Aligning Robot Behaviors with Human Intents by Exposing Learned Behaviors and Resolving Misspecifications

Serena Booth
MIT CSAIL
Cambridge, MA, USA
sbooth@mit.edu

ABSTRACT

Human-robot interaction is limited by the challenge of writing specifications for robots. We desire alignment between humans' goals and robot behaviors, but this alignment is very hard to achieve [1, 8]. My research tackles this problem. I first study how humans currently write reward functions, and I profile common errors they make when doing so [2]. I then study how humans can inspect robot's learned behaviors. To do so, I introduce a Bayesian inference method for finding behavior examples which cover information-rich test cases [4, 12]. I also study how these examples should be presented to the human through applying human concept learning theories [10, 5, 3]. For the remainder of my thesis, I am studying two questions. First, how these components can be combined such that humans are able to iteratively design better behavioral specifications? Second, can robots smartly interpret humans' erroneous specifications, to correct for these errors?

CCS CONCEPTS

• **Human-centered computing → Interaction design**; • **Computing methodologies → Cognitive science**.

KEYWORDS

Human-Robot Interaction, Explainable AI, Reward Design

ACM Reference Format:
Serena Booth. 2023. Aligning Robot Behaviors with Human Intents by Exposing Learned Behaviors and Resolving Misspecifications. In *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction (HRI '23 Companion), March 13–16, 2023, Stockholm, Sweden.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3568294.3579971

1 INTRODUCTION

Robot behaviors should align with human intents. In service of this goal, people—*especially* experts—should be able to easily specify, debug, change, and comprehend robot behaviors. Challenges abound: when writing reward function specifications, humans often fail to encode their true intent [2]. Debugging learned behaviors is also challenging, as typical explanation methods are flawed [13], and, without appropriate structure, humans can easily form incorrect conceptual models about robot behaviors [3]. To improve the debugging and comprehension processes, I have introduced a method to find representative environments which expose learned behaviors [4, 12]. My larger thesis studies how we can better support people in these joint tasks of writing reward functions and interpreting robots' learned behaviors. My remaining work will focus on two open questions: first, can we build an interactive system which supports humans in recognizing their misspecifications and guides them toward resolving these errors? Second, can we incorporate knowledge of humans' propensity for certain types of misspecifications into inference about a human's intended true reward function (e.g., building off of the inverse reward design framework [6])?

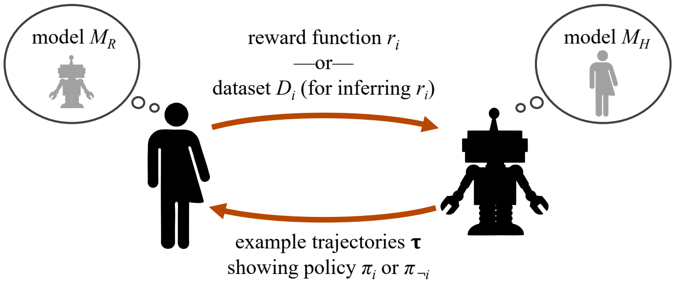
model M_R
reward function r_i
—or—
dataset D_i (for inferring r_i)
model M_H
example trajectories $\boldsymbol{\tau}$
showing policy π_i or $\pi_{\neg i}$

Figure 1: A human specifies a reward function r_i or other learning signal (e.g., a dataset of preferences D_i [9]). The robot models the human (M_H) and updates its policy (π_i). The robot presents strategically-selected examples of its learned behavior to the human, in the form of trajectories (τ), environments, or explanations. The human updates their model of the robot (M_R). This process is iterative. My thesis studies how we can support each of these interactions.

2 SPECIFYING BEHAVIORS

Reinforcement learning (RL) is a promising approach for building robot systems. Reward functions are an exceptionally flexible framework for specifying behaviors, and, as such, there is tremendous optimism about RL's potential [11]. Nonetheless, RL's usefulness is limited by the difficulty of specifying the reward function, which can be misspecified or underspecified [1]. In my recent work [2], I studied humans' reward design process. I first showed that reward functions can be easily overfit to learning algorithms, wherein the reward function is overloaded to both encode the desired outcomes and also facilitate fast and successful learning for a specific algorithm or hyperparameter choice, at the expense of interoperability and generality. I also conducted a user study to assess whether this problem of overfitting equally manifests with human experts

HRI '23 Companion, March 13–16, 2023, Stockholm, Sweden

ACM ISBN 978-1-4503-9970-8/23/03.
https://doi.org/10.1145/3568294.3579971

designing the reward functions. While I confirmed this problem of overfitting is indeed persistent, I was also surprised to discover that—in a simple gridworld—the *majority of expert humans* wrote reward functions which failed to encode the task. I attribute these failures to the mismatched interpretations of the reward function between the human designers and the goals of RL algorithms writ large. Humans view reward functions with a myopic lens, as a mechanism for encoding the relative goodness of each possible state, but the RL objective is instead to maximize the cumulative discounted return. This first study raises the questions: how can we enable humans to write better reward functions, and how can we enable robots to better interpret flawed reward functions?

3 INSPECTING BEHAVIORS

After specifying a reward function and using an RL algorithm to optimize it, how can a person assess whether the robot or AI has learned the behavior that meets the their needs and expectations (i.e., is aligned to their intent)? The most common practice is to observe examples of the robot acting in the world in random or a fixed set of environments. Without adding structure and discipline to this practice, however, this observation process is limited in its usefulness. I propose that we instead support humans in searching for examples that communicate specific, targeted behaviors. In this vein, I first introduced a method for inspecting the behaviors of neural network or other classifiers. In Bayes-TrEx [4], a user specifies a prediction target (e.g., ambiguous across two classes) and a generative model, and we use Bayesian inference to find examples which meet the prediction target. Bayes-TrEx helps with debugging and understanding neural networks, as it can be used to find *ambiguous* examples to communicate class boundaries or highly-confident incorrect classifications to communicate systematic failures. We subsequently adapted this approach to create RoCUS, a method for debugging and improving robot controller behaviors by finding environments in which interesting behavior occurs [12]. RoCUS can be used to assess the behaviors learned through RL with a user-designed reward function; as such, this method can be applied to help the human iterate on their reward function.

4 BUILDING CONCEPTUAL MODELS

How do humans come to understand the behavioral patterns encoded in a reward function, or learned by a robot through this reward function? More generally, how do humans maintain and mitigate uncertainty about their own beliefs? This uncertainty relates to humans' ability to form conceptual models, which are abstract models used for reasoning. The storied study of human concept learning [10, 5] provides a rough blueprint for how to help people build and update accurate and flexible conceptual models, and can be leveraged for human-robot interaction. These theories assert that conceptual models are best formed by experiencing examples that follow highly-structured patterns of variance and invariance [10], and by experiencing structurally-aligned analogous examples [5], which support rapid knowledge transfer. When interacting with a robot or an AI system, a person will inevitably develop a conceptual model of the system's behaviors—but without structure to their learning, the resulting conceptual model may be incorrect or inflexible. I have studied how these theories of human concept learning should be adapted for HRI: my analysis of 35 HRI works showed ad-hoc incorporation of *some* of these patterns [3], but that the community still has many blind spots (e.g., it is exceedingly rare to show counterexamples of robot capabilities, but counterexamples are essential for establishing the bounds of capabilities). My work provides design guidance for better structuring humans' observations of robots' learned behaviors.

5 FUTURE WORK

How can we benefit from the flexibility of the reward function formulation while enabling people to write better behavioral specifications? I will devote my future work to answering this question. I envisage two exciting directions for my work. The first has an HCI-spin and draws on the set of literature I have contributed to thus far: *How can we design interfaces and interactions to support humans in writing better reward functions?* While this first question focuses on how we can improve the humans' reasoning ability, the second question focuses on how robots can compensate for common human errors: *If we can predict human errors, can we leverage that information to inform beliefs over candidate reward functions?*

Q1: How can we better support humans in writing better reward functions? To answer this question, I will integrate my work on RoCUS and human concept learning, and apply these methods to reward design. Specifically, I imagine a person iteratively specifying candidate reward functions (similar to [7]). Using RoCUS, I will find environments and trajectories where the changing reward functions lead to interesting and divergent behavior, where interestingness is a property of the human's comprehension. To assess which trajectories to put before the human, I will consult the design guidance of human concept learning theory: for example, I will study whether directly incorporating variation patterns into the environments selected with RoCUS can help the human develop better uncertainty estimates about the preferences encoded in their own behavioral specifications, and of their comprehension of the robots' learned behaviors given these specifications. This study will be successful if, as a consequence of this interface and assisted reward design process, people are able to write reward functions which better align with their true intent. Ideally, this process would require less human effort and expertise, too.

Q2: How can we better interpret human's erroneous reward functions? Instead of trying to coax humans into writing better reward functions, another approach is to assume their ability to write reward functions is more or less fixed and imperfect. Given this assumption, we can study the types of errors which people make, and extract patterns in those errors—such as our previous observation that people commonly fail to reason about temporal discounting [2]. From these errors, we may be able to infer a better approximation of the human's intended reward function. This study would build on Inverse Reward Design methods [6], but in the inference step for approximating true reward functions, we would modify the model of the human expert to incorporate these known failure proclivities instead of assuming that the human is approximately optimal. This study will be successful if, as a consequence of this interpretation of the reward design problem, the inferred reward functions better align with humans' true intent.

REFERENCES

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

[2] Serena Booth, Bradley W. Knox, Julie Shah, Scott Niekum, Peter Stone, and Alessandro Allievi. 2023. The perils of trial-and-error reward design: misdesign through overfitting and invalid task specifications. *AAAI Conference on Artificial Intelligence*.

[3] Serena Booth, Sanjana Sharma, Sarah Chung, Julie Shah, and Elena L. Glassman. 2022. Revisiting human-robot teaching and learning through the lens of human concept learning. *Proceedings of the Human-Robot Interaction Conference (HRI)*.

[4] Serena Booth, Yilun Zhou, Ankit Shah, and Julie Shah. 2021. Bayes-TrEx: a Bayesian sampling approach to model transparency by example. *AAAI Conference on Artificial Intelligence*.

[5] Dedre Gentner and Linsey A Smith. 2013. Analogical learning and reasoning. *The Oxford handbook of cognitive psychology*, 668–681.

[6] Dylan Hadfield-Menell, Smitha Milli, Pieter Abbeel, Stuart J Russell, and Anca Dragan. 2017. Inverse reward design. *Advances in neural information processing systems*, 30.

[7] Jerry Zhi-Yang He and Anca D Dragan. 2021. Assisted robust reward design. *Conference on Robot Learning (CoRL)*.

[8] W Bradley Knox, Alessandro Allievi, Holger Banzhaf, Felix Schmitt, and Peter Stone. 2021. Reward (mis) design for autonomous driving. *arXiv preprint arXiv:2104.13906*.

[9] W Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. 2022. Models of human preference for learning reward functions. *arXiv preprint arXiv:2206.02231*.

[10] Ference Marton. 2014. *Necessary conditions of learning*. Routledge.

[11] David Silver, Satinder Singh, Doina Precup, and Richard S Sutton. 2021. Reward is enough. *Artificial Intelligence*, 299, 103535.

[12] Yilun Zhou, Serena Booth, Nadia Figueroa, and Julie Shah. 2021. RoCUS: robot controller understanding via sampling. *Conference on Robot Learning*.

[13] Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. 2022. Do feature attribution methods correctly attribute features? *AAAI Conference on Artificial Intelligence*.