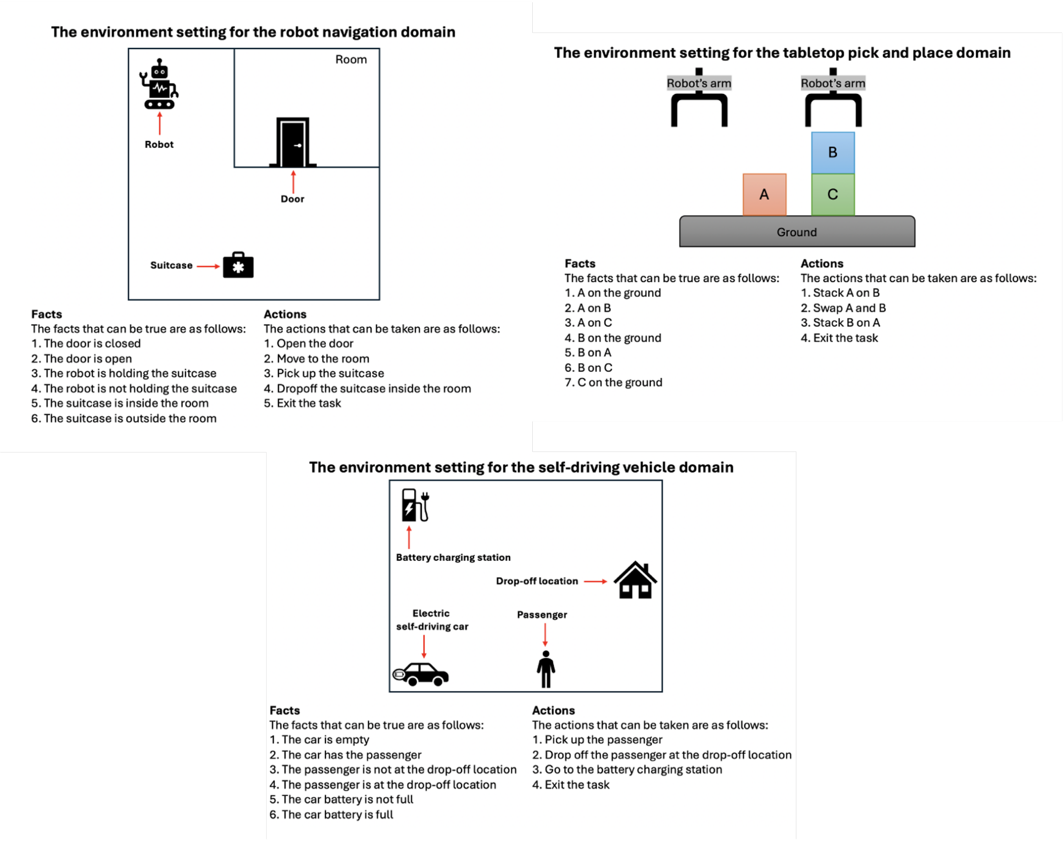
# 6 Methods

## 6.1 Study Design

To compare the two mechanisms, we designed three intuitive but diverse domains in which two primary tasks related to each mechanism can be tested: (1) the user's ability to provide an objective specification that will result in a given behavior and (2) their ability to predict the behavior from a given specification. We chose domains that non-AI experts could understand without considerable training, but that corresponded to potential real-world robotics applications. Specifically, the domains included (1) a robot navigation task, (2) a tabletop pick-and-place task, and (3) a task with a self-driving vehicle. We chose deterministic versions of the tasks to avoid potential confounders that may arise from the stochasticity of the environment dynamics. The environment setting for each domain can be seen in Figure 1.

The navigation task involves robots navigating through a workspace. In this case, a robot needs to pick up and drop off a suitcase in different locations within a small workspace. The pick and place domain contains a set of blocks that can be stacked on top of one another. The objective is usually to achieve a specific configuration of the blocks. For the self-driving vehicle domain, we have a self-driving car powered by a battery that needs to pick up and drop off a passenger in different locations. It also needs to charge the battery to make sure that the battery is enough to perform its task. In each environment setting, the current state is defined by a set of binary variables, henceforth referred to as facts. There is also a set of actions that can be taken by the robot, including an exit action that will allow the robot to end the task. Each domain had about 6-7 facts and 4-5 actions. We choose to keep the facts and action counts similar to roughly balance the workload between domains.

We created surveys that test the participants' ability to specify an objective that will result in some provided behavior or their ability to predict what behavior will result from a given objective for each of these domains. The survey uses a mixed study design, combining both between-subjects and within-subjects study designs. The participants are shown either the specification task or prediction task (making this study design between-subjects), chosen from three different problem domains as mentioned above. Given the problem domain, the participants are tested on how well they are able to complete the specified task across the two objective specification mechanisms (within subjects). We use a counterbalancing technique to vary the order in which participants will be shown the different



The environment setting for the robot navigation domain

Facts
The facts that can be true are as follows:
1. The door is closed
2. The door is open
3. The robot is holding the suitcase
4. The robot is not holding the suitcase
5. The suitcase is inside the room
6. The suitcase is outside the room

Actions
The actions that can be taken are as follows:
1. Open the door
2. Move to the room
3. Pick up the suitcase
4. Dropoff the suitcase inside the room
5. Exit the task

The environment setting for the tabletop pick and place domain

Facts
The facts that can be true are as follows:
1. A on the ground
2. A on B
3. A on C
4. B on the ground
5. B on A
6. B on C
7. C on the ground

Actions
The actions that can be taken are as follows:
1. Stack A on B
2. Swap A and B
3. Stack B on A
4. Exit the task

The environment setting for the self-driving vehicle domain

Facts
The facts that can be true are as follows:
1. The car is empty
2. The car has the passenger
3. The passenger is not at the drop-off location
4. The passenger is at the drop-off location
5. The car battery is not full
6. The car battery is full

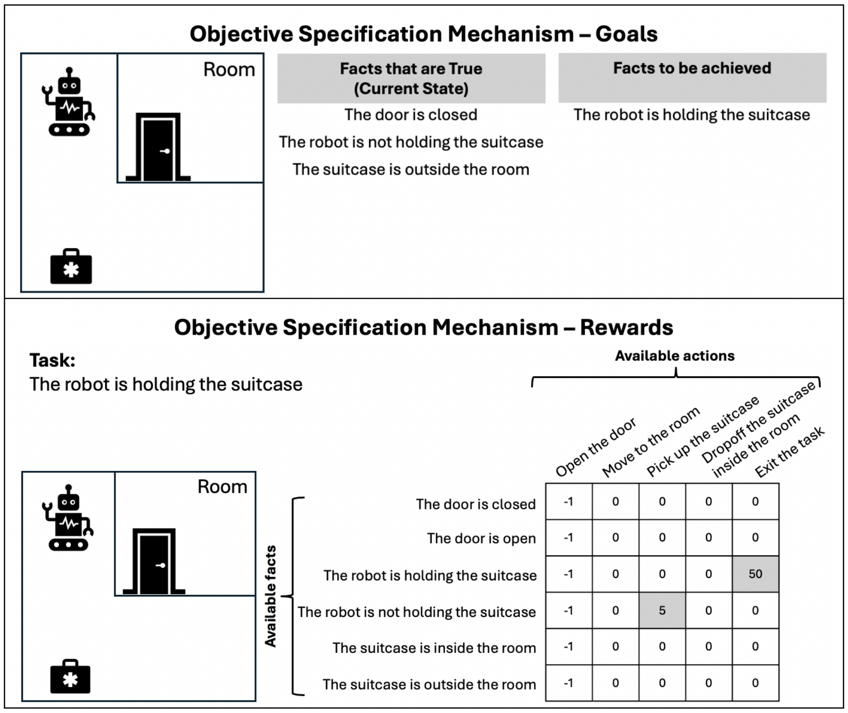
Actions
The actions that can be taken are as follows:
1. Pick up the passenger
2. Drop off the passenger at the drop-off location
3. Go to the battery charging station
4. Exit the task

Figure 1: A visualization of each domain used in the study. Top left: a robot navigation task. Top right: a pick-and-place task. Bottom: a self-driving vehicle task.

specification mechanisms. This is to ensure that no single order influences the results of the study. The counterbalancing is achieved through a least-fill random basis.

For each objective specification mechanism, there are two sections in the survey: demo and test. The demo section is a learning phase, where participants are familiarized and introduced to the concepts of goal and reward specifications. In this section, participants are shown a video that demonstrates a simple behavior along with the corresponding goal or reward (see the example illustration in Figure 2). For goals, the video shows the "facts to be achieved (goal state)" and how the "facts that are true (current state)" change during the duration of robot behavior until it reaches the goal state. On the other hand, for rewards, the video shows the rewards matrix and how individual rewards from the matrix are added to the total when the agent performs specific actions. For example, based on the illustration in Figure 2, the agent will get 50 points if it takes an "exit the task" action while the fact that "the robot is holding the suitcase" is true.

For the first task, i.e., ease of objective specification, the test section shows a sample behavior to the user. Then, participants are asked to come up with goals and/or rewards for that scenario. Figure 3 presents screenshots of the interface provided to the user to specify the objective. We refer to goals as facts and rewards as scores to simplify the description to non-AI expert participants. From the participants' answers, we can determine whether their specifications are correct or incorrect. Correct specifications for goals were measured by comparing the facts to be achieved listed by participants with the correct list of facts to be achieved that were associated with the given video. As for the reward specifications, the correctness was measured by using the value iteration algorithm to get the trajectory implied by the participant's reward matrix. If the trajectory is similar to the trajectory shown in the video, then the reward specification is correct.



**Objective Specification Mechanism – Goals**

| Facts that are True (Current State) | Facts to be achieved |
|---|---|
| The door is closed<br>The robot is not holding the suitcase<br>The suitcase is outside the room | The robot is holding the suitcase |

**Objective Specification Mechanism – Rewards**

**Task:**
The robot is holding the suitcase

Available actions

Available facts

| | Open the door | Move to the room | Pick up the suitcase | Dropoff the suitcase inside the room | Exit the task |
|---|---|---|---|---|---|
| The door is closed | -1 | 0 | 0 | 0 | 0 |
| The door is open | -1 | 0 | 0 | 0 | 0 |
| The robot is holding the suitcase | -1 | 0 | 0 | 0 | 50 |
| The robot is not holding the suitcase | -1 | 0 | 5 | 0 | 0 |
| The suitcase is inside the room | -1 | 0 | 0 | 0 | 0 |
| The suitcase is outside the room | -1 | 0 | 0 | 0 | 0 |

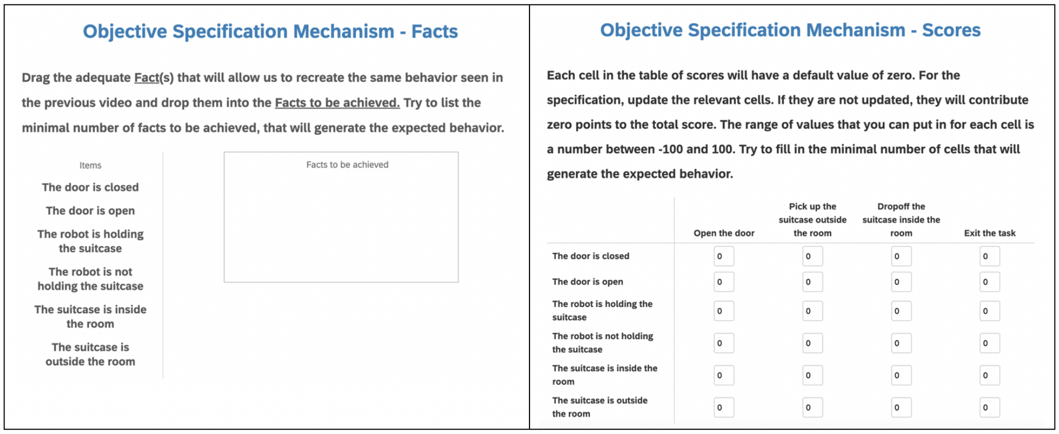
Figure 2: Illustrations for the sample specifications that could be shown to the participants.

For the incorrect goals or rewards, the potential sources of errors can be analyzed, including underspecification. We have provided the formal definition of underspecification in Section 4, and we used these definitions to measure underspecification in our experimental results. In our task design, underspecification for goals occurs when a participant specifies only a subset of the goals. This incomplete specification can lead to unexpected behavior (and is hence an incorrect specification). For the rewards specification, we implemented the value iteration algorithm to get the trajectory implied by the participant's reward matrix. If the participant's reward specification is correct but allows for multiple possible optimal traces (i.e., the rewards matrix does not uniquely determine the intended behavior), we classify this as underspecification.

We similarly measure overspecification. We have provided the formal definition of overspecification in Section 4. We should note that overspecification is a correct specification. Overspecification for goals is measured by comparing the set of facts a participant lists as goals to the correct minimal specification. If a participant includes additional facts that are true in the final state but are not necessary for achieving the intended outcome, this is considered overspecification. For rewards, overspecification is identified when a participant assigns non-zero values to more fluent-action pairs in the reward matrix than are present in the correct minimal specification.

On the other hand, to test how easily non-AI experts can understand goals and rewards, instead of showing the demonstration, we show the correct goal (list of facts to be achieved) or the rewards specification (in the form of scores). Then, we ask the participants to predict or interpret the behavior of the agent based on that. Specifically, we provide three video options and ask them to choose one that most aligns with the given goals or rewards. For the prediction task, only one option is correct.

Additionally, at the end of the survey, we ask the participants to directly compare the two specification mechanisms in terms of their ease, intuitiveness, likeability, and challenge. We also ask for qualitative feedback on why they think that particular objective specification mechanism is easier or harder than the other. Finally, we collect demographic information, including age, gender, highest level of education, and familiarity with computer science and AI subjects.



Objective Specification Mechanism - Facts

Drag the adequate Fact(s) that will allow us to recreate the same behavior seen in the previous video and drop them into the Facts to be achieved. Try to list the minimal number of facts to be achieved, that will generate the expected behavior.

Items

The door is closed

The door is open

The robot is holding the suitcase

The robot is not holding the suitcase

The suitcase is inside the room

The suitcase is outside the room

Facts to be achieved

Objective Specification Mechanism - Scores

Each cell in the table of scores will have a default value of zero. For the specification, update the relevant cells. If they are not updated, they will contribute zero points to the total score. The range of values that you can put in for each cell is a number between -100 and 100. Try to fill in the minimal number of cells that will generate the expected behavior.

| | Open the door | Pick up the suitcase outside the room | Dropoff the suitcase inside the room | Exit the task |
|---|---|---|---|---|
| The door is closed | 0 | 0 | 0 | 0 |
| The door is open | 0 | 0 | 0 | 0 |
| The robot is holding the suitcase | 0 | 0 | 0 | 0 |
| The robot is not holding the suitcase | 0 | 0 | 0 | 0 |
| The suitcase is inside the room | 0 | 0 | 0 | 0 |
| The suitcase is outside the room | 0 | 0 | 0 | 0 |

Figure 3: Sample interfaces used by the participants to specify goals (left) and rewards (right). These examples are taken from the navigation task.

### 6.2 Participants and Procedure

Before the main study, we ran a small pilot think-aloud study with three participants to refine the study design. For the primary user studies, we recruited a total of 30 participants: 15 participants (8 males and 7 females) for the specification task and 15 participants (7 males and 8 females) for the prediction task. We recruited participants on the Prolific platform. The participants were given a link to take the survey. They were paid $18.5 USD per hour, and they identified their native language as English. The majority of them (21 out of 30) reported having never taken an AI course.

In addition to the primary user studies, we also conducted two additional user studies that served as variants of the specification task. Altogether, we carried out four distinct user studies: (1) Study 1: assessed participants' ability to provide objective specifications, (2) Study 2: also assessed participants' ability to provide objective specifications, but used different videos to highlight how intermediate fact values change, (3) Study 3: also tested participants' ability to provide objective specifications, but focused on a variation of the navigation task where the participants simply provided scores for each state variable achievement, (4) Study 4: tested participants' ability to predict behavior based on the provided objective specifications. Further details about participant demographics for all four user studies are provided in the supplementary materials.

This study was IRB-approved. Participants were provided with informed consent before they started the survey. Multiple attention check questions were included throughout the study. For the main study, each participant was shown all three domains in random order. For the first variant of the specification study (Study 2), each participant was also shown all three domains in random order. Finally, for the second variant of the specification study (Study 3), each participant was only shown one domain, i.e., the robot navigation domain. For all user studies, the order in which the specification mechanism was shown was randomized to ensure the results between the two objective specification mechanisms were counterbalanced.

## 7 Results and Discussions

### 7.1 Impressions from the Think-Aloud Study

We used the think-aloud study (Baxter et al., 2015) as a means of both testing our interface, particularly for specification tasks, and collecting some initial anecdotal information on the mechanisms. The reactions we observed were aligned with what we hypothesized (H1-a and H2-a), where the participants showed more positive reactions to the goal specification interface as opposed to the reward. Some reactions to goal specifications included: "This one is fun, like playing," and "The task was super easy." On the other hand, for the reward specification, users reported a lack of confidence about their ability to correctly provide such specifications: "I don't understand, I'm very bad at this," and "I don't know why this is confusing me." Their qualitative feedback at the end of the survey also reflected their strong preference for using the goal specification mechanism.

### 7.2 Specification Task

We started by analyzing the initial results from the specification task. In regard to hypothesis H1-a, we calculated the number of times the participants were able to provide correct specifications (presented in Table 1). We were surprised to find that the participants were actually able to identify correct reward functions more frequently than correct goal specifications. Further, analyses of the results showed that the most frequent mistakes made by participants in goal specification involved the inclusion of intermediate facts in the goal specification. These intermediate facts, while made true by the agent's action, are also made false by further actions in the plan. For example, the subjects might indicate that "the robot is holding the suitcase", but in the observation of the environment, the robot places the suitcase down at the end of the video. As such, including these intermediate facts in the final goal specification leads to an unachievable objective specification. The goals provided by the users reflected a more procedural description of the agent's behavior than a final goal state description. On the other hand, such intermediate state scores can be more naturally incorporated into the reward function. To analyze the factors that could explain these results, we created two follow-up variants of the specification and reran our study.

In the first follow-up, we updated all our videos to highlight how intermediate fact values change. In each of our demonstration videos, we added animations that showed which facts became false. We reran the experiment on five participants (thus collecting 15 specifications per mechanism). The results from the study are presented in Table 2. While we see that the additional information does improve the overall percentage of correct goal specification, the resulting percentage is similar to that of the rewards. Thus indicating that the additional information balances participants' ability to craft goals or rewards.

In the second follow-up, we considered a variation of the navigation task in which participants provided scores for each state variable's achievement. This variant was motivated by the possibility that including actions in the specification mechanism might help the participant by allowing them to think procedurally about the task. Here, we set a specific absorbing state, and the reward for each state was set to the sum of rewards associated with each state factor. We ran this variant on 15 participants, and the percentage of correct goal and reward specifications was, in fact, the same (Table 3). This shows that the presence of actions assisted participants in crafting rewards, and that crafting rewards over states instead of state-action-state tuples is a harder task in these domains.

Taken together, this collection of results suggests that the hypothesis that goals are easier to specify than rewards may not be true. This is particularly surprising, given that this hypothesis is quite frequently taken to be self-evident in the literature (cf. (Mechergui & Sreedharan, 2024)).

However, when we move on to the hypothesis related to workload and time taken (H2-a), we see a clear distinction between the two specification mechanisms, with subjects overwhelmingly preferring the goal mechanism. Running paired t-tests shows that there is a statistically significant difference between the cognitive load of goal specification ($M$ = 9.444, $SD$ = 5.057) and reward specification ($M$ = 12.689, $SD$ = 5.008). There is also a statistically significant difference between the time taken to complete the goal specification ($M$ = 82.014, $SD$ = 36.225) and reward specification ($M$ = 148.521, $SD$ = 87.015). In addition, we also see similar responses with respect to the qualitative responses, with most participants finding goals easier to specify (86.67%) and more intuitive (73.33%). The supplementary file provides the breakdown of individual dimensions of the workload and more details on the qualitative feedback. These results support our hypothesis H2-a.

Finally, moving to H3, our results again do not support our hypothesis. In fact, we say more instances of the users overspecifying their objectives than underspecification (see Table 1, 2, and 3). Such patterns were also replicated in the incorrect specifications. Looking at incorrect goal specification, we saw a larger set of participants (75%) added incorrect facts as opposed to leaving out some facts (0.027%). We include the more complete result of the statistical analysis in the supplementary materials.

Table 1: Results from the main specification user study

| **Category** | **Sub-category** | **Percentage of total response** | |
|---|---|---|---|
| | | **Goals** | **Rewards** |
| Correct | Correct minimal specification | 4.45 | 2.22 |
| | Correct but overspecified | 13.33 | 35.56 |
| | Correct but underspecified | - | 8.89 |
| Incorrect | Incorrect because gave subset | 82.22 | 53.33 |
| | **Total** | **100** | **100** |

Table 2: Results from the first variant of the specification study

| **Category** | **Sub-category** | **Percentage of total response** | |
|---|---|---|---|
| | | **Goals** | **Rewards** |
| Correct | Correct minimal specification | - | - |
| | Correct but overspecified | 73.33 | 66.67 |
| | Correct but underspecified | - | - |
| Incorrect | Incorrect specification | 26.67 | 33.33 |
| | **Total** | **100** | **100** |

Table 3: Results from the second variant of the specification study

| **Category** | **Sub-category** | **Percentage of total response** | |
|---|---|---|---|
| | | **Goals** | **Rewards** |
| Correct | Correct and minimal specification | - | - |
| | Correct and overspecification | 66.67 | 66.67 |
| | Correct and underspecification | - | - |
| Incorrect | Incorrect specification | 33.33 | 33.33 |
| | **Total** | **100** | **100** |

### 7.3 Prediction Task

As discussed, the goal of the prediction task was to test whether a user can predict the behavior that could result from a given specification. We see that as a proxy for the ease with which users can correctly interpret specifications expressed using each mechanism. For the prediction task, we also see a similar pattern. The participant's accuracy in predicting behavior based on the given goals function and reward function is comparably high, with 93.33% predicting the goals function correctly and 91.11% predicting the rewards function correctly. Here, the difference is not high enough to establish any statistically significant difference between the two groups. As for the results related to the cognitive workload, our t-test was not able to establish any significant difference between the prediction from the goal function ($M$ = 6.267, $SD$ = 6.308) and from the reward function ($M$ = 7.044, $SD$ = 6.502) with $P$-value equal to .251. There was also no significant difference between the time taken to complete the prediction from goal function ($M$ = 82.840, $SD$ = 75.285) and from reward function ($M$ = 90.873, $SD$ = 59.339); $t(44)$ = -0.878, $P$ = .385. This seems to suggest that both of our hypotheses, H1-b and H2-b, may not hold.

However, when we move on to the participant preference between the two mechanisms, most participants find the goal function easier to predict (86.67%) and more intuitive (80%). In addition to this, most participants reported that the reward function is more challenging to predict (80%). These preferences are consistent in both specification and prediction tasks.

### 7.4 Results Summary

From our experiments, we find that our hypotheses H1-a and H1-b are surprisingly not supported: we did not find evidence that people are able to more correctly specify or interpret goals over reward functions. Despite this, we find that there was a significant difference in the cognitive effort and time needed to specify objectives: goals were a clear winner on these axes (H2-a). We were also surprised to find that people are not more likely to underspecify goals than to overspecify (H3). Overall, though, the subjective feedback reflects that participants strongly preferred using goals over reward functions.

### 7.5 Limitations of Study Scenarios

It is important to acknowledge the limitations in our study scenarios. All studies were carried out in purely deterministic settings, where the agents can not get stuck in loops or face probabilistic transitions. While this is stereotypical of many tasks where goals are used, this does not necessarily represent all the ways rewards could be utilized, which is a more general specification mechanism.

Similarly, we considered simple enough scenarios where the participants could easily enumerate all possible facts and incorporate them into the specification. While this design choice helps participants avoid feeling overwhelmed and enables the clear measurement of specification correctness, it also means that our results may not generalize to more complex domains where the state space is larger or the dynamics are not deterministic.

Additionally, several potential confounding factors are present in the experimental design, including task complexity. The simplicity of the scenarios may not adequately reveal the full range of participants' ability to provide specifications and predict the behavior of the agents. If the tasks are too simple, participants may not encounter the ambiguities or difficulties that would arise in more complex settings. Uncontrolled environmental factors, such as the testing environment, time of day, or participant fatigue, can also influence performance, which could potentially confound the results.

Given these limitations, our findings should be viewed as preliminary. Future work is needed to examine whether the results we observed in this study persist under more varied and complex settings to ensure broader validity.

## 8 Conclusion

In this paper, we performed a comparison to assess how easy it would be for non-expert users to provide and understand reward specifications and goal specifications. Our results provide evidence that people's ability to provide and understand rewards is fairly comparable to that of goals. However, there is a clear difference in the user preferences and the cognitive load imposed by the two methods (at least for the specification task). One interesting question to ask in this context would be whether this difference can be explained by the interface we used for our study. As such, one would want to investigate if it is possible to develop interfaces that allow users to intuitively provide reward functions. Such interfaces would have pretty immediate advantages, given that reward functions are more expressive than goals. Future work should investigate how goals and rewards compare with other objective specification mechanisms, such as policy sketches and reward machines.

**Acknowledgments**

This study was funded by NSF grant 2303019.

## References

David Abel, Will Dabney, Anna Harutyunyan, Mark K. Ho, Michael L. Littman, Doina Precup, and Satinder Singh. On the expressivity of markov reward. In *NeurIPS*, pp. 7799–7812, 2021.

Saleema Amershi, Maya Cakmak, William Bradley Knox, and Todd Kulesza. Power to the people: The role of humans in interactive machine learning. *AI magazine*, 35(4):105–120, 2014.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Andrea Bajcsy, Dylan P Losey, Marcia K O'Malley, and Anca D Dragan. Learning from physical human corrections, one feature at a time. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 141–149, 2018.

Kathy Baxter, Catherine Courage, and Kelly Caine. *Understanding your users: a practical guide to user research methods*. Morgan Kaufmann, 2015.

Erdem Biyik and Dorsa Sadigh. Batch active preference-based learning of reward functions. In *Conference on robot learning*, pp. 519–528. PMLR, 2018.

Serena Booth, W. Bradley Knox, Julie Shah, Scott Niekum, Peter Stone, and Alessandro Allievi. The perils of trial-and-error reward design: Misdesign through overfitting and invalid task specifications. In *AAAI*, pp. 5920–5929. AAAI Press, 2023.

Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, 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.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Michael T Cox. A model of planning, action, and interpretation with goal reasoning. In *Proceedings of the 4th Annual Conference on Advances in Cognitive Systems*, pp. 48–63, 2016.

Nadia Figueroa, Salman Faraji, Mikhail Koptev, and Aude Billard. A dynamical system approach for adaptive grasping, navigation and co-manipulation with humanoid robots. In *2020 IEEE International conference on robotics and automation (ICRA)*, pp. 7676–7682. IEEE, 2020.

Hector Geffner and Blai Bonet. *A concise introduction to models and methods for automated planning*. Morgan & Claypool Publishers, 2013.

Sandra G Hart. Nasa task load index (tlx). 1986.

Rodrigo Toro Icarte, Toryn Q Klassen, Richard Valenzano, and Sheila A McIlraith. Reward machines: Exploiting reward function structure in reinforcement learning. *Journal of Artificial Intelligence Research*, 73:173–208, 2022.

W Bradley Knox and Peter Stone. Interactively shaping agents via human reinforcement: The tamer framework. In *Proceedings of the fifth international conference on Knowledge capture*, pp. 9–16, 2009.

W Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. Models of human preference for learning reward functions. *Transactions on Machine Learning Research*, 2022.

W. Bradley Knox, Alessandro Allievi, Holger Banzhaf, Felix Schmitt, and Peter Stone. Reward (Mis)design for autonomous driving. *Artificial Intelligence*, 316(103829), 2023.

Dylan P Losey and Marcia K O'Malley. Including uncertainty when learning from human corrections. In *Conference on Robot Learning*, pp. 123–132. PMLR, 2018.

James MacGlashan, Mark K Ho, Robert Loftin, Bei Peng, Guan Wang, David L Roberts, Matthew E Taylor, and Michael L Littman. Interactive learning from policy-dependent human feedback. In *International conference on machine learning*, pp. 2285–2294. PMLR, 2017.

Malek Mechergui and Sarath Sreedharan. Goal alignment: Re-analyzing value alignment problems using human-aware AI. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pp. 10110–10118. AAAI Press, 2024.

Martin L Puterman. Markov decision processes. *Handbooks in operations research and management science*, 2:331–434, 1990.

Harish Ravichandar, Athanasios S Polydoros, Sonia Chernova, and Aude Billard. Recent advances in robot learning from demonstration. *Annual review of control, robotics, and autonomous systems*, 3(1):297–330, 2020.

Stuart Russell. Human-compatible artificial intelligence., 2022.

Rohin Shah, Vikrant Varma, Ramana Kumar, Mary Phuong, Victoria Krakovna, Jonathan Uesato, and Zac Kenton. Goal misgeneralization: Why correct specifications aren't enough for correct goals. *arXiv preprint arXiv:2210.01790*, 2022.

Herbert A Simon. *The Sciences of the Artificial, reissue of the third edition with a new introduction by John Laird*. MIT press, 2019.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Christopher Taylor. Aristotle on practical reason. In *The Oxford Handbook of Topics in Philosophy*. Oxford University Press, 2019. ISBN 9780199935314.

Andrea L Thomaz and Cynthia Breazeal. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence*, 172(6-7):716–737, 2008.

Nhi Tran. Goals vs. actions as user-facing representations for robot programming. In *2024 AAAI Fall Symposium Series*. AAAI, 2024.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

# Supplementary Materials

*The following content was not necessarily subject to peer review.*

## 9 Participants' Demographics for the Objective Specification Mechanisms User Studies

Table 4 provides more detailed information about the participants' demographics from the four user studies that we conducted. As mentioned in Section 6, the following is a brief description of each user study. Study 1 is a primary study that assessed participants' ability to provide objective specifications. Similarly, Study 2 also assessed participants' ability to provide objective specifications, but used different videos to highlight how intermediate fact values change. Study 3 also tested participants' ability to provide objective specifications, but focused on a variation of the navigation task where the participants simply provided scores for each state variable achievement. Finally, Study 4 is also a primary study, with the objective to test participants' ability to predict behavior based on the provided objective specifications. For Studies 1, 3, and 4, we have 15 participants in each study. And for Study 2, we have 5 participants. Based on the demographic variable "have taken AI courses", we can see that the majority of the participants in each study declared that they have never taken an AI course.

Table 4: Participants' demographics for the objective specification mechanisms user studies

| **Demographic Variable** | **Category** | **Frequency** | | | |
|---|---|---|---|---|---|
| | | **Study 1** | **Study 2** | **Study 3** | **Study 4** |
| Age | 18-24 years old | 2 | 0 | 2 | 2 |
| | 25-34 years old | 4 | 0 | 7 | 4 |
| | 35-44 years old | 4 | 3 | 4 | 3 |
| | 45-54 years old | 2 | 2 | 2 | 4 |
| | 55+ years old | 3 | 0 | 0 | 2 |
| Sex | Male | 8 | 2 | 8 | 7 |
| | Female | 7 | 3 | 7 | 8 |
| Highest level of education | High school or equivalent | 6 | 0 | 3 | 3 |
| | Attended college/ university | 2 | 1 | 5 | 1 |
| | Associate degree | 0 | 0 | 0 | 2 |
| | Bachelor's degree | 4 | 2 | 5 | 6 |
| | Master's degree | 3 | 1 | 2 | 2 |
| | Doctorate degree | 0 | 1 | 0 | 1 |
| Have taken Computer Science courses | Yes | 7 | 2 | 6 | 9 |
| | No | 8 | 3 | 9 | 6 |
| Have taken AI courses | Yes | 5 | 2 | 6 | 4 |
| | No | 10 | 3 | 9 | 11 |
| Self-declared AI knowledge | Novice | 6 | 2 | 6 | 9 |
| | Intermediate | 6 | 3 | 5 | 5 |
| | Advanced | 2 | 0 | 3 | 1 |
| | Expert | 1 | 0 | 1 | 0 |

## 10 Statistical Analysis Results from the User Studies

Table 5 provides more detailed information about the statistical results from the four user studies that we conducted. (Note. p < .05. Means and standard deviations are reported for each dependent variable and each condition. Confidence intervals are 95%. t = paired two-sample for means t-test).

Table 5: Statistical analysis results from the user studies

| Experiment | Sample Size | Dependent Variable | Mean | | Standard Deviation | | Confidence Interval | | t(df) | p-value |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Goals | Rewards | Goals | Rewards | Goals | Rewards | | |
| Study 1 (Specification Task - Main Study) | 45 (15 participants * 3 tasks) | Mental Demand | 9.44 | 12.69 | 5.06 | 5.01 | 9.44 ± 1.52 | 12.69 ± 1.50 | -4.63(44) | 0.00003239* |
| | | Physical Demand | 6.67 | 7.20 | 5.99 | 6.64 | 6.67 ± 1.80 | 7.20 ± 1.99 | -0.82(44) | 0.41430453 |
| | | Temporal Demand | 9.16 | 10.96 | 5.30 | 5.06 | 9.16 ± 1.59 | 10.96 ± 1.52 | -2.14(44) | 0.03757128* |
| | | Performance | 14.18 | 10.18 | 4.44 | 5.32 | 14.18 ± 1.33 | 10.18 ± 1.60 | 5.09(44) | 0.00000712* |
| | | Effort | 9.93 | 12.27 | 4.87 | 4.97 | 9.93 ± 1.46 | 12.27 ± 1.49 | -3.66(44) | 0.00066865* |
| | | Frustration | 7.71 | 9.69 | 5.53 | 5.25 | 7.71 ± 1.66 | 9.69 ± 1.58 | -2.67(44) | 0.01067228* |
| | | Task Completion Time | 82.01 | 148.52 | 36.22 | 87.02 | 82.01 ± 10.88 | 148.52 ± 26.14 | -4.58(44) | 0.00003813* |
| Study 2 (Specification Task - First Variant) | 15 (5 participants * 3 tasks) | Mental Demand | 9.67 | 10.93 | 5.75 | 5.78 | 9.67 ± 3.19 | 10.93 ± 3.20 | -0.89(14) | 0.38826852 |
| | | Physical Demand | 5.33 | 4.07 | 7.12 | 5.74 | 5.33 ± 3.94 | 4.07 ± 3.18 | 1.29(14) | 0.21629361 |
| | | Temporal Demand | 9.40 | 10.20 | 5.19 | 6.35 | 9.40 ± 2.88 | 10.20 ± 3.52 | -0.76(14) | 0.45768791 |
| | | Performance | 11.67 | 7.33 | 5.63 | 4.86 | 11.67 ± 3.12 | 7.33 ± 2.69 | 3.39(14) | 0.00440816* |
| | | Effort | 8.07 | 10.33 | 4.61 | 3.70 | 8.07 ± 2.55 | 10.33 ± 2.05 | -2.35(14) | 0.03379337* |
| | | Frustration | 7.00 | 10.53 | 5.26 | 5.64 | 7.00 ± 2.92 | 10.53 ± 3.12 | -2.92(14) | 0.01121695* |
| | | Task Completion Time | 93.40 | 166.85 | 41.76 | 93.57 | 93.40 ± 23.12 | 166.85 ± 51.82 | -2.73(14) | 0.01612311* |
| Study 3 (Specification Task - Second Variant) | 15 (15 participants * 1 task) | Mental Demand | 6.60 | 9.13 | 3.98 | 5.32 | 6.60 ± 2.20 | 9.13 ± 2.94 | -2.30(14) | 0.03765855* |
| | | Physical Demand | 2.80 | 3.20 | 3.12 | 3.43 | 2.80 ± 1.73 | 3.20 ± 1.90 | -0.46(14) | 0.65339767 |
| | | Temporal Demand | 5.20 | 7.87 | 3.84 | 4.52 | 5.20 ± 2.13 | 7.87 ± 2.50 | -2.29(14) | 0.03822985* |
| | | Performance | 16.47 | 12.8 | 4.76 | 4.71 | 16.47 ± 2.64 | 12.8 ± 2.61 | 2.85(14) | 0.01282284* |
| | | Effort | 7.67 | 8.07 | 4.12 | 4.85 | 7.67 ± 2.28 | 8.07 ± 2.68 | -0.43(14) | 0.67695360 |
| | | Frustration | 4.40 | 6.07 | 5.36 | 5.15 | 4.40 ± 2.97 | 6.07 ± 2.85 | -1.53(14) | 0.14840122 |
| | | Task Completion Time | 84.13 | 119.75 | 24.17 | 52.19 | 84.13 ± 13.38 | 119.75 ± 28.90 | -2.64(14) | 0.01922023* |
| Study 4 (Prediction Task - Main Study) | 45 (15 participants * 3 tasks) | Mental Demand | 6.27 | 7.04 | 6.31 | 6.50 | 6.27 ± 1.90 | 7.04 ± 1.95 | -1.16(44) | 0.25115629 |
| | | Physical Demand | 3.29 | 3.07 | 4.77 | 4.78 | 3.29 ± 1.43 | 3.07 ± 1.44 | 0.41(44) | 0.68535270 |
| | | Temporal Demand | 5.20 | 5.47 | 5.14 | 5.48 | 5.20 ± 1.54 | 5.47 ± 1.65 | -0.60(44) | 0.55040858 |
| | | Performance | 17.31 | 15.38 | 4.23 | 5.77 | 17.31 ± 1.27 | 15.38 ± 1.73 | 2.24(44) | 0.03039505* |
| | | Effort | 5.80 | 6.82 | 6.07 | 6.47 | 5.80 ± 1.82 | 6.82 ± 1.94 | -2.04(44) | 0.04776901* |
| | | Frustration | 4.89 | 4.62 | 5.76 | 5.13 | 4.89 ± 1.73 | 4.62 ± 1.54 | 0.60(44) | 0.55040858 |
| | | Task Completion Time | 82.84 | 90.87 | 75.28 | 59.34 | 82.84 ± 22.62 | 90.87 ± 17.83 | -0.88(44) | 0.38475783 |

## 11 Raw Nasa TLX Scores

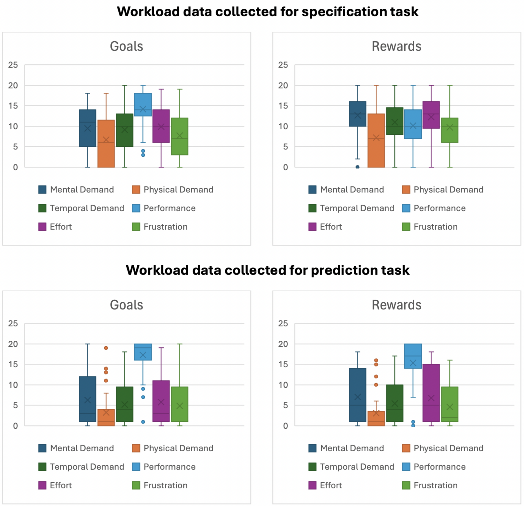


Figure 4: Box tables representing the NASA TLX score collected for both specification and prediction tasks.

## 12 Subjective Feedback from Participants

Here is the subjective feedback provided by the participants for each of the objective specification mechanisms. Here, the participants were asked to select the objective mechanisms they felt most closely matched the description provided

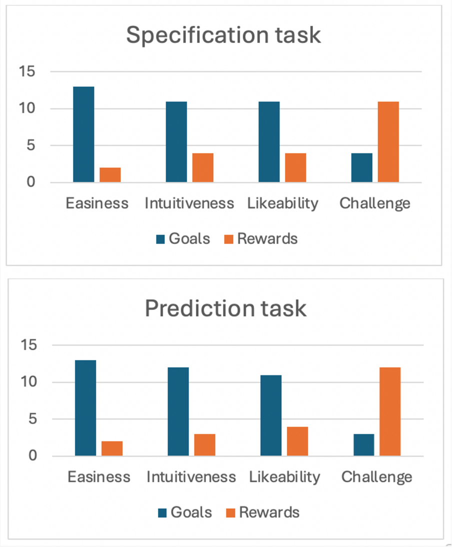


Figure 5: The raw number of selections provided by the participants for each task.

## 13 Screenshots from the Variants

Here are some of the screenshots from the two variants of the specification tasks.

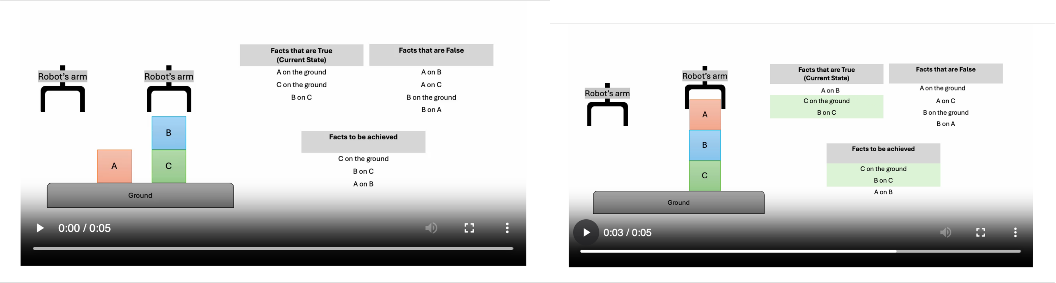
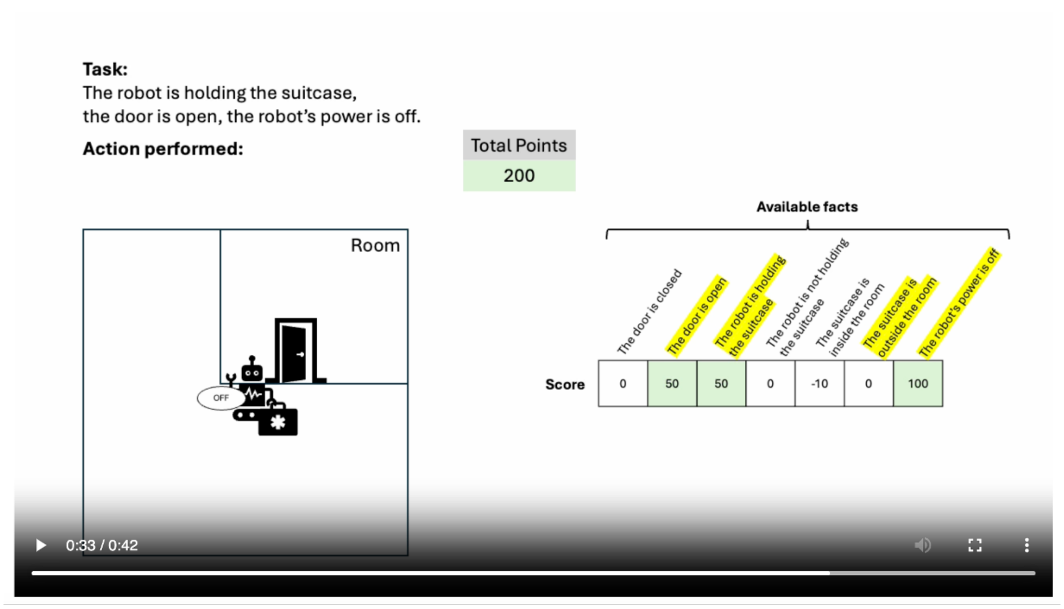


Figure 6: Screenshots from the new video for the first variant that highlights the false facts and how they change over actions.



**Task:**
The robot is holding the suitcase,
the door is open, the robot's power is off.

**Action performed:**

| Total Points |
|---|
| 200 |

Available facts

| | The door is closed | The door is open | The robot is holding the suitcase | The robot is not holding the suitcase | The suitcase is inside the room | The suitcase is outside the room | The robot's power is off |
|---|---|---|---|---|---|---|---|
| Score | 0 | 50 | 50 | 0 | -10 | 0 | 100 |

Figure 7: Screenshots from the second variant that show the task and the new reward specification mechanism.

## 14 Additional Resources

Here are some of the additional resources, including code and an example of the survey: Goals vs. Rewards Repository.