Inferring the Human's Objective for Human-Robot Interaction

Joshua Hoegerman

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfilment of the requirements for the degree of

Master of Science
in
Mechanical Engineering

Dylan P. Losey, Chair
Kaveh Akbari Hamed
Erik E. Komendera

April 10, 2024
Blacksburg, Virginia

Keywords: Human Robot Interaction, Bayesian Inference, Shared Autonomy

ABSTRACT

Inferring the Human's Objective for Human-Robot Interaction

by

Joshua Hoegerman

This thesis discusses the use of Bayesian Inference in inferring over the human's objective for Human-Robot Interaction, more specifically, it focuses upon the adaptation of methods to better utilize the information for inferring upon the human's objective for Reward Learning and Communicative Shared Autonomy settings. To accomplish this, we first examine state-of-the-art methods for approaching Bayesian Inverse Reinforcement learning where we explore the strengths and weaknesses of current approaches. After which we explore alternative methods for approaching the problem, borrowing similar approaches to those of the statistics community to apply alternative methods to improve the sampling process over the human's belief. After this, I then move to a discussion on the setting of Shared Autonomy in the presence and absence of communication. These differences are then explored in our method for inferring upon an environment where the human is aware of the robot's intention and how this can be used to dramatically improve the robot's ability to cooperate and infer upon the human's objective. In total, I conclude that the use of these methods to better infer upon the human's objective significantly improves the performance and cohesion between the human and robot agents within these settings.

Inferring the Human's Objective for Human-Robot Interaction

Joshua Hoegerman

(GENERAL AUDIENCE ABSTRACT)

This thesis discusses the use of various methods to allow robots to better understand human actions so that they can learn and work with those humans. In this work we focus upon two areas of inferring the human's objective: The first is where we work with learning what things the human prioritizes when completing certain tasks to better utilize the information inherent in the environment to best learn those priorities such that a robot can replicate the given task. The second body of work surrounds Shared Autonomy where we work to have the robot better infer what task a human is going to do and thus better allow the robot to assist with this goal through using communicative interfaces to alter the information dynamic the robot uses to infer upon that human intent. Collectively, the work of the thesis works to push that the current inference methods for Human-Robot Interaction can be improved through the further progression of inference to best approximate the human's internal model in a given setting.

Acknowledgments

I would like to thank my advisor, Dylan P. Losey, for the many hours of guidance and research that enabled me to come as far as I have. I hope to take the lessons they have given and grow to contribute at least a fraction of the grand work they have done. I also acknowledge my thesis committee, for their efforts in improving this work. I appreciate the time I spent with the other members of the Collaborative Robotics Lab. For all the hours of intense discussion which helped to broaden my horizons, especially as I was getting started. It was through them I became the engineer I am today and it goes without saying this thesis would not have been possible without them. Most of all, I owe a special thanks to the friends whose company kept me sane during many of the harder weeks and my family whose encouragement kept me going through long uncertain years of this path.

Contents

List of Figures

Chapter 1

Introduction

As robotic agents become more and more integrated into human-adjacent applications like autonomous vehicles and assistive AI, there comes a need to improve our ability to integrate and match these agents to the environments they act within. Robots developed for these human-robot interactions strive to mediate the robot's desire to match itself to its idea of some task with the human's separate belief over that task [1, 2]. Although there exists a wide variety of methods to aid in inferring upon the human and then mediating this relationship, this work will focus on applications of Bayesian Inference in bridging the distance between these human and robot agents [3, 4, 5]. Specifically, I will address the use of Bayesian Inference for applications of more accurately replicating human tasks and in creating a more communicative Shared Autonomy system using contrasted informed vs uninformed human agent models.

This integration of robotic assistance is motivated in large part by the increasing number of robotic agents in ever more populated environments. If we want real-world integration of robotic systems into normal society, they must not only be able to meaningfully understand humans but also be able to know how to assist them. One meaningful way to accomplish this is the use of inference methods to interpret and understand human objectives from their actions in order to allow a robot to replicate this understanding for later assistance. For this work, we focus specifically on the use of Bayesian inference for this purpose. Bayesian inference being the process of using Bayes Theorem to judge an

example of human action in relation to a sampled belief. This has been used in inverse reinforcement learning to reproduce human tasks by replicating the underlying that inspired the human to complete that task such that the robot can take that understanding for wider applications of that task in alternative environments. However, just because a robot knows how to complete a human-relevant task does not mean it would know when to assist a human with this task. For this, we turn to a separate tool known as Shared Autonomy. For this, we have the human and robot working to operate the same system, imagine a robot arm helping with the aforementioned task; Shared Autonomy works to mediate *when* the human wants assistance with a task and *what* the human wants help to do. This work focuses on the improvement of these methods in learning and assisting in the human's objective for human-robot interaction.

For example, imagine the hundreds of thousands whose long-term mobility and independence are affected by neurological or physical disability, imagine just how impactful having a cooperative robot assistant could be in even simple day-to-day tasks like drinking or eating. So far the implementation of these systems in modern applications, while undoubtedly beneficial, suffers from several problems ranging from reliable integration of the system with the human participants to even simple concerns over its long-term accuracy in day-to-day use [1, 6]. These are very simple tasks that, due to the more personalized nature of each person's movement, require a more tailored solution to achieve any true long-term success. Therefore tools like Bayesian Inference and Shared Autonomy which allow for a greater degree of understanding and integration with the human agent are well suited for this application. These approaches also allow a certain amount of personalized understanding and adaptation to individuals allowing for a connected interaction.

Many modern implementations have bridged the gaps of some of these problems, however, the end result of these implementations requires simplistic tasks with limited user

autonomy. To increase the ability to reliably learn these tasks, we need to look into how they are learning these tasks. For the contents of this thesis, we focus on one of the more popular methods, Inverse Reinforcement learning, specifically Bayesian Inverse Reinforcement Learning (a.k.a. Bayesian Inference)[4, 5, 7]. For this method, we focus on seeing how the current iterative process [4, 5, 7] by examining the Bayesian normalizer for this application of Bayesian Inference. To date, current approaches have focused on one of three approaches for this normalizing function. Some works choose not to explicitly account for the normalizing function [8, 9, 10] using assumptions upon their choice of setting to reason this normalizer is invariant to sampled belief. Other alternatives use *sampling* to approximate the normalizer through a sampling approach taking some variation on the average normalizer for a given belief space [11, 12, 13, 14]. While other related works have also used the *maximum* trajectory reward in place of the normalizing function [15]. Each one of these approaches can work for a given setting but have their own individual strengths and weaknesses.

Notably however, when one approaches similar applications of sampling with Bayesian Inference in the statistics community, one can find a much larger variety of approaches for this problem [16, 17, 18, 19]. In Chapter 2, we discuss the approach of Bayesian inference for existing methods as compared to the introduction of novel improvements from the statistic community when applied to the problem of Bayesian Inverse Reinforcement Learning. Our study compares all approaches across multiple different settings varying the conditional dependence and proposed rationality in accordance with the expected affinity of each method with different applicable scenarios. We found that across both conditionally dependent and independent settings that the proposed sampling method significantly outperformed the given baselines. Chapter 2, which summarizes these results and their relevant study, was largely taken from [20].

Having explored methods regarding task recreation, this then motivated an exploration into how similar observations upon inference methods could be used in alternative robotic applications. This led to a study of how modern inference methods were used and applied across several studies, but due to a personal interest in assistive robotics, these studies led to a focus on Shared Autonomy. Prior work has made several interesting approaches for inferring the human's objective when working in a single system collaborative setting [3, 21, 22]. However, given our prior interest in Bayesian methods, we focus our attention on how similar Bayesian approaches have used the information of the environment to best infer upon the human's objective. This time, focusing on real-time goal inference as opposed to reward learning. In this setting, we wanted to look beyond just applying alternate normalizing methods as before and look at how information was being communicated and inferred upon from the human to the robot and vice versa. For this, we observed similar works in communicative shared autonomy [23, 24, 25, 26] to see how prior approaches had communicated variations on the robot's belief upon the human's goal. While there were a number of approaches covering how to communicate this belief [23, 24, 25, 27], there was less available work on how this shifted how the robot should infer upon the human's goal.

Thus, for Chapter 3, we focus on how to align this learning with communication in Shared Autonomy which is taken from [28]. For this paper, we first explored the variation in performance in the absence and presence of communication to see how humans adapted to explicit knowledge of the robot's proposed belief. We then investigated these results, which showed that the humans were not only able to give clearer instructions to the robot agent but were more likely to give up control upon seeing an aligned belief. We used these results to develop a basic algorithm for transferring control for improved inference in settings with communication. This was then tested to measure both human performance and preference

with our method as opposed to the relevant baseline [21]. The results measured within this study highlighted not only a significant user preference for our method given its ability to translate the explicit signal of the human in reaction to the now explicit communication of the robot's belief, but also a significant performative difference in the amount of effort and time required for the user to align the system with their belief. With these results, we conclude that acknowledging this difference in information dynamics with and without communication for the given inference method provides significant benefit given the updated human understanding of the human-robot alignment for the given system.

In summary, this thesis explores the inference of human tasks and goals to better take advantage of the inherent information held within the system. In Chapter 2, we will focus upon the use of alternative normalizing functions to better infer upon the human's task. Although it is not included in this thesis, work was also done for the use of shared autonomy in enabling soft robotic systems for those presented in [29] which inspired me to continue this work for Shared Autonomy application. Thus, in Chapter 3, we will move to focus upon aligning the inference of the human's goal in a shared autonomy setting with additional information gained through a more communicative setting. I claim that, in both of these cases, the additional consideration for additional information in the system should change how one infers upon the human objective to best infer upon the human's intent.

Chapter 2

Reward learning with Intractable Normalizing Functions

2.1 Introduction

Consider a robot arm learning from demonstrations (see Figure 2.1). The human guides the robot through example trajectories and the robot tries to infer the human's objective based on their demonstrations. For instance, here the robot arm should learn to slide the box across the table.

State-of-the-art research often tackles this *reward learning* problem using *Bayesian inference* [4, 5, 7]. The trajectories provided by the human are observations of their latent objective (i.e., their reward function) and the robot recovers a distribution over the rewards by applying Bayes' theorem. Put intuitively: the robot infers rewards under which the human's demonstrations are approximately optimal. Unfortunately, reward learning in continuous spaces is *doubly intractable*. The robot must normalize across the space of trajectories (i.e., what other demonstrations could the human have provided?) and over the space of rewards (i.e., what else could the human be optimizing for?). Today's robots recognize that they cannot compute these normalizers exactly and so they make a variety of *approximations* [8, 9, 10, 11, 13, 14, 15, 30, 31, 32].

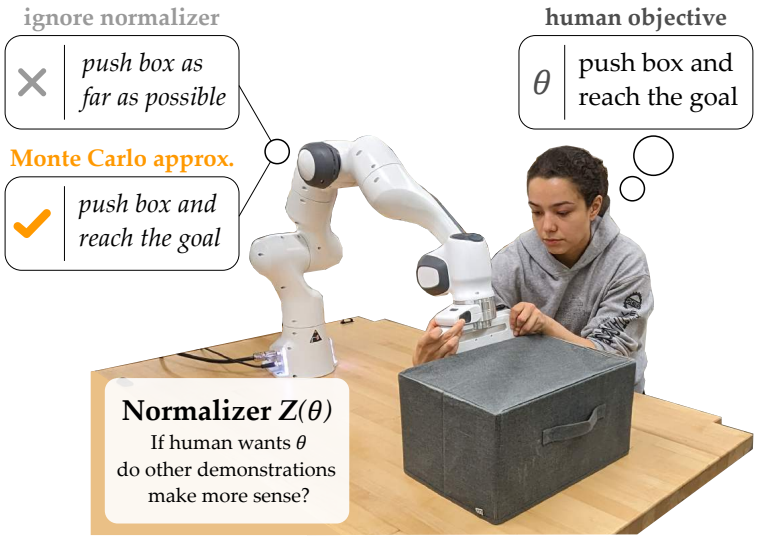

Figure 2.1: To infer the user's reward, (i.e., objective) robots must compare the human's actual demonstration to other demonstrations the human *could* have provided. Computing this normalizer makes Bayesian reward learning doubly-intractable and ignoring the normalizer entirely can lead to incorrect robot learning. We propose a Monte Carlo method for a more accurate approximation.

Inaccurate approximations of the normalizing function can lead to incorrect inference: instead of learning what the human meant, the robot learns to perform different and potentially undesirable tasks. Refer back to Figure 2.1: here ignoring the normalizer can cause the robot to knock the box off the table.

In this work, we focus on inferring the human's reward from demonstrations or corrections. Our insight is that:

We can enable asymptotic reward learning by leveraging novel sampling methods from the statistics community.

Our resulting framework applies to problems where the robot has a predictive model of the environment. Given a sequence of independent or interconnected human demonstrations, we enable robots to accurately learn the human's reward despite doubly intractable normalizing functions. Returning to Figure 2.1: our robot learns to push the box correctly

using the same amount of data as the baseline.

Overall, we make the following contributions:

Comparing Approximations We theoretically and experimentally analyze three existing classes of methods for approximating the normalizer during Bayesian reward learning.

Introducing Double Metropolis-Hastings Sampling We apply novel statistics research to develop a Double MH algorithm for Bayesian reward learning in continuous spaces.

Learning from Demonstrations and Corrections Across simulations and user studies, we show that Double MH results in more accurate reward learning from both conditionally independent (e.g., separate) demonstrations and conditionally dependent (e.g., interconnected) corrections.

2.2 Related Work

IRL. Our problem is an instance of Inverse Reinforcement Learning (IRL) where the robot tries to recover the reward that the human wants the robot to optimize for [33]. Related work formalizes this as *Bayesian inference* [4, 5, 7]. Wherein, given a prior distribution and the human's inputs (e.g., state-action pairs [5] or trajectories [7] provided by the human), the robot arm infers a posterior over the space of possible rewards. In order to connect the human's inputs to rewards, today's robots model the human as a noisily rational teacher [4, 34, 35]. Unfortunately, the human model becomes *intractable* in continuous spaces; without this accurate human model, the robot cannot correctly infer the human's reward.

As we will demonstrate in Section 2.3, the key challenge is the *normalizing function* of the human model. Within this normalizing function, the robot integrates over the space of all

possible human inputs (e.g., their actions or trajectories). This normalizing function makes reward learning doubly intractable: we can use standard Markov chain Monte Carlo (MCMC) methods to eliminate a part of this problem [5, 8, 9], but the normalizing function still remains. Existing IRL algorithms have developed different approaches for dealing with the normalizer. Some works may not explicitly account for the normalizing function [8, 9, 10], and there are settings where *ignoring* this normalizer is reasonable. Others use *sampling* to approximate the normalizer: this includes sampling uniformly from the trajectory space [11, 12], sampling trajectories close to the human's inputs [30], or using importance sampling to convert between the robot's trajectory distribution and a uniform distribution over trajectories [13, 14]. Finally, related works have also used the *maximum* trajectory reward in place of the normalizing function [15]. This substitution can be justified using Laplace's approximation [32, 36].

To summarize, prior work on Bayesian IRL uses ignore, sampling, and maximum approaches to estimate the normalizing function and infer the human's reward. In this work, we theoretically and experimentally compare these different approaches to understand their relative advantages.

Approximate Bayesian Inference Within the robotics community, we have developed a variety of techniques for learning with an intractable normalizing function. But what about work *outside* of robotics? Statistics research has recently proposed several Markov chain Monte Carlo (MCMC) algorithms for Bayesian inference in the presence of intractable normalizing functions [16, 17, 18, 19]. These approaches modify techniques such as Metropolis-Hastings sampling to obtain computationally efficient and accurate algorithms for inferring *doubly intractable* posterior distributions. In this work, we apply recent breakthroughs from the statistics community to propose a new method for Bayesian reward learning.

2.3 Problem Formulation

We consider settings where a robot arm is using Bayesian inference to learn from human examples. The human teacher knows what task the robot should perform. More specifically, the human has in mind a reward function that the robot should optimize, and the robot is trying to infer this reward from the human's data. The human might provide complete examples of their desired behaviour (i.e., demonstrations), or they could just modify snippets of the robot's existing motion (i.e., corrections). We recognize that humans are not perfect teachers: when showing the robot how to carry a glass of water, the human may not have enough time or ability to meticulously orchestrate every joint. The robot views the human as a *noisily rational* agent that approximately maximizes their reward.

Task and Reward This problem is an instance of a Markov decision process (MDP) where the robot does not know the reward function. Let the MDP be a tuple $M = \langle \mathcal{S}, \mathcal{A}, f, r, T \rangle$ where $s \in \mathcal{S}$ is the system state and $a \in \mathcal{A}$ is the robot's action. For example, s could be the arm's joint position and the pose of the cup, and a could be the robot's joint velocity. At timestep t, the system transitions to a new state according to the deterministic dynamics $s^{t+1} = f(s^t, a^t)$. The task ends after a total of T timesteps. Let $\xi = (s^0, \dots s^T)$ be the robot's *trajectory*, i.e., the sequence of visited states across T timesteps, and let Ξ be the set of possible trajectories.

During every timestep, the robot receives a scalar reward $r(s)$. Remember that the robot *does not* know the desired reward function. Without loss of generality, we will write the reward as $r(s, \theta)$, where vector $\theta \in \mathbb{R}^d$ captures the aspects of the reward function that the robot does not know. For example, in related works [7, 33] the reward function is often a linear combination of features such that $r(s, \theta) = \theta \cdot \phi(s)$. Here $\phi : \mathcal{S} \to \mathbb{R}^d$ is a d-length feature vector that captures task-relevant aspects of the state (e.g., the distance from the

table, the orientation of the cup), and θ determines the relative importance of these features. Across an entire trajectory, the robot's cumulative reward is: $R(\xi, \theta) = \sum_{s \in \xi} r(s, \theta)$. If the reward function is a linear combination of features, this simplifies to: $R(\xi, \theta) = \theta \cdot \sum_{s \in \xi} \phi(s) = \theta \cdot \Phi(\xi)$.

Human Data The robot is attempting to learn the reward (and more precisely, the unknown parameters θ) from human examples. Let $\mathcal{D} = \{\xi_1, \ldots \xi_K\}$ be a dataset of K trajectories provided by the human expert. Our approach is not tied to any specific way of gathering this dataset. The trajectories could be collected *offline* as the human kinesthetically guides the robot through task demonstrations [10, 13, 30]. Although, we could also add improved trajectories *online* as the human corrects the robot arm [11, 15, 37]. In either case, we aggregate the human's trajectories into the dataset $\mathcal{D}$.

For simplicity, we now assume that the human teacher only provides a *single* trajectory, i.e., $\mathcal{D} = \xi$. In Section 2.5 we will extend our analysis to a dataset of K trajectories.

Bayesian Inference The robot infers the parameters θ from the human's trajectory ξ. Let $P(\theta \mid \xi)$ denote the probability that the human is optimizing for reward parameters θ given the input trajectory ξ. Applying Bayes' theorem:

$$P(\theta \mid \xi) \propto P(\xi \mid \theta) \cdot P(\theta) \tag{2.1}$$

where $P(\theta)$ is the prior and $P(\xi \mid \theta)$ is the likelihood function. Intuitively, $P(\xi \mid \theta)$ expresses how likely it is (from the robot's perspective) that the human provides trajectory ξ given the human is optimizing for reward parameters θ.

Human Model The likelihood function $P(\xi \mid \theta)$ is a human model: it tries to capture how the human teacher maps their hidden objective to an example trajectory. Prior work in behavioral economics [35], cognitive science [34], and reward learning [4] suggests that

humans are *noisily rational* agents. These humans are not perfect: instead of always choosing the best possible trajectory, noisily rational humans are exponentially more likely to select behaviors with higher rewards. Under the noisily rational model:

$$P(\xi \mid \theta) = \frac{\exp\big(\beta \cdot R(\xi, \theta)\big)}{\int_{\Xi} \exp\big(\beta \cdot R(\xi', \theta)\big)\ d\xi'} \tag{2.2}$$

Where $\beta \in [0, \infty)$ is a hyperparameter set by the designer. As $\beta \to 0$ each trajectory becomes equally likely and the robot treats the human as a random agent. At the other extreme, as $\beta \to \infty$ the human is only likely to input optimal trajectories and the robot views the human as perfectly rational.

Normalizing Function The numerator of Equation (2.2) is straightforward: we simply substitute ξ and θ into the reward function and evaluate. But once we find $\exp\big(\beta \cdot R(\xi, \theta)\big)$, how good (i.e., how likely) is that trajectory? We need a sense of scale to understand the relative reward for ξ as compared to the alternatives — perhaps there is another trajectory ξ' that achieves a much higher reward. This is where the denominator of Equation (2.2) comes in. The denominator is a *normalizing function* that integrates over the continuous space of possible trajectories $\xi' \in \Xi$ given reward parameters θ. We refer to the normalizing function as $Z(\theta)$:

$$Z(\theta) = \int_{\Xi} \exp\big(\beta \cdot R(\xi', \theta)\big)\ d\xi' \tag{2.3}$$

The normalizing function $Z(\theta)$ serves to calibrate our human model. Importantly, $Z(\theta)$ can be different for different values of θ. We will show examples of how Z changes (or does not change) as a function of θ in the following sections.

Summary To infer the human's reward through Bayesian inference we need to find $P(\xi \mid \theta)$ in Equation (2.1). But to get $P(\xi \mid \theta)$ we first must be able to solve

Equation (2.3), and this normalizing function is *intractable* when Ξ is a continuous manifold [13]. This leads to our core problem: how should robots approximate (or avoid) the normalizing function $Z(\theta)$ when learning from human data?

2.4 Approximating the Normalizer

In this section, we analyze three state-of-the-art approaches for dealing with the normalizing function in Bayesian inference. In Section 2.4.1 we prove that robots can completely ignore the normalizing function if Z does not depend on θ; we also identify the necessary conditions for this special case when the reward is a linear combination of features. Moving beyond this special case, we next explore two methods for approximating $Z(\theta)$. In Section 2.4.2 we discuss a sampling approach and in Section 2.4.3 we use the maximum value in place of the normalizer. We prove that the maximum approach will match or outperform the sampling approach as $\beta \to \infty$ in our noisily rational human model.

Working Example. We first introduce a simplified example to illustrate the analysis throughout this section. Consider a robot arm that is learning how to carry a cup. The robot can hold the cup at any angle between 0 radians (horizontal) and $\pi/2$ radians (vertical). The human teacher inputs a trajectory $\xi = s$ where they specify a single orientation of the cup. The reward is: $r(s, \theta) = -5\cos(\theta) \cdot (s + 1) - \sin(\theta) \cdot (\pi/2 - s)$. Based on the human's input ξ, the robot is trying to determine whether (a) $\theta = 0$ and the robot should hold the cup horizontally at angle $s = 0$, or whether (b) $\theta = \pi/2$ and the robot should hold the cup vertically at angle $\pi/2$. Let the robot have a uniform prior over these two possible reward parameters. Applying Equations (2.1)-(2.3), the robot's belief

that $\theta = 0$ is:

$$P(0 \mid \xi) = \frac{\exp\big(\beta R(\xi, 0)\big)}{\exp\big(\beta R(\xi, 0)\big) + \frac{Z(0)}{Z(\pi/2)} \cdot \exp\big(\beta R(\xi, \frac{\pi}{2})\big)} \tag{2.4}$$

In this simple example, numerical integration is possible and we can exactly find $Z(0)$ and $Z(\pi/2)$. Plugging these exact values into Equation (2.4) gives us the *ideal* belief. Put another way, this is what the robot *should* learn. We will compare this ideal result to approaches that approximate the normalizer.

2.4.1 Ignoring the Normalizing Function

In some settings, it may be reasonable to ignore the normalizing function altogether. Methods such as [8, 9, 10] use MCMC sampling to cancel out the partition function $P(\xi)$, but they do not explicitly account for the normalizing function within $P(\xi \mid \theta)$. In our working example for instance, these approaches may omit the Z terms from Equation (2.4).

Proposition 1. *We can ignore the normalizing function when Z does not depend on θ.*

Proof. Consider a problem setting where Z is a constant, i.e., where $Z(\theta_i) = Z(\theta_j)$ for any choice of θ_i and θ_j. When we substitute Equation (2.2) back into Equation (2.1) to infer the reward, both the numerator and denominator are multiplied by Z and this normalizing constant cancels out:

$$P(\theta \mid \xi) = \frac{Z}{Z}\left(\frac{\exp\big(\beta \cdot R(\xi', \theta)\big) \cdot P(\theta)}{\int_\Theta \exp\big(\beta \cdot R(\xi, \theta')\big) \cdot P(\theta')\, d\theta'}\right) \tag{2.5}$$

Looking specifically at the working example in Equation (2.4), if $Z(0) = Z(\pi/2)$ then the Z terms cancel. □

So far our analysis shows that we can ignore Z without any loss in performance *if* the normalizer is a constant. But when is this the case? To answer this question we focus on a

common framework where the reward function is a linear combination of features, $R(\xi, \theta) = \theta \cdot \Phi(\xi)$. We find that:

Proposition 2. *If* $R(\xi, \theta) = \theta \cdot \Phi(\xi)$ *and* $\theta \in \mathbb{R}^d$ *is a d-dimensional unit vector,* Z *does not depend on* θ if and only if *the feature space* $\Phi(\Xi)$ *is a sphere centered at zero with radius* $\sigma \geq 0$*, i.e.,* $\Phi(\Xi) = \{v \in \mathbb{R}^d : \|v\| = \sigma\}$.

Proof. Let θ_i and θ_j be two arbitrary unit vectors. Consider Equation (2.2) with $\beta \in [0, \infty)$. For the integrals $Z(\theta_i)$ and $Z(\theta_j)$ to be equal, for every $\xi \in \Xi$ there must exist some $\xi' \in \Xi$ such that: $\theta_i \cdot \Phi(\xi) = \theta_j \cdot \Phi(\xi')$. This is only satisfied when $\Phi(\Xi)$ is proportional to a unit sphere. □

In practice, it is challenging to ensure the feature space is a sphere. Not only do we need the average of each individual feature to be zero, but the combination of features must always have the same radius. Consider Figure 2.1 where the human is teaching the robot arm: along the human's trajectory they might minimize the robot's height and orientation, resulting in a feature vector $\Phi(\xi)$ where each element is close to zero. Alternatively, the human might move the robot far from the table while changing the orientation, leading to a $\Phi(\xi)$ where each element is close to one. The magnitude of these two feature vectors is different — and thus we *cannot* apply Proposition 2 and ignore the normalizer.

In Propositions 1 and 2 we have identified a special case where the robot does not need to evaluate Z. However, for common settings where Z *is* a function of θ, ignoring the normalizing function results in errors in the robot's learning. See our working example in Figure 2.2 where we plot the error between Equation (2.4) with and without $Z(\theta)$.

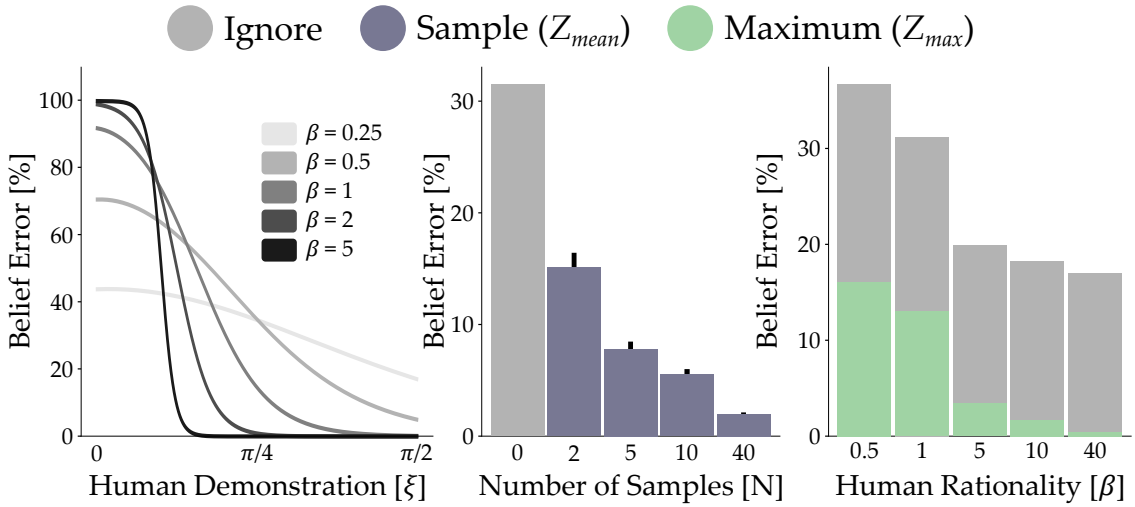

Figure 2.2: Approximating the normalizer in our working example. *Belief Error* is the difference between evaluating Equation (2.4) with the exact $Z(\theta)$ and Equation (2.4) with approximations for $Z(\theta)$. The lower error indicates the robot learned the correct belief. (Left) The human provides demonstrations ξ of carrying the cup at different angles. If we **ignore** the normalizer, as $\beta \to \infty$ the robot always learns to carry the cup vertically, even when the human wants the opposite. (Middle) Here we left $\beta = 1.0$. As the number of **samples** for Z_{mean} increases, the belief error converges to zero. (Right) As $\beta \to \infty$ the error with the **maximum** approach converges to zero.

2.4.2 Approximating the Normalizer with Sampling

Instead of ignoring the normalizing function, next we will estimate $Z(\theta)$. One approach is to approximate the integral in Equation (2.3) through sampling [11, 12, 13, 14, 30, 31]. Let $\{\xi_1, \ldots, \xi_N\}$ be N trajectories sampled uniformly at random from the trajectory space Ξ. We use these samples to approximate the mean value of $\exp\left(\beta \cdot R(\xi, \theta)\right)$ as shown below:

$$Z_{mean}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \exp\left(\beta \cdot R(\xi_i, \theta)\right) \tag{2.6}$$

Applying the law of the unconscious statistician, this estimate noisily converges to the actual mean as the number of ξ samples increases. It may seem unintuitive at first that we are estimating the mean and not $Z(\theta)$. However, $Z(\theta)$ is equal to this mean multiplied by a volumetric constant that does not depend on θ; because the constant cancels out during

Bayesian inference, we only need the mean.

We test this sampling approach on our working example in Figure 2.2. Compared to a robot that ignores the normalizing function, attempting to estimate $Z(\theta)$ leads to more accurate learning. But we do recognize that the sampling approach comes with an assumption: specifically, we now assume that the robot knows the space of possible trajectories Ξ.

2.4.3 Approximating the Normalizer as the Maximum

Other state-of-the-art algorithms use a maximum value in place of the normalizing function [15, 32, 36]. These approaches find the maximum of the numerator in Equation (2.2), and then treat this maximum as the denominator:

$$Z_{max}(\theta) = \max_{\xi \in \Xi} \ \exp\left(\beta \cdot R(\xi, \theta)\right) \tag{2.7}$$

Intuitively, scaling by Z_{max} makes sense because it ensures that the resulting $P(\theta \mid \xi)$ is always less than or equal to one. Recall that for the sampling approach Z_{mean} converges as the number of ξ samples increases. Interestingly, we find an analogous convergence for the maximum approach:

Proposition 3. *The error between Bayesian inference with Z_{max} and an ideal robot that uses Z converges to zero as the human becomes increasingly optimal, i.e., as $\beta \to \infty$.*

Proof. For any given θ, let there be a single trajectory ξ^* that maximizes the human's reward such that $R(\xi^*, \theta) > R(\xi, \theta)$ for all $\xi \in \Xi \setminus \xi^*$. Use numerical integration to estimate Z:

$$Z(\theta) \doteq C\left(Z_{max}(\theta) + \sum_{i=1}^{N} \ \exp\left(\beta \cdot R(\xi_i, \theta)\right)\right) \tag{2.8}$$

where C is a volumetric constant that cancels out in Bayesian inference, and $\{\xi_1, \ldots, \xi_N\}$ are N non-optimal trajectories sampled uniformly at random from $\Xi \setminus \xi^*$. Taking the limit as $\beta \to \infty$, and remembering that $R(\xi^*, \theta) > R(\xi, \theta)$, we have that $Z_{max}(\theta)$ dominates the remaining terms. Accordingly, as $\beta \to \infty$ Equation (2.8) converges to $C \cdot Z_{max}(\theta)$, and the difference between a robot that learns using $Z(\theta)$ and a robot that learns using Z_{max} converges to zero. □

We apply Proposition 3 to our working example in Figure 2.2. As expected, using Z_{max} as the normalizing function becomes increasingly accurate as $\beta \to \infty$. But now that we have two different methods for approximating the normalizer, we are left with a decision: when should designers use Z_{mean} and when should designers use Z_{max}? The answer to this question varies as the problem setting and reward function change. However, we do find a general trend:

Proposition 4. *As $\beta \to \infty$ in the human model, robots learn an equal or more accurate estimate of $P(\theta \mid \xi)$ using Bayesian inference with Z_{max} instead of Z_{mean}.*

Proof. From Proposition 3 we already know that Z_{max} converges to ideal performance as $\beta \to \infty$. It only remains to evaluate the performance of Z_{mean}. Recall from Equation (2.6) that Z_{mean} samples N trajectories from space Ξ. Because N is a finite number, there will be cases when the robot does not sample the optimal trajectory ξ^*. Compare the numerical integration in Equation (2.8) to the sampled mean in Equation (2.6). If the robot does not sample ξ^* in Equation (2.6), then as $\beta \to \infty$ Equation (2.6) is not necessarily proportional to Equation (2.8), where Z is dominated by the exponentiated reward of ξ^*. Because Z_{mean} is not necessarily proportional to Z, a robot that learns using Z_{mean} may not match the performance of an ideal robot that learns with Z. □

We demonstrate Proposition 4 in Figure 2.3. For *lower values* of β we find that

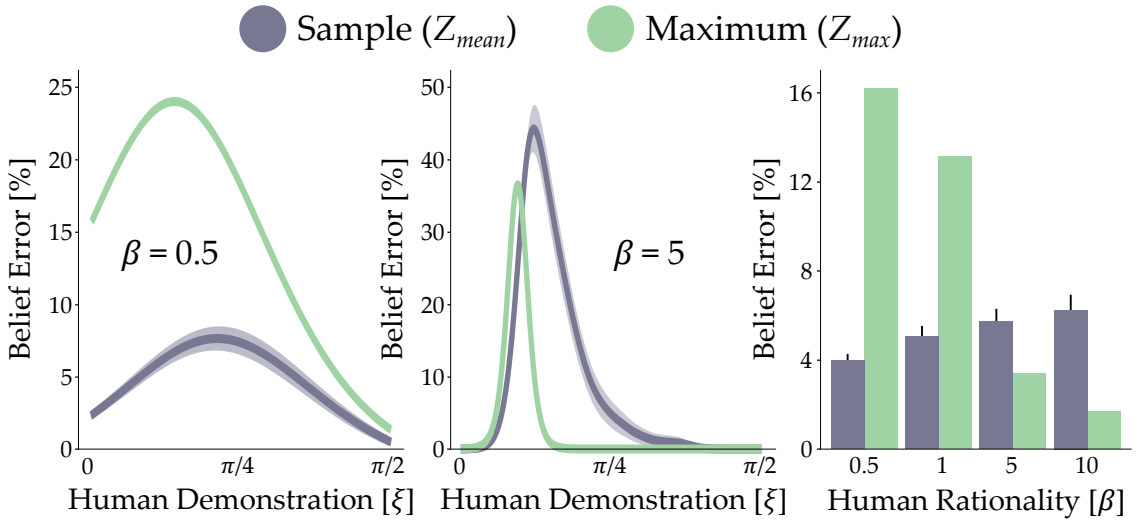

Figure 2.3: Comparing **sampling** and **maximum** approaches in our working example. Remember that β from Equation (2.2) captures how close-to-optimal the human is. (Left) Error when $\beta = 0.5$. (Middle) Error when $\beta = 5$. (Right) For lower values of β we find that the sampling approach is more accurate. However, as $\beta \to \infty$ the maximum approach leads to less error. For **sampling** we perform 100 separate runs where Z_{mean} samples $N = 10$ trajectories each run; the shaded region and bars show standard error.

approximating the normalizer with *sampling* outperforms the maximum approach. By contrast, for *higher values* of β using the *maximum* as the normalizing function results in more accurate learning. The exact trade-off point is problem-specific, but this general trend holds across our theoretical analysis and experimental results.

2.5 Scaling up with Metropolis-Hastings Sampling

Now that we've analyzed the normalizing function when the human only provides a single trajectory, i.e., when $\mathcal{D} = \xi$. In this section, we scale up to general cases where the robot is learning from a dataset $\mathcal{D}$ of K trajectories. As we scale up, we recognize that the space of rewards Θ is *continuous*. In our working example we assumed that the human wanted the robot to hold the cup either horizontally or vertically; but, more generally, the human may want the robot to hold the cup at any angle. To enable Bayesian inference in continuous

reward spaces, we turn to *Metropolis-Hastings (MH) sampling* [38]. We first formulate conditionally independent and dependent reward learning from multiple trajectories (Section 2.5.1) and then combine approaches for approximating the normalizer with MH sampling (Section 2.5.2). In Section 2.5.3 we introduce the *Double MH algorithm* for Bayesian reward learning.

2.5.1 Learning from Multiple Trajectories

Let $\mathcal{D} = \{\xi_1, \ldots, \xi_K\}$ be a dataset of K trajectories input by the human teacher. The probability that the human has in mind reward θ given dataset $\mathcal{D}$ is:

$$P(\theta \mid \mathcal{D}) \propto P(\mathcal{D} \mid \theta) \cdot P(\theta) \tag{2.9}$$

Similar to Equation (2.1), here $P(\theta)$ is the prior over the space of rewards and $P(\mathcal{D} \mid \theta)$ is the likelihood the human inputs $\mathcal{D}$ given their reward is parameterized by θ.

Conditionally Independent If the human selects each trajectory separately then the human's inputs are conditionally *independent*. For instance, consider a human that repeatedly demonstrates a task to the robot: each demonstration depends on their reward θ, but the human does not reason over ξ_i when selecting ξ_j [4, 7, 10]. When the trajectories are conditionally independent Equation (2.9) reduces to:

$$P(\theta \mid \mathcal{D}) \propto P(\theta) \cdot \prod_{i=1}^{K} P(\xi_i \mid \theta) \tag{2.10}$$

Plugging in our human model from Equation (2.2), we reach:

$$P(\theta \mid \mathcal{D}) \propto \frac{\exp\left(\beta \cdot \sum_{i=1}^{K} R(\xi_i, \theta)\right) \cdot P(\theta)}{Z(\theta)^K} \tag{2.11}$$

where $R(\xi, \theta) = \sum_{s \in \xi} r(s, \theta)$ is the total reward along input ξ and $Z(\theta)$ is the normalizing function from Equation (2.3).

Conditionally Dependent Alternatively, if the human provides multiple interconnected trajectories the human's inputs are conditionally *dependent*. For example, imagine a human that iteratively improves the robot's motion by making small corrections: the human's input ξ_j will depend on the human's reward but also on the distance between ξ_j and the previous trajectory ξ_i [15]. In this case, we cannot simplify Equation (2.9). Instead, we define an augmented human model:

$$P(\mathcal{D} \mid \theta) = \frac{\exp\left(\beta \cdot \mathbf{R}(\mathcal{D}, \theta)\right)}{\int_{\mathbb{D}} \exp\left(\beta \cdot \mathbf{R}(\mathcal{D}', \theta)\right) \, d\mathcal{D}'} \tag{2.12}$$

where $\mathbf{R}$ is the total reward over dataset $\mathcal{D}$. Going back to our example of a human that makes small improvements, $\mathbf{R}$ could be [15]: $\mathbf{R}(\mathcal{D}, \theta) = \sum_{i=2}^{K} R(\xi_i, \theta) - \|\xi_i - \xi_{i-1}\|^2$. Looking at the denominator of Equation (2.12), for conditionally dependent trajectories we need to normalize over the entire dataset $\mathcal{D}$ rather than the individual inputs ξ:

$$\mathbf{Z}(\theta) = \int_{\mathbb{D}} \exp\left(\beta \cdot \mathbf{R}(\mathcal{D}', \theta)\right) \, d\mathcal{D}' \tag{2.13}$$

Here $\mathbb{D}$ is the space of possible datasets $\mathcal{D}$. To find $P(\theta \mid \mathcal{D})$ and perform Bayesian inference we plug Equation (2.12) with normalizing function $\mathbf{Z}(\theta)$ back into Equation (2.9).

2.5.2 Metropolis-Hastings Sampling

Regardless of whether the human's inputs are conditionally independent or conditionally dependent, we want to use dataset $\mathcal{D}$ to infer the reward parameters θ. This leads us back to the posterior distribution $P(\theta \mid \mathcal{D})$. To evaluate $P(\theta \mid \mathcal{D})$ in Equation (2.9) we have to deal with *another* normalizer; specifically, the denominator $P(\mathcal{D}) = \int_{\Theta} P(\mathcal{D} \mid \theta) \cdot P(\theta) d\theta$.

When Θ is a discrete space (e.g., in our working example where the human wants the cup either horizontal or vertical) we can compute this denominator and find the probability of each $\theta \in \Theta$. But when Θ is continuous, Bayesian inference becomes *doubly intractable* and we cannot typically find closed-form expressions for $P(\theta \mid \mathcal{D})$. Instead, the robot learner uses the MH algorithm to *sample* values of θ from the non-normalized posterior, i.e., $\theta \sim P(\cdot \mid \mathcal{D})$.

Algorithm 1 Bayesian Reward Learning with Normalizer Approximation

1: $\theta \leftarrow$ sample from $P(\theta)$
2: **for** each iteration **do**
3: $\quad \theta' \leftarrow$ sample from Θ near θ
4: $\quad$ *Conditionally Independent:*
5: $\quad \frac{P(\theta'|\mathcal{D})}{P(\theta|\mathcal{D})} \leftarrow \frac{\exp\left(\beta \cdot \sum_{i=1}^{K} R(\xi_i, \theta')\right) \cdot Z(\theta)^K \cdot P(\theta')}{\exp\left(\beta \cdot \sum_{i=1}^{K} R(\xi_i, \theta)\right) \cdot Z(\theta')^K \cdot P(\theta)}$
6: $\quad$ *Conditionally Dependent:*
7: $\quad \frac{P(\theta'|\mathcal{D})}{P(\theta|\mathcal{D})} \leftarrow \frac{\exp\left(\beta \cdot \mathbf{R}(\mathcal{D}, \theta')\right) \cdot \mathbf{Z}(\theta) \cdot P(\theta')}{\exp\left(\beta \cdot \mathbf{R}(\mathcal{D}, \theta)\right) \cdot \mathbf{Z}(\theta') \cdot P(\theta)}$
8: $\quad$ **if** $P(\theta' \mid \mathcal{D}) / P(\theta \mid \mathcal{D}) > \alpha \sim \mathcal{U}[0, 1]$ **then** $\theta \leftarrow \theta'$
9: Return θ

MH Algorithm. We combine MH sampling with methods for approximating the normalizer in Algorithm 1. At each iteration, we propose a new reward parameter θ'. The robot then compares the probability of θ' with the probability of θ, and accepts θ' with probability $\min\{1, P(\theta' \mid \mathcal{D}) / P(\theta \mid \mathcal{D})\}$. Similar to our analysis in Section 2.4.3, any terms that do not depend on θ cancel out when we divide the posteriors. Each different approach for approximating the normalizer uses a different method for selecting Z or $\mathbf{Z}$.

- *Ignore:* Set $Z(\theta) = 1$
- *Sampling:* Approximate $Z(\theta)$ using Equation (2.6)
- *Maximum:* Approximate $Z(\theta)$ using Equation (2.7)

These same equations extend to $\mathbf{Z}$, but now the robot samples from the space of datasets $\mathbb{D}$ instead of trajectories Ξ.

2.5.3 Reward Learning with Double MH Sampling

In addition to the ignore, sample, and maximum methods from Section 2.4.3, we can now introduce one final approach for approximating the normalizer. Standard MH approaches divide $P(\theta' \mid \mathcal{D})$ by $P(\theta \mid \mathcal{D})$ so that any terms that do not depend on θ are cancelled out. Here we take this concept one step further through *double* MH sampling [16]. At a high level, the double MH algorithm introduces an auxiliary variable such that, when we divide the posteriors, $Z(\theta) \cdot Z(\theta')$ appears in both the numerator and denominator, enabling us for the first time to cancel out the normalizing function. This shifts our problem: instead of approximating $Z(\theta)$, we need a method for generating the auxiliary variable.

Algorithm 2 Bayesian Reward Learning with Double MH

$\theta \leftarrow$ sample from $P(\theta)$
for each iteration **do**
  $\theta' \leftarrow$ sample from Θ near θ
  $\xi' \sim \mathcal{T}(\mathcal{D}, \theta')$ ▷ inner sampler in Algorithm 3
  $\frac{P(\theta'|\mathcal{D})}{P(\theta|\mathcal{D})} \leftarrow \frac{\exp \beta \cdot \sum_{i=1}^{K} \Big(R(\xi_i,\theta')/R(\xi_i,\theta)\Big) \cdot P(\theta')}{\exp \beta \cdot \sum_{i=1}^{K} \Big(R(\xi',\theta')/R(\xi',\theta)\Big) \cdot P(\theta)}$
  if $P(\theta' \mid \mathcal{D})/P(\theta \mid \mathcal{D}) > \alpha \sim \mathcal{U}[0,1]$ **then** $\theta \leftarrow \theta'$
Return θ

Algorithm 3 Inner Sampler for Double MH

Input dataset $\mathcal{D}$ and reward parameter θ
$\xi \leftarrow$ sample trajectory from $\mathcal{D}$
for each iteration **do**
  $\xi' \leftarrow$ sample from Ξ near ξ
  if $e^{\beta\Big(R(\xi',\theta)-R(\xi,\theta)\Big)} > \alpha \sim \mathcal{U}[0,1]$ **then** $\xi \leftarrow \xi'$
Return ξ

Double MH Algorithm. We outline Double MH sampling for Bayesian reward learning

in Algorithms 2 (outer sampler) and 3 (inner sampler). For clarity we focus on *conditionally independent* trajectories; it is straightforward to modify this pseudocode for the *conditionally dependent* case. At each iteration, the outer sampler proposes a new θ'. The inner sampler then inputs this θ' and generates a trajectory ξ' from the distribution $P(\xi \mid \theta')$. The new trajectory — which is sampled, and does not come from human demonstrations — is the auxiliary variable. We leverage this auxiliary variable to cancel out the normalizing functions and avoid computing $Z(\theta)$. Specifically, the robot accepts θ' with probability:

$$\min\left\{1, \frac{P(\theta') \cdot \left(\prod_{i=1}^{K} P(\xi_i \mid \theta') P(\xi' \mid \theta)\right)}{P(\theta) \cdot \left(\prod_{i=1}^{K} P(\xi_i \mid \theta) P(\xi' \mid \theta')\right)}\right\} \tag{2.14}$$

Substituting in our human model and normalizing function:

$$\min\left\{1, \frac{P(\theta') Z(\theta)^K Z(\theta')^K \cdot e^{\beta \sum_{i=1}^{K} \left(R(\xi_i, \theta') + R(\xi', \theta)\right)}}{P(\theta) Z(\theta')^K Z(\theta)^K \cdot e^{\beta \sum_{i=1}^{K} \left(R(\xi_i, \theta) + R(\xi', \theta')\right)}}\right\}$$

Hence, the normalizing functions cancel out and we are left with the acceptance rule in Algorithm 2. We note that this Double MH approach also extends to learning rewards from state-action pairs (instead of trajectories) if we replace $R(\xi, \theta)$ with the Q-function (i.e., the cost-to-go).

Parameters. In our approach there are three main parameters for the designer to tune: a) the number of iterations in Algorithm 2, b) the number of iterations in Algorithm 3, and c) the rationality constant β. Increasing the number of outer and inner samples increases the expected accuracy of the learned θ [16, 19], but also leads to longer run-times[1]. For example, when tested on our working example from Section 2.4.3, Double MH took 20% longer to complete the same number of MCMC iterations as sampling or maximization

[1]We provide our code, environments, and additional results in this repository. This includes an example of learning from state-action pairs.

baselines.

2.6 Simulations

Here we compare approaches for approximating the normalizer and performing Bayesian reward learning in controlled environments. We simulate noisily rational humans who provide multiple, conditionally independent demonstrations. The space of rewards Θ is continuous, and we attempt to infer the simulated human's $\theta \in \Theta$ based on their demonstrations.

Independent Variables We vary the robot's learning method across the algorithms introduced in Section 2.5. This includes naïve robots that **Ignore** the normalizer, robots that approximate the normalizer using **Sampling** or **Maximum**, and our proposed **Double MH** approach. We also vary the simulated human's rationality β at two levels: noisy humans ($\beta = 5$) and consistent humans ($\beta = 25$). These values of β were identified through a preliminary round of simulations: below $\beta = 5$ humans acted almost completely randomly, and above $\beta = 25$ the humans converged to always select the optimal ξ.

Dependent Variable Each human samples their true reward θ uniformly at random. We report the *Error* between the actual θ and $\hat{\theta}$, the mean of the robot's estimate: *Error* $= \|\theta - \hat{\theta}\|$.

Environments We performed simulations across three dynamic physics-based environments where each environment had two features (see 2.4, 2.5, and 2.6). In *Push* the human's reward traded off between the distance the robot pushed the box and the length the robot travelled. In *Close*, the human's reward traded off between pushing the door closed (i.e., the angle of the door) and keeping the robot's end-effector close to the table

(i.e., the robot's height). Finally, in *Pour*, the robot needed to pour coffee at a specific position, and the features were the distance travelled and holding the cup upright. In each environment, the robot had an accurate predictive model of the world and could simulate the outcomes of each trajectory.

Procedure Each simulated human chose a θ vector uniformly at random. The human then generated $K = 3$ demonstrations of their desired motion so that $\mathcal{D} = \{\xi_1, \xi_2, \xi_3\}$. These demonstrations were sampled from our noisily rational model in Equation (2.2) and were conditionally independent (i.e., ξ_2 did not depend on ξ_1). The robot then observed $\mathcal{D}$ and used its Bayesian reward learning approach to get a mean estimate of θ. For each environment, we repeated this procedure across 100 simulated humans and reported the average results.

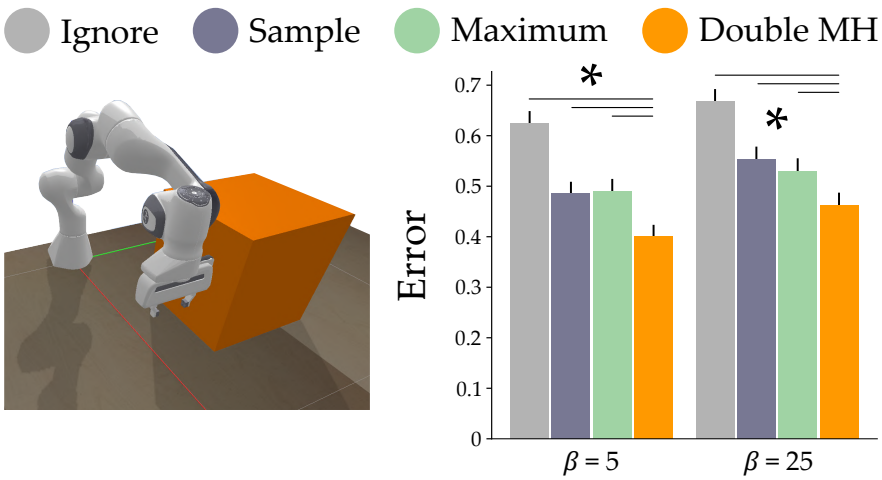

Figure 2.4: Results from the *Push* simulation. (Left) The reward depends on the distance the box is moved and the distance the end-effector travels. (Right) Error in the learned θ across 100 simulated humans. Error bars show standard error, and an $*$ denotes statistical significance ($p < .05$).

Results Our results for *Push*, *Close*, and *Pour* are shown in Figures 2.4, 2.5, and 2.6. To analyze these results we first performed separate repeated measures ANOVAs on each environment and found that the normalizer approximation had a statistically significant

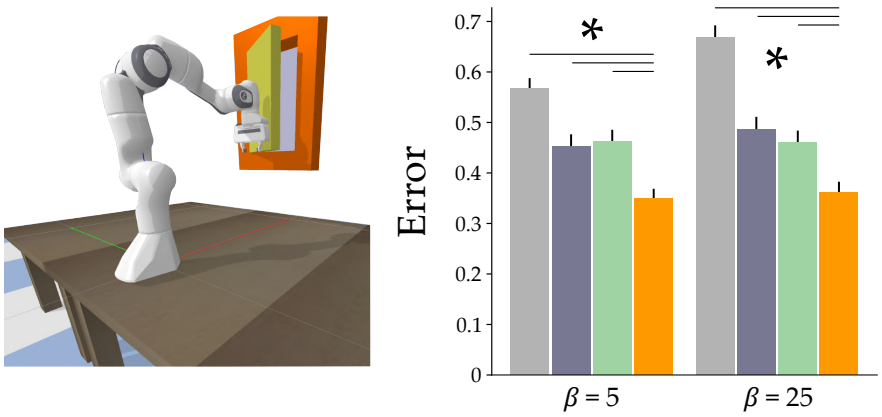

Figure 2.5: Results from the *Close* simulation. (Left) The reward depends on the angle of the door and the robot's height from the table.

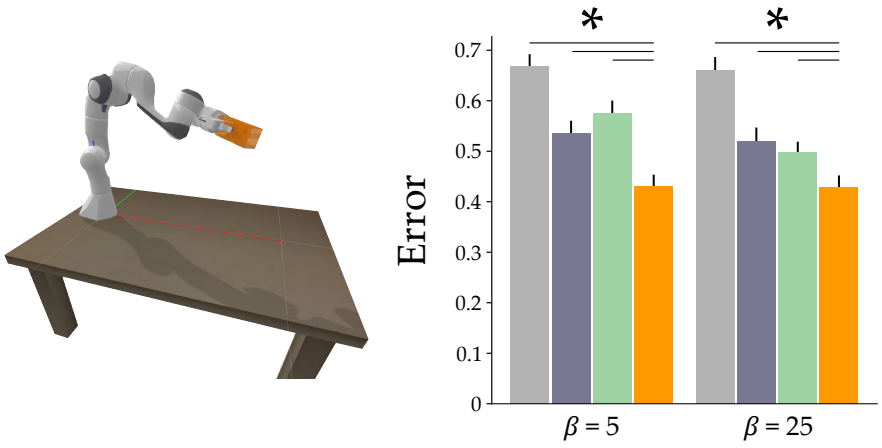

Figure 2.6: Results from the *Pour* simulation. (Left) The reward depends on the orientation of the cup and the length of the robot's trajectory in joint space.

effect. Post-hoc t-tests revealed that **Ignore** learned the *least accurate* estimate (i.e., had the most error) across the board. At the other end of the spectrum, **Double MH** resulted in the *most accurate* estimate for each environment and rationality β. Consider Figure 2.4 with $\beta = 5$ for instance: here t-tests show that **Double MH** has significantly lower error than **Ignore** ($t(99) = 7.8$, $p < .001$), **Sample** ($t(99) = 3.0$, $p < .05$), and **Maximum** ($t(99) = 3.7$, $p < .001$). Overall, our simulation results in these three physics-based tasks suggest that (a) ignoring the normalizer altogether leads to inaccurate inference, and (b) using Double MH sampling to approximate the normalizer outperforms existing approximation methods.

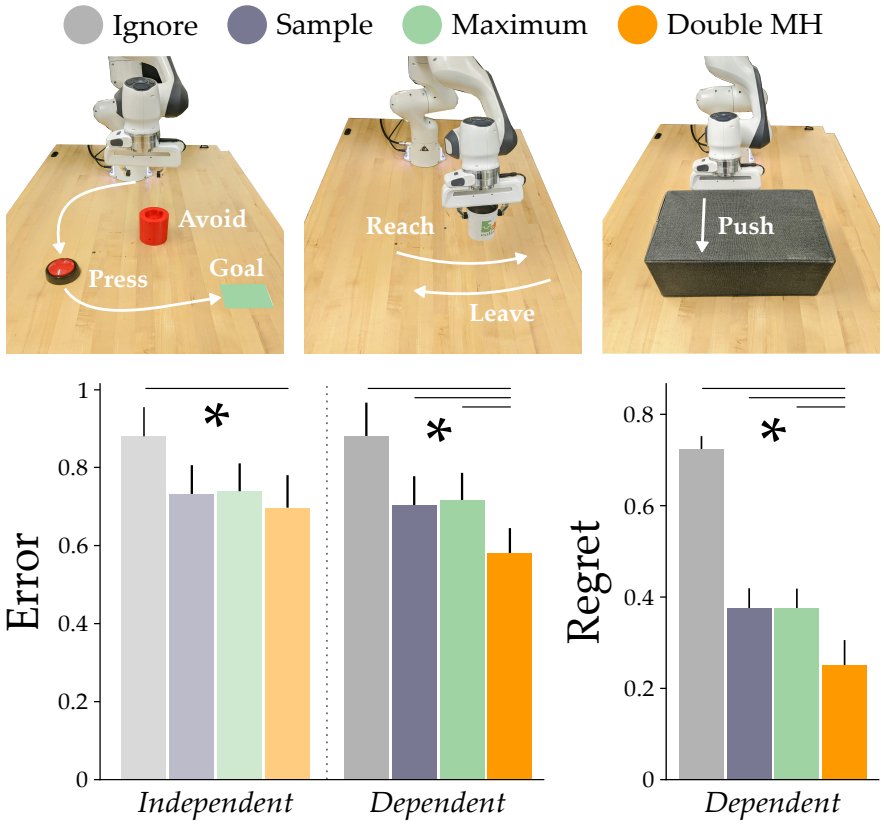

Figure 2.7: Results from our user study in Section 3.5. (Left) The *Press*, *Reach*, and *Push* tasks. In each task, the robot moved along an initial trajectory, and users physically corrected the robot to teach it their desired behavior. (Right) *Error* and *Regret* averaged across the 10 users and three tasks. Lower *Regret* indicates that the robot's learned trajectory was better aligned with the desired behavior. Error bars show standard error, and an $*$ denotes $p < .001$.

2.7 User Study

We compared our proposed approach to existing approximations when learning rewards from actual users. In each task, the robot started with an initial trajectory and users physically *corrected* the robot arm to better align its motion with their objective. To standardize these results, we first displayed the desired trajectory that the human should teach to the robot (i.e., we specified the user's reward parameters θ). The participant's corrections were then used to infer an estimate of θ, and we compared what the robot learned to the objective that the human was trying to teach the robot.

Independent Variables For this study, we varied the robot's learning along two factors: approximation type and conditional dependence. The robot used the **Ignore**, **Sample**, **Maximum**, and **Double MH** algorithms. We emphasize that **Ignore** [8, 9, 10], **Sample** [11, 12, 30], and **Maximum** [15, 32] come from prior work. We also compared *Conditionally Independent* and *Conditionally Dependent* versions of these algorithms. Recall from 2.5.1 that — when the robot treats the human's inputs as conditionally dependent — it recognizes that the human's current correction could build upon their prior corrections. Given that the human's corrections are sequential and interconnected, we anticipated that conditionally dependent learning would result in more accurate inference. To sample conditionally dependent corrections $\mathcal{D}'$ in Equation (2.13), we gave the robot an initial trajectory and then applied uniformly distributed perturbations to the waypoints along that trajectory.

Dependent Measures We recorded each participant's corrections and applied Bayesian reward learning to infer their objective θ. As in Section 2.6, we compared the *Error* between the θ given to users and the robot's learned estimate $\hat{\theta}$: $\textit{Error} = \|\theta - \hat{\theta}\|$. We also computed the *Regret* between the ideal trajectory ξ (i.e., the trajectory we showed to participants which optimizes for θ) and the learned trajectory $\hat{\xi}$ (i.e., the optimal trajectory under the robot's estimate $\hat{\theta}$).

$$\textit{Regret} = R(\xi, \theta) - R(\hat{\xi}, \theta), \quad \hat{\xi} = \arg\max_{\xi \in \Xi} R(\xi, \hat{\theta}) \tag{2.15}$$

Lower *Regret* means the robot has learned the correct behavior.

Experimental Setup Users taught the robot three tasks (see figure user study). One of these tasks (*Push*) was consistent with the Simulations, and we introduced two new tasks to test the generality of our approach. In *Press* the robot traded off between pressing a

button, reaching a goal, and avoiding an obstacle. In *Reach*, the robot tries to offer coffee to the user and then moves away after the coffee is delivered. Here *Press* had four features and *Push* and *Reach* each had three features. For each task, the human provided three separate sets of corrections for three different values of θ. Users first watched the robot demonstrate the ideal motion (i.e., the trajectory that optimized θ) then gave a sequence of corrections to convey that θ to the robot.

Participants A total of 10 participants from the Virginia Tech community took part in this study (2 female, ages 27 ± 6.4 years). Eight of the ten users had interacted with robots before, and the other two users had no prior experience in robotics. Users provided written consent under IRB#20-755.

Results Our results averaged across these 10 users and three tasks are summarized in Figure 2.7.

We first analyzed the effects of treating the human's corrections as conditionally independent or dependent. A repeated measures ANOVA revealed that conditionally dependent learning led to lower *Error* across the board: $F(1, 29) = 10.1$, $p < .001$. This result matched our intuition: it appeared that users often tried to fix something in their current correction based on what went wrong in the previous correction.

We next focused on the type of normalizer. Looking specifically at conditionally dependent learning, post-hoc analysis showed that **Double MH** resulted in lower *Error* and *Regret* as compared to each state-of-the-art alternative ($p < .001$). This suggests that — not only does **Double MH** lead to a more accurate estimate of the human's reward — but that estimate also results in robot trajectories that better match the human's desired behavior. See videos of our user study and the learned behaviors here: With these

2.8 Conclusions

Our work is a step toward robot learners that infer the human's objective from demonstrations and corrections. In this work, we explored the doubly-intractable nature of Bayesian reward learning, where the robot must reason over all possible trajectories and rewards. We grouped existing robotic approximations into three classes and theoretically derived their relative strengths and weaknesses. We then introduced a new Monte Carlo approximation method from the statistics community. Overall, our simulations and user studies suggest that this Double MH approach more accurately infers the human's objective, and is versatile enough to learn from independent demonstrations or interconnected corrections.

Continuing upon this work with Bayesian Inference, we move on to our research regarding Shared Autonomy for inferring the human's objective. For this, we pull a similar application of Bayesian Inference for the human's objective, in this case, which target in an environment would the human like assistance in interacting with. However, instead of the change within the normalizing function as we examined in this chapter. We turn our focus to how the use of information within a communicative framework of Shared Autonomy can be used in improving the inference over the human's objective.

Chapter 3

Aligning Learning with Communication in Shared Autonomy

3.1 Introduction

More than 24 million American adults need external assistance when performing activities of daily living [39]. Assistive robot arms that *share autonomy* with humans have the potential to help address this challenge [2, 40]. In these shared autonomy settings the human controls the robot arm using an input device (e.g., a joystick) to indicate their intent, and the robot helps automate tasks on the human's behalf (e.g., picking up foods and feeding them to the operator).

To achieve seamless assistance, both the human operator and robot arm must be on the same page. Consider Figure 3.1, where a human is using a robot arm to manipulate kitchen items. The human wants the robot to pick up a fork, and so the human provides joystick inputs that guide the robot towards that goal. *For the robot to align with the human*, the robot must *learn* from these inputs to determine the human's intent and partially automate their task. Here the robot might correctly infer what the human wants (e.g., a fork) and then coordinate its own motions to help reach that goal (e.g., fixing any errors in the human's inputs to precisely pick up the fork). On the other hand — *for the human to align with the robot* — the robot needs to *communicate* its intended assistance

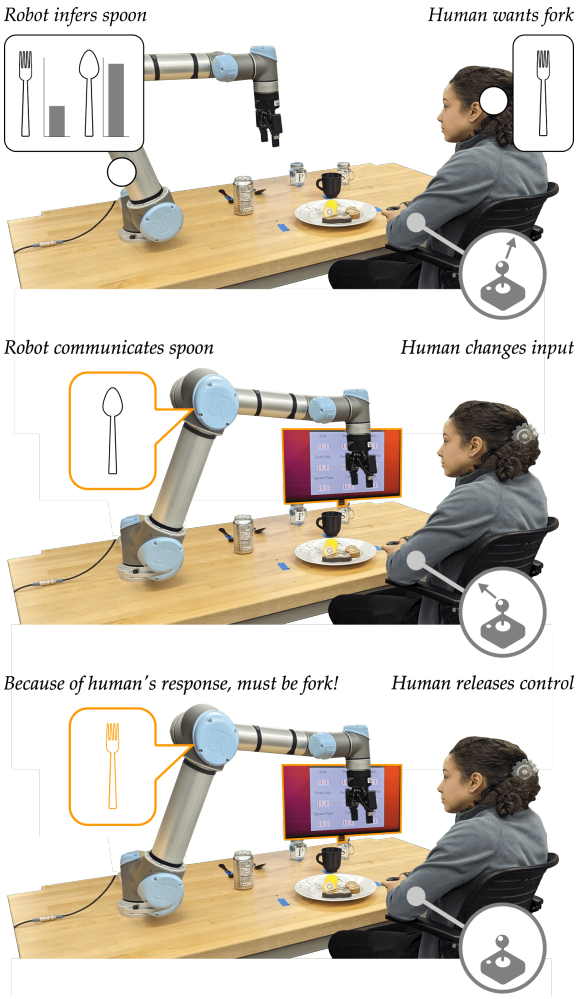

Figure 3.1: Human sharing control with an assistive robot arm. (Top) The robot tries to infer the correct task from the human's joystick inputs. (Middle) We show that — when the robot communicates what it has inferred — the way humans provide inputs *changes*. (Bottom) If robots are aware of these changes, they can more accurately infer the human's goal.

back to the user. Without this communication the human does not know what to expect from the robot: is the robot going to help automate the motion to the fork, or does the robot think the human wants something else entirely?

Existing research on shared autonomy has largely separated learning and communication. On the one hand, methods such as [3, 21, 36, 41, 42] focus on inferring the human's task and partially automating the robot's motion, but do not consider communication back to the human. On the other hand, approaches like [24, 27, 43] develop visual and haptic

communication interfaces for shared autonomy, but do not modify the robot's learning algorithm. In this work, we explore the intersection of learning and communication within shared autonomy settings. More specifically, we hypothesize that:

Humans will interact with shared autonomy systems differently *when those systems communicate their learning.*

This is important because — if humans do provide different inputs in the presence of communication — then the way the robot interprets and learns from human actions should also be modified. Accordingly, our work has two main parts. First, in Section 3.3.2 we test our hypothesis and measure how communication can affect the way humans interact with assistive robot arms. Second, in Sections 3.4 and 3.5 we harness the changes caused by communication to modify the robot's learning algorithm. In practice, this combination of learning and communication enables a) the robot to more seamlessly infer the human's task, and b) the human to more clearly indicate their intent. Returning to Figure 2.1, perhaps the human stops providing inputs because they observe from the robot's feedback that the fork is the robot's most likely goal. In response, our robot is able to confirm its prediction (i.e., because the human released control the robot must be correct), and complete the task more efficiently.

Overall, we make the following contributions:

Measuring the Effects of Communication. We consider shared autonomy settings where a human is operating a robot arm, and the robot updates the likelihood of each potential task based on the human's inputs. For these settings, we perform online and in-person user studies with and without robot communication. We find evidence that humans behave differently in the presence of communication.

Updating the Robot's Learning Rule. Our experimental results suggest that — when

communication is present — humans are more likely to intervene if the robot has inferred the wrong task, and more likely to relinquish control if the robot is correct. We use these findings to modify the human model of an existing shared autonomy algorithm.

Combining Learning and Communication. We conduct another in-person user study with three conditions: learning (where the robot does not provide explicit feedback), communication (where the robot communicates its intent but does not adjust its learning rule), and our proposed approach. Our results suggest that the combination of learning and communication increases subjective and objective performance in shared autonomy settings.

3.2 Related Works

Below we discuss shared autonomy research that focuses on either learning (i.e., inferring the task and providing assistance) or communication (i.e., visual and haptic interfaces to convey the robot's internal state).

Learning in Shared Autonomy. Shared autonomy is a collaborative framework for human-robot interaction where the robot's behavior is a blend of the human's inputs and the robot's autonomous assistance [44]. The human's inputs convey the high-level task (e.g., grasping a fork), and the robot's inputs provide fine-grained corrections (e.g., coordinating the motion of the arm to reach that fork). Prior works develop algorithms to learn both the high-level task and low-level assistance. For example, in [3, 21, 36, 41, 42, 45] the human's desired task is to reach a goal from a discrete set of options, and the robot infers this goal based on the human's inputs. As the robot becomes more confident in which goal the human wants, it can increasingly provide assistance to automate that task. Similarly, in [46, 47, 48, 49] the robot builds an estimate of the task's

reward function, and overrides any accidental or incorrect human inputs that would result in poor performance (e.g., preventing the human from moving the robot arm into a collision). Other methods such as [6, 22, 50, 51, 52] learn to assist the human by imitating their previous behaviors. For instance, if the human showed the robot how to pick up a fork in a past interaction, the robot leverages that data to help pick up forks during future interactions. Overall, each of these works provides a way for the robot to learn from and assist the human. However, they do not explicitly communicate what the robot has learned — hence, the user may not know what to expect from the autonomous agent.

Communication in Shared Autonomy. Research outside of shared autonomy contexts suggests that communicating robot learning has benefits for both the human and the robot. From the human's perspective, communication increases the user's acceptance and trust in the system [23]; from the robot's perspective, communication can result in more effective human teaching and accelerated robot learning [26]. Accordingly, recent works have started to apply communication strategies to shared autonomy [53]. In some scenarios, it is possible for the robot to *implicitly* convey what it has learned by exaggerating its motions [25]. However, for the robot to clearly indicate its latent state in everyday settings, *explicit* communication with visual, auditory, or haptic interfaces is often necessary. In [27] and [43] augmented reality headsets show the operator what the robot has learned about their high-level task (e.g., placing visual markers at the most likely goals) and how the robot plans to assist (e.g., displaying the robot's planned trajectory). Similarly, in [24] a wearable haptic interface notifies the human when the shared autonomy system is uncertain about their intent. Our work will build upon these related works by using explicit communication to convey the robot's inferred task back to the human. However, instead of focusing on the communication interface itself, we are interested in the effects of this communication on the human operator and assistive agent.

3.3 Effects of Communication in Shared Autonomy

We consider shared autonomy settings where the human and robot collaborate in achieving a common goal. A key aspect of shared autonomy is the ability of the robot to infer the human's goal (i.e., the task they are trying to complete). If the robot correctly infers the human's goal, it can complete the remaining task without requiring further human input. Alternatively, if the robot's inference is incorrect, the human must keep providing inputs towards their intended goal. However, it can be challenging for humans to determine what goal the robot has inferred without explicit communication.

In this section, we investigate how explicitly communicating the robot's belief about the human's goal affects their actions. We first introduce the policies of the human and the robot collaborator in the absence of communication. Then, we conduct a user study to understand the role of communication in shared-autonomy settings and determine how the users' actions change when communication is introduced. We aim to use these findings to improve the robot's inference of the human's goal and provide better assistance.

3.3.1 Shared Autonomy without Communication

We let $s \in \mathcal{S}$ be the environment state which includes the state of the robot, $a_{\mathcal{H}} \in \mathcal{A}$ and $a_{\mathcal{R}} \in \mathcal{A}$ be the human's and robot's actions respectively. The environment state transitions based on both the human and robot actions.

$$s^{t+1} = f\left(s^t, a_{\mathcal{H}}, a_{\mathcal{R}}\right) \tag{3.1}$$

We assume that the human chooses actions to minimize an internal cost-value function $Q^\star$:

$$a_{\mathcal{H}} \sim \pi^\star_{\mathcal{H}} \left(\circ \mid s, \theta, Q^\star \right) \tag{3.2}$$

Correspondingly, as the robot is trying to achieve the same goal as the human, it should take actions that minimize the human's cost-value function $Q^\star$. The robot does not directly observe the human's goal or their cost-value function. Instead, the robot selects actions according to an approximation Q of the cost-value function from prior work [21] where the robot's belief is not directly communicated to the human:

$$a_{\mathcal{R}} \sim \pi^\star_{\mathcal{R}} \left(\circ \mid s, b\left(\theta\right), Q \right) \tag{3.3}$$

where $b(\theta)$ is the robot's *belief* of the human's goal θ.

We suspect that the human's actions will change in the presence of communication. If the belief communicated by the robot aligns with the human's goal — will the human continue to provide actions that navigate the robot towards their goal or will they allow the robot to assume full control? On the other hand, if the robot's belief is incorrect — will the human *exaggerate* their corrective actions because they know that the robot's belief is incorrect? To evaluate how real users respond to robots that communicate their belief, we conducted two user studies in the absence and presence of communication.

3.3.2 Shared Autonomy with Communication

We performed online and in-person user studies to gain insight into the effect of communication on shared autonomy. Participants collaborated with a robot to reach a goal while choosing how much input they think is enough for the robot to learn the task. In half

of the interactions, the robot communicated its current belief of the user goal as a percentage using a digital interface. Our results from 25 online users and 10 in-person users show that people provide less input when the robot communicates its belief over the user goal. Additionally, the subjective polled results from the in-person study show a significant preference for a system that communicates the robot's intention for its cooperation.

Experimental Setup. In the online study, participants taught a robot to reach a goal in multiple shared autonomy settings. In each setting, there were three objects with varying colors and the user's goal was to reach the green square (see Figure 3.2 (Left)). The position of these objects varied between settings. To simulate the settings we used an animated 2D environment with a top-down view. Online participants first watched the beginning of the robot arm's motion and then selected their choice of input to guide the robot toward the desired goal or to allow the robot to continue on its partially demonstrated path. In the in-person study, users commanded a robot arm using a Logitech F710 gamepad to perform a similar task of reaching a green cube.

We had five different settings for the objects in the online study and three settings in the in-person study. All participants interacted with the robot in each setting twice — with and without communication. In total, participants had six interactions in the in-person study and ten interactions in the online study. Each interaction ended when the robot reached the correct goal. The order of the settings was randomly counterbalanced across all users.

Independent Variables. For the online study, the users interacted with the robot in each setting across two variations. In one variation, the users had to infer the robot's intended goal through its animated motion (**Without Interface**). In the other variation, users were provided with the probabilities of the robot's belief over the goals (**With Interface**) in addition to their observation of the robot's motion.

For the in-person study, the robot used a state-of-the-art shared autonomy algorithm [21] to select its action $a_{\mathcal{R}}$ in each setting. In half of the interactions, the robot communicated its current belief, $b(\theta)$, as percentages using a digital interface (**With Interface**) and for the other half, the users had to infer the robot's belief from its motion (**Without Interface**).

Dependent Variables. In both studies, we focused on how the user responses change when performing the shared autonomy tasks with and without communication. For the online study, we recorded whether the human chose to command the robot toward the desired goal or not. For the in-person study, we recorded the time that users spent using the gamepad and other joystick inputs (*Total Human Inputs*) as well as their subjective responses on a 7-point Likert scale for whether they preferred the settings with explicit communication or without the interface.

Participants. For the online study, we recruited 25 anonymous participants. We included an instruction and a qualifying question at the beginning of the survey for this study. For the in-person study, we recruited 10 participants from the Virginia Tech community (2 female, ages 23 ± 9 years). All participants provided informed consent as per university guidelines (IRB #20-755). To assist the participants in becoming familiar with the gamepad and the robot we provided practice time at the beginning of the interaction.

Hypothesis. We hypothesized that:

> **H1.** *When the robot communicates its belief over the goal, users will require less effort in commanding the robot to reach the desired goal.*
>
> **H2.** *Users will prefer using a shared autonomy system where the robot's belief is communicated.*

Results. Our results from the online and in-person user studies are summarized in Figure

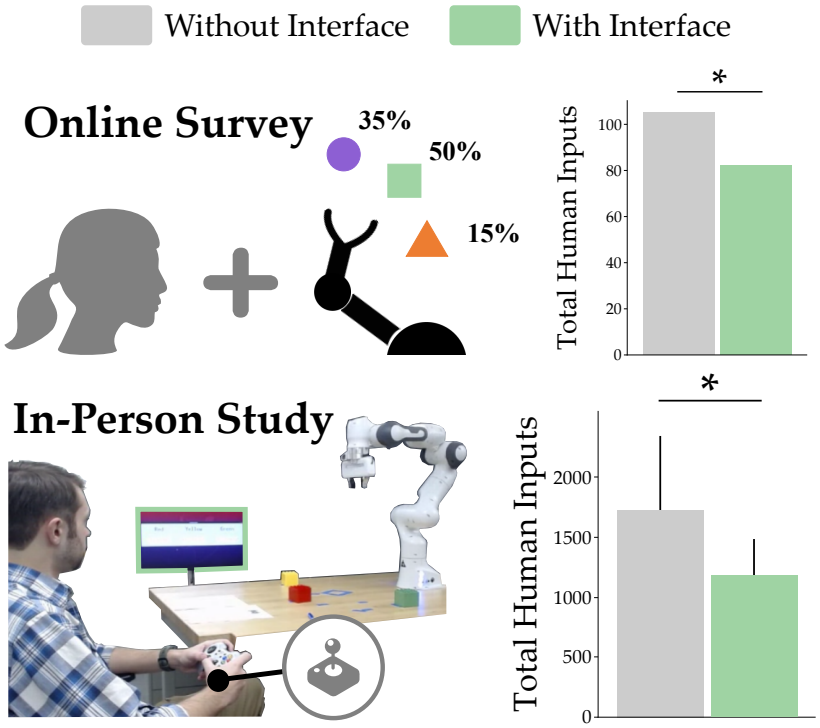

Figure 3.2: Example settings and results from our user studies in Section 3.3.2. Here we explored how communicating the robot's inferred distribution over a discrete set of tasks affected the human's inputs during shared autonomy. In all conditions, the robot used the same learning algorithm. (Left) Results from the online survey with and without a communication interface. Humans were more likely to release control to an assistive robot that conveyed its learned distribution over the tasks ($t(24) = 4.271, p < 0.005$). (Right) Corresponding results from our in-person study. Here humans required fewer inputs to guide the robot to their goal when the robot communicated its learning ($t(29) = 2.986, p < 0.005$). Overall, these results suggest that humans are more willing to yield control to a communicative system. An asterisk (*) denotes statistical significance.

3.2. To address **H1**, we evaluate the level of *effort* that users exhibited through the number of human inputs given through the gamepad. Here, there was a significant difference ($t(24) = 4.271, p < 0.005$) in the requisite human effort to reach the given goals. This result shows that when the robot communicates its belief, the user no longer has to provide the same level of effort for the robot to reach the goal, supporting **H1**.

For **H2**, we turn to our Likert-scale survey. We performed a Paired-Samples T-Test across polled user preferences for communication; these results were significant

$(t(9) = 17.676, p < 0.001)$. In our in-person user study, participants preferred interacting with a robot that communicated its belief of the human's intent.

3.4 Harnessing Communication to Improve Learning

Our results from the first user study (Sec. 3.3.2) demonstrate that humans behave differently in settings with communication than those without it. In this section, we leverage the human's response to the robot's communication in a novel shared-autonomy formalism. Instead of solely using communication to aid the human's guidance of the robot, we treat the human's feedback to the communication as an indication of the user's confidence in the robot.

We use this idea to present model human policies for both modalities: in the presence and absence of communication. The robot policy uses the appropriate human model to choose assistive actions that minimize the human's modality-specific cost-value function.

Human. The human takes actions that minimize their internal cost-value function Q. Following previous works [3, 25], we model the human as a nosily rational agent according to the Boltzmann distribution:

$$\pi_{\mathcal{H}}\left(a_{\mathcal{H}} \mid s, \theta\right) = \frac{\exp\left(\beta \cdot Q\left(s, a_{\mathcal{H}}, \theta\right)\right)}{\int \exp\left(\beta \cdot Q\left(s, a'_{\mathcal{H}}, \theta, \right)\right) da'_{\mathcal{H}}} \tag{3.4}$$

Here, $\pi_{\mathcal{H}}$ is a model of the human's true policy $\pi^{\star}_{\mathcal{H}}$. In the Boltzmann rational distribution, $\beta \in [0, \infty)$ is the rationality hyperparameter: as β approaches 0, the human is considered to be more irrational; their actions are essentially uniformly distributed. On the other hand, as β increases, the human's actions are increasingly optimal (i.e. "rational"). The robot does not have access to the human's policy; instead, it assumes an apriori model of

the human. In continuous spaces, Equation 3.4 is intractable. Similar to [3], we tractably estimate the human's policy using the principle of maximum entropy: the probability of a goal decreases exponentially as its cost increases. This yields the following approximation:

$$\pi_{\mathcal{H}}\left(a_{\mathcal{H}} \mid s, \theta\right) \propto \exp\left(-\beta \cdot \mathcal{Q}(s, a_{\mathcal{H}}, \theta)\right) \tag{3.5}$$

Firstly, in the absence of communication, we approximate the human's cost-value function as:

$$\mathcal{Q}(s, a_{\mathcal{H}}, \theta) = \text{dist}(a_{\mathcal{H}} + s, \theta) - \text{dist}(s, \theta) + \|a_{\mathcal{H}}\| \tag{3.6}$$

The first two terms measure the distance by which the human actions move the robot away from the human's goal, while the last term measures the magnitude of the human actions. Formally, Equation 3.6 is minimized when the human takes *low-effort* actions that minimize the distance between the robot and their goal and require the least effort to do so. However, in the absence of communication, humans cannot directly observe whether or not their actions have influenced the robot's belief to a state where they no longer need to provide input actions and thus, have no reliable basis on which to determine when they can minimize their effort.

In the absence of communication, the human must infer the robot's belief by observing the robot's actions. However, in many cases there can be uncertainty in determining the robot's goal — for example, if the spoon and the fork are close to one another, how can the human reliably tell which goal the robot is moving towards? On the other hand, in the presence of communication, the human has a reliable prediction of the robot's future assistive actions given its belief and can respond to this communication *positively* by removing input or *negatively* by continuing to work against the robot. Our key insight is that when the robot's belief is communicated, human inputs can be interpreted as

assurance or rebuttal of this communicated belief.

Therefore, we propose that the human's internal cost-value function in the presence of communication can be modelled by incorporating the robot's belief into the cost of the human's actions.

$$\mathcal{Q}(s, a_{\mathcal{H}}, \theta) = \text{dist}(a_{\mathcal{H}} + s, \theta) - \text{dist}(s, \theta) + b(\theta) \cdot \|a_{\mathcal{H}}\| \tag{3.7}$$

In the presence of communication, if the robot's belief is correct, then the human's cost is minimized by providing little effort in agreement with the robot's assistance. If the robot's belief is *incorrect*, then the human will provide inputs that contradict this belief. For example, in the case of ambiguous goals (i.e., the spoon and fork placed close together), with the presence of communication, the human will hold a definite answer for whether the robot is correct. This will result in either further adjustments to correct a misaligned belief or a submission of control seeing that they can minimize their effort by relying on the robot's assistance.

Robot. The robot updates its belief $b(\theta)$ based on the observed human actions. Let $P(\theta \mid s, a_{\mathcal{H}})$ denote the probability that the human is optimizing for the goal θ given the state s and human action $a_{\mathcal{H}}$. Using Bayes' theorem, the posterior probability is defined as:

$$P(\theta \mid s, a_{\mathcal{H}}) \propto P(a_{\mathcal{H}} \mid s, \theta) \cdot P(\theta) \tag{3.8}$$

Here, $P(\theta)$ is the prior of the robot's belief over the human's goal and $P(a_{\mathcal{H}} \mid s, \theta)$ is the likelihood function for the robot's prediction. Note that $P(a_{\mathcal{H}} \mid s, \theta)$ is equivalent to $\pi^{\star}_{\mathcal{H}}$, which we model as $\pi_{\mathcal{H}}$. Similar to Equation 3.5, we use the principle of maximum entropy

to derive an equivalent form for Equation 3.8:

$$P(\theta \mid s, a_{\mathcal{H}}) \propto \exp\left(-\beta \cdot Q\left(s, a_{\mathcal{H}}, \theta\right)\right) \cdot P(\theta) \tag{3.9}$$

The robot takes actions $a_{\mathcal{R}}$ that minimize Equation 3.6 in the absence of communication and Equation 3.7 in the presence of communication according to:

$$a_{\mathcal{R}} = \sum_{\theta \in \Theta} P\left(\theta \mid s, a_{\mathcal{H}}\right) \cdot (\theta - s) \tag{3.10}$$

Since the robot's belief may be incorrect, the robot *blends* the human's commanded action with an assistive action:

$$a_{\mathcal{B}} = (1 - \alpha) \cdot a_{\mathcal{H}} + \alpha \cdot a_{\mathcal{R}} \tag{3.11}$$

The hyperparameter $\alpha \in [0, 1]$ is determined by a threshold according to the human's action such that when the robot displays the correct belief and the human surrenders control, the robot is allowed to take a higher level of control to assist. For this, we transition alpha from a minimum value in the presence of human action to a maximum value when the robot is in full control.

$$\begin{cases} \alpha = \alpha + \mathit{step}, \ \alpha \leq \alpha_{\max} & \text{if } \|a_{\mathcal{H}}\| \approx 0 \\ \alpha = \alpha - \mathit{step}, \ \alpha \geq \alpha_{\min} & \text{if } \|a_{\mathcal{H}}\| \not\approx 0 \end{cases} \tag{3.12}$$

Here *step* is a hyperparameter chosen by the designer to control the rate at which the robot will increase its assistance proportionally to the number of timesteps that the user has allowed for complete robot assistance.

Altogether, Equations 3.7-3.12 form our method for selecting optimal actions in the presence of communication. These equations build upon existing shared autonomy

approaches for inferring the human's goal and providing assistance [3, 21, 36]. But we have modified this existing learning framework to explicitly account for communication and the effect communication may have on the human's internal cost function Q^*. Without communication, earlier works such as [21] suggest that the human optimizes for their error and effort as shown in Equation 3.6. However, with communication, our experimental results from Section 3.3.2 indicate that humans are willing to increase their effort if the robot is wrong and release control when the robot is correct. Using these findings we update our human model for Q in Equation 3.7. Up to this point, our modified learning rule is informed by experiments but has not yet been tested. Accordingly, in Section 3.5 we will compare our proposed method for aligning learning with communication against baselines that separately learn and communicate.

3.5 Testing the Combination of Learning and Communication

Lastly, we conduct an in-person user study to evaluate the performance of our proposed method in comparison to the state-of-the-art shared autonomy baseline [21] with and without a communicative interface. We wish to demonstrate that accounting for the knowledge of the robot's belief in the human's cost function, in addition to communicating the robot's belief, allows the robot to provide better assistance than simply communicating the robot's belief with the baseline shared autonomy approach.

Experimental Setup. Users were instructed to complete three tasks in a more complicated environment than the first in-person user study to highlight the utility of this approach:

1. **Seasoning:** Retrieve a salt or pepper shaker, bring it to a plate of food, and then return it to its base position.

2. **Drink:** Go to the can of soda, bring the can of soda to a mug, and return the can to its base position.

3. **Utensil:** Retrieve the spoon or fork and bring it to the relevant side of the plate.

Participants commanded the robot using a Logitech F310 gamepad to complete each of the three tasks using one of three methods: **Without Interface**, **With Interface**, and **Ours**. The order in which participants interacted with these methods was randomized to avoid any proficiency bias. Details of these tasks are illustrated in Figure 3.3 (Left).

Independent Variables. In each task, the robot starts with a uniform prior over the goals which is gradually updated according to the methods discussed in section 3.4. Participants performed each task three times — using the baseline shared autonomy approach **Without Interface**, using the same baseline **With Interface**, and using our method of feedback-enabled shared autonomy - **Ours** (which combines learning with communication).

Dependent Variables. We recorded the *Total User Inputs* to measure the amount of effort spent by the users in completing each task. We also recorded subjective *User Scores* through a 7-point Likert scale survey with four items — for how easy it was to *Control* the robot, how often they could tell when the robot *Assisted* them, whether the robot was able to *Predict* their goals, and if the robot *Adapted* to their actions.

Participants. A total of 12 participants from the Virginia Tech community took part in this study (2 female, ages 28.5 ± 6.5 years). Two of the twelve users had not interacted with robots before. Users provided written consent as per university guidelines (IRB #20-755).

Hypothesis. We hypothesized that for this study:

H3. *The human will spend less effort in completing the tasks when using* Our *method.*

H4. *Users will provide higher scores on the subjective metrics for* Our *method than the baselines.*

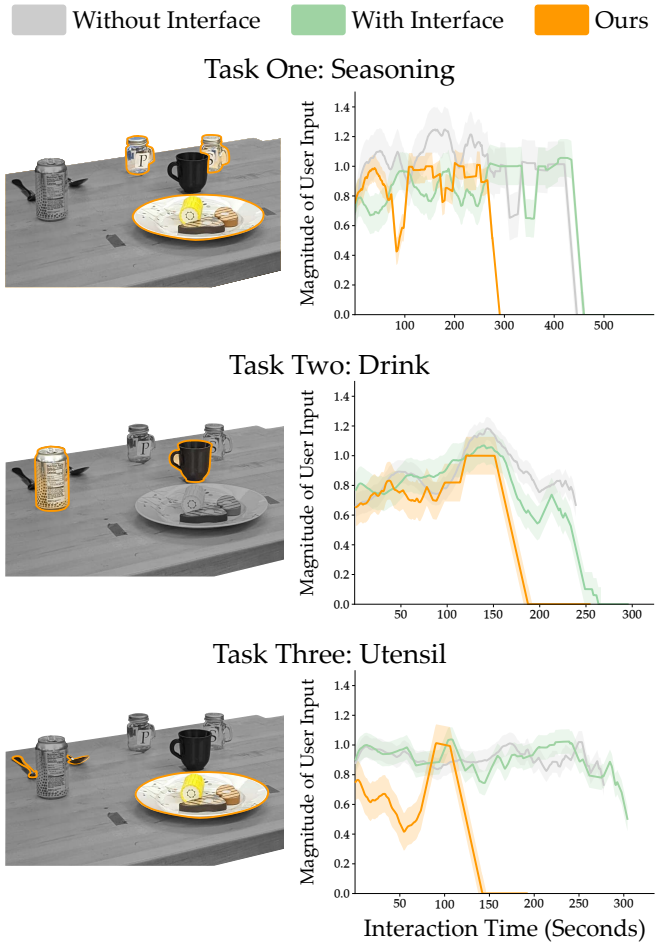

Figure 3.3: Tasks and user inputs from the user study in Section 3.5. (Left) The items the human led the robot to interact within each task. (Right) The magnitude of the human's inputs over time averaged across all users. These results show that users completed the tasks more quickly with **Ours**, and overall needed fewer inputs to convey their intended goals to the robot.

Results. The results of our user study are summarized in Figure 3.4. To address **H3**, we measured the number of user inputs across three separate tasks for each method. Here, a lower score is better: fewer inputs imply that the user is exhibiting less effort when completing the task. Paired-sample T-tests showed that participants used significantly fewer inputs when the robot used **Ours** for each task ($t(11) = 4.106, p < 0.001$, $t(11) = 5.806, p < 0.001$, $t(11) = 9.636, p < 0.001$). Figure 3.3 shows the average magnitude of the user input over time for each task; these results further support **H3**.

Regarding **H4**, we present the subjective results from our Likert-scale survey in Figure 3.4 (right). A one-way ANOVA analysis of the users' responses showed a significant difference in the perceived *Control*, *Assistance*, *Prediction*, and *Adaptation* that the robot exhibited when using our method ($F(69) = 11.901, p < 0.001$, $F(69) = 6.368, p < 0.005$, $F(69) = 8.794, p < 0.001$, $F(69) = 13.345, p < 0.001$). Actions chosen by **Ours** were preferable to those selected by baselines; this supports **H4**.

3.6 Conclusion

In this work, we explored the effects of communicating learned assistance back to the human operator in shared autonomy. While previous research has focused on learning the human's task and providing assistance, we instead focused on harnessing the effect of the communication. We hypothesized that humans will interact with shared autonomy systems differently when those systems communicate their learning back to the human. Using the results from online and in-person user studies, we showed that humans are more likely to intervene when the robot incorrectly predicts their intent, and release control when the robot correctly understands their task. We used the insights from these results to modify the robot's learning algorithm: under our proposed approach, the robot adjusts its model

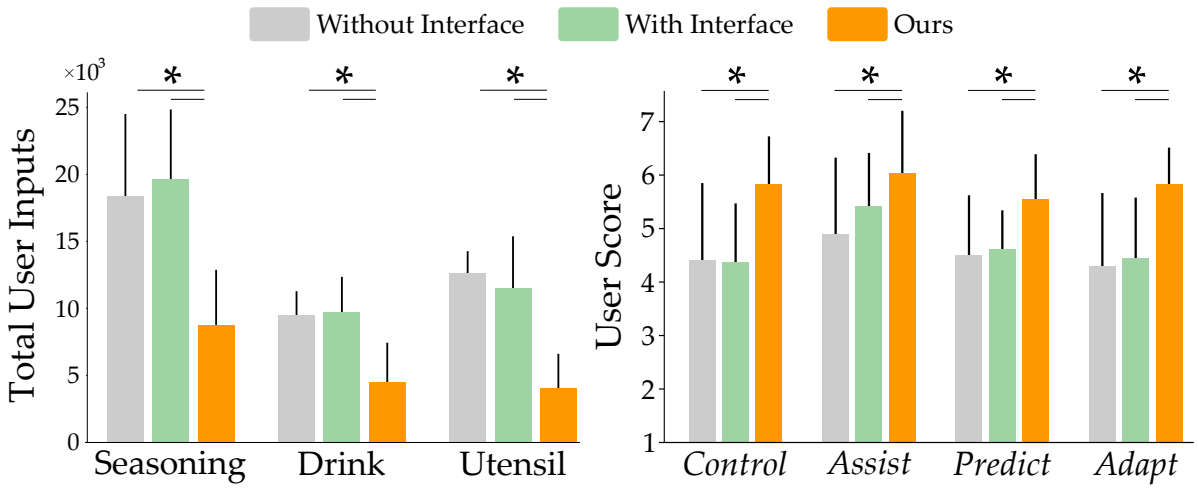

Figure 3.4: Objective and subjective results from the user study in Section 3.5. (Left) Total user inputs for **Seasoning**, **Drink**, and **Utensil** tasks. To count the number of inputs, the robot measured whether the human had pressed the joystick every 0.02 seconds. Across each task, users provided fewer inputs and relied on the robot's assistance more when using **Ours** ($p < 0.001$, $p < 0.001$, $p < 0.001$). These results support **H3**: Users spent less effort when using **Ours**. (Right) Subjective results for the three baselines. Across the four Likert-Scale items, users preferred **Our** method: they felt that they could easily *control* the system ($p < 0.001$), the robot provided effective *assistance* ($p < 0.005$), the robot better *predicted* their goal ($p < 0.001$), and the robot *adapted* more quickly to their actions ($p < 0.001$).

of the human's cost function to account for how communication changes the human's input patterns. Finally, we compared our approach for combining learning and communication against shared autonomy baselines that separately handle learning or communication. In a user study with 12 in-person participants across three kitchen tasks, we found that our proposed approach for combining learning and communication increased the subjective and objective performance of the human-robot team.

Chapter 4

Conclusion

This thesis has presented two contributions to the task of inferring the human's intent for human-robot interaction, which summarizes my work presented in [20, 28]. In Chapter 2, I work to provide experimental and theoretical judgment on the use of modern normalizing functions for Bayesian Inference in an intractable environment, then move to the use of alternative methods from the statistics community to infer upon the human's reward for this intractable environment. In Chapter 3, I discuss and work upon the dynamics of communication for Shared Autonomy and how the change in these dynamics with the addition of communicative elements can be used to better infer upon the human's objective given changes in the qualitative performance with such communication. Both of these works fall within the larger discussions of Bayesian Inference and Shared Autonomy and demonstrate how further improvements in the way we infer upon the human's intent in these areas can improve the relevant performance of inferring upon said human intent.

For future work, we hope to further expand on both topics covered within this thesis. For our work within inferring the human's reward function for task replication, we hope to further explore alternatives in sampling and normalization to better optimize this process of Bayesian Inference for alternative settings. Furthermore, although our method was significantly more accurate in the replication of the human's task objective, it was mentioned to be slower than the sampling and naive methods; so in future work, it is worth exploring how alternative sampling methods other than MCMC could be used to further

optimize the loop time for this reward learning to better expand upon these methods. Similarly, within our work for aligning communication in Shared Autonomy, the method we applied was one made with basic and widely applicable extension, but there is a lot of room to expand in detail. In our work, we note that there is a change in the modality of how the human interacts with the environment with and without communication and design an algorithm to take advantage of that change, however, we didn't extensively explore the more precise areas and transition of this change. For instance, one notable way we could work to better take advantage of this would be to train a model on recognizing more implicit features of the human avoiding incorrect signals, or amplifying correct signals as visualized through the communicative link. Given how a human with this interface HAS notably different behavior in guiding the robot, it would be quite interesting to see how the use of either AI modeling or more extensive algorithmic methods for inferring the human's reaction could be expanded to more precisely monitor this behavior to take advantage of the human's now explicit knowledge over the robot's belief.

Bibliography

[1] H. J. Jeon, D. P. Losey, and D. Sadigh, "Shared autonomy with learned latent actions," *arXiv preprint arXiv:2005.03210*, 2020.

[2] B. D. Argall, "Autonomy in rehabilitation robotics: An intersection," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 441–463, 2018.

[3] S. Jain and B. Argall, "Probabilistic human intent recognition for shared autonomy in assistive robotics," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 9, no. 1, pp. 1–23, 2019.

[4] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *AAAI*, 2008.

[5] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," in *IJCAI*, vol. 7, 2007, pp. 2586–2591.

[6] D. P. Losey, H. J. Jeon, M. Li, K. Srinivasan, A. Mandlekar, A. Garg, J. Bohg, and D. Sadigh, "Learning latent actions to control assistive robots," *Autonomous robots*, vol. 46, no. 1, pp. 115–147, 2022.

[7] H. J. Jeon, S. Milli, and A. Dragan, "Reward-rational (implicit) choice: A unifying formalism for reward learning," *NeurIPS*, 2020.

[8] Y. Cui and S. Niekum, "Active reward learning from critiques," in *IEEE International Conference on Robotics and Automation*, 2018.

[9] D. S. Brown, Y. Cui, and S. Niekum, "Risk-aware active inverse reinforcement learning," in *Conference on Robot Learning*, 2018.

[10] E. Bıyık, D. P. Losey, M. Palan, N. C. Landolfi, G. Shevchuk, and D. Sadigh, "Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences," *The International Journal of Robotics Research*, vol. 41, pp. 45–67, 2022.

[11] A. Bobu, A. Bajcsy, J. F. Fisac, S. Deglurkar, and A. D. Dragan, "Quantifying hypothesis space misspecification in learning from human–robot demonstrations and physical corrections," *IEEE Transactions on Robotics*, vol. 36, no. 3, pp. 835–854, 2020.

[12] A. Jonnavittula and D. P. Losey, "I know what you meant: Learning human objectives by (under) estimating their choice set," in *IEEE International Conference on Robotics and Automation*, 2021.

[13] C. Finn, S. Levine, and P. Abbeel, "Guided cost learning: Deep inverse optimal control via policy optimization," in *ICML*, 2016.

[14] A. Boularias, J. Kober, and J. Peters, "Relative entropy inverse reinforcement learning," in *AISTATS*, 2011, pp. 182–189.

[15] M. Li, A. Canberk, D. P. Losey, and D. Sadigh, "Learning human objectives from sequences of physical corrections," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 2877–2883.

[16] F. Liang, "A double Metropolis–Hastings sampler for spatial models with intractable normalizing constants," *Journal of Statistical Computation and Simulation*, vol. 80, no. 9, pp. 1007–1022, 2010.

[17] A.-M. Lyne, M. Girolami, Y. Atchadé, H. Strathmann, and D. Simpson, "On Russian

roulette estimates for Bayesian inference with doubly-intractable likelihoods," *Statistical Science*, vol. 30, pp. 443–467, 2015.

[18] P. Alquier, N. Friel, R. Everitt, and A. Boland, "Noisy Monte Carlo: Convergence of Markov chains with approximate transition kernels," *Statistics and Computing*, vol. 26, no. 1-2, pp. 29–47, 2016.

[19] J. Park and M. Haran, "Bayesian inference in the presence of intractable normalizing functions," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1372–1390, 2018.

[20] J. Hoegerman and D. Losey, "Reward learning with intractable normalizing functions," *IEEE Robotics and Automation Letters Reprinted, with permission*, vol. 8, no. 11, pp. 7511–7518, 2023.

[21] S. Javdani, H. Admoni, S. Pellegrinelli, S. S. Srinivasa, and J. A. Bagnell, "Shared autonomy via hindsight optimization for teleoperation and teaming," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 717–742, 2018.

[22] A. Jonnavittula, S. A. Mehta, and D. P. Losey, "SARI: Shared autonomy across repeated interaction," *ACM Transactions on Human-Robot Interaction*, 2024.

[23] E. Rosen, D. Whitney, E. Phillips, G. Chien, J. Tompkin, G. Konidaris, and S. Tellex, "Communicating and controlling robot arm motion intent through mixed-reality head-mounted displays," *The International Journal of Robotics Research*, pp. 1513–1526, 2019.

[24] J. F. Mullen, J. Mosier, S. Chakrabarti, A. Chen, T. White, and D. P. Losey, "Communicating inferred goals with passive augmented reality and active haptic feedback," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8522–8529, 2021.

[25] A. Jonnavittula and D. P. Losey, "Communicating robot conventions through shared autonomy," in *IEEE International Conference on Robotics and Automation*, 2022, pp. 7423–7429.

[26] S. Habibian, A. A. Valdivia, L. H. Blumenschein, and D. P. Losey, "A review of communicating robot learning during human-robot interaction," *arXiv preprint arXiv:2312.00948*, 2023.

[27] M. Zolotas and Y. Demiris, "Towards explainable shared control using augmented reality," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2019, pp. 3020–3026.

[28] J. Hoegerman, S. Sagheb, B. Christie, and D. Losey, "Aligning learning with communication in shared autonomy," *IEEE/RSJ International Conference on Intelligent Robots and Systems, In review*, 2024.

[29] S. A. Mehta, Y. Kim, J. Hoegerman, M. D. Bartlett, and D. P. Losey, "Riso: Combining rigid grippers with soft switchable adhesives," in *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, 2023, pp. 1–8.

[30] M. Kalakrishnan, P. Pastor, L. Righetti, and S. Schaal, "Learning objective functions for manipulation," in *IEEE International Conference on Robotics and Automation*, 2013, pp. 1331–1336.

[31] D. Hadfield-Menell, S. Milli, P. Abbeel, S. J. Russell, and A. Dragan, "Inverse reward design," *NeurIPS*, 2017.

[32] S. Levine and V. Koltun, "Continuous inverse optimal control with locally optimal examples," in *ICML*, 2012, pp. 475–482.

[33] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, "An algorithmic perspective on imitation learning," *Foundations and Trends in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.

[34] C. L. Baker, R. Saxe, and J. B. Tenenbaum, "Action understanding as inverse planning," *Cognition*, vol. 113, no. 3, pp. 329–349, 2009.

[35] R. D. Luce, *Individual Choice Behavior: A Theoretical Analysis*. Courier Corporation, 2012.

[36] A. D. Dragan and S. S. Srinivasa, "A policy-blending formalism for shared control," *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 790–805, 2013.

[37] D. P. Losey, A. Bajcsy, M. K. O'Malley, and A. D. Dragan, "Physical interaction as communication: Learning robot objectives online from human corrections," *IJRR*, vol. 41, no. 1, pp. 20–44, 2022.

[38] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson Education, Inc., 2022.

[39] D. M. Taylor, "Americans with disabilities: 2014," *US Census Bureau*, pp. 1–32, 2018.

[40] T. Bhattacharjee, E. K. Gordon, R. Scalise, M. E. Cabrera, A. Caspi, M. Cakmak, and S. S. Srinivasa, "Is more autonomy always better? Exploring preferences of users with mobility impairments in robot-assisted feeding," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2020, pp. 181–190.

[41] R. M. Aronson, T. Santini, T. C. Kübler, E. Kasneci, S. Srinivasa, and H. Admoni, "Eye-hand behavior in human-robot shared manipulation," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 4–13.

[42] C. Brooks and D. Szafir, "Balanced information gathering and goal-oriented actions in shared autonomy," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2019, pp. 85–94.

[43] ——, "Visualization of intended assistance for acceptance of shared control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 11 425–11 430.

[44] D. P. Losey, C. G. McDonald, E. Battaglia, and M. K. O'Malley, "A review of intent detection, arbitration, and communication aspects of shared control for physical human–robot interaction," *Applied Mechanics Reviews*, vol. 70, no. 1, p. 010804, 2018.

[45] M. Fontaine and S. Nikolaidis, "A quality diversity approach to automatically generating human-robot interaction scenarios in shared autonomy," in *Robotics: Science and Systems*, 2020.

[46] A. Broad, I. Abraham, T. Murphey, and B. Argall, "Data-driven Koopman operators for model-based shared control of human–machine systems," *The International Journal of Robotics Research*, vol. 39, no. 9, pp. 1178–1195, 2020.

[47] S. Reddy, A. D. Dragan, and S. Levine, "Shared autonomy via deep reinforcement learning," in *Robotics: Science and Systems*, 2018.

[48] C. Schaff and M. R. Walter, "Residual policy learning for shared autonomy," in *Robotics: Science and Systems*, 2020.

[49] M. Hagenow, E. Senft, R. Radwin, M. Gleicher, B. Mutlu, and M. Zinn, "Corrective shared autonomy for addressing task variability," *IEEE Robotics and Automation Letters*, vol. 6, pp. 3720–3727, 2021.

[50] M. Zurek, A. Bobu, D. S. Brown, and A. D. Dragan, "Situational confidence assistance for lifelong shared autonomy," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 2783–2789.

[51] Y. Cui, S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh, "No, to the right: Online language corrections for robotic manipulation via shared autonomy," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 93–101.

[52] J. Z.-Y. He, Z. Erickson, D. S. Brown, A. Raghunathan, and A. Dragan, "Learning representations that enable generalization in assistive tasks," in *Conference on Robot Learning*, 2023, pp. 2105–2114.

[53] V. Alonso and P. De La Puente, "System transparency in shared autonomy: A mini review," *Frontiers in Neurorobotics*, 2018.