An Information Theoretic Formalism for Intent Disambiguation

Author Names Omitted for Anonymous Review. [Paper ID: **187**]

***Abstract*—The effectiveness of assistive robots is closely related to their ability to infer the user's needs and intentions *unambiguously* and provide appropriate assistance quickly and accurately. In this paper, we formulate the problem of intent disambiguation in information theoretic terms. We propose two different methods for intent disambiguation using the information theoretic concepts of *entropy* and *KL divergence*. The proposed algorithms characterize the disambiguation capabilities of the system through the forward projection of probability distributions over goals, utilizing an intent inference scheme and a kinematics model, and computing the average information gain and content. Previous studies have shown that the success of a disambiguation algorithm depends on a variety of factors and parameters. To thoroughly investigate the impact of these various components, we present results from an extensive simulation-based study for both a point robot and physics-based simulation of a six degrees-of-freedom (DoF) robotic arm. Our results indicate that, compared to a baseline, the proposed disambiguation algorithms do facilitate faster intent disambiguation, which in turn allows autonomy assistance to step in earlier during task execution. Additionally, we find that goal inference is more accurate and the total amount of time assistance is engaged is higher.**

I. Introduction

Assistive machines such as robotic arms and smart wheelchairs have the potential to transform the lives of millions of people with motor impairments. [12]. These machines can promote independence, enhance the quality of lives and revolutionize the way people interact with society. They can also help extend the mobility and manipulation capabilities of individuals, thereby helping motor-impaired people perform activities of daily living in a more effective manner.

An assistive robotic machine is typically controlled using an interface such as a joystick, a switch-based head array or a sip-and-puff. These interfaces are low-dimensional, low-bandwidth and occasionally discrete. For this reason, they can often only operate in a subset of the entire control space. These subsets are referred to as *control modes* [22]. To schedule and execute switches between control modes can be both mentally and physically demanding, and might be alleviated in part by the introduction of robotics autonomy. The efficacy of such autonomy-endowed machines relies on their ability to infer the users' needs and intentions, and is often a bottleneck for providing appropriate assistance quickly, confidently and accurately. Due to the dimensionality mismatch between high-dimensional robots and low-dimensional interfaces, the user is constrained to produce control commands that likely do not reveal their true intent. That is, the control interfaces act like filters that restrict the amount of information regarding the user's intent that gets passed through to the machine and autonomy.

In the context of assistive robotic manipulation, the first step of a task is often to reach towards discrete objects in the environment. Therefore, *intent inference* can be cast as a problem of computing the probability distributions over all possible discrete goals (objects) in the workspace. To estimate the most probable goal, intent inference mechanisms typically rely on various environmental cues and task-relevant features such as the robot and goal positions, human control commands and biometric measures [5]. In contrast, our system infers intent primarily based on the information contained in the constrained control commands issued to the assistive machine.

Our insight is that, at any given time, there is a specific set of control modes that are more informative *for* the autonomy than others. Therefore, by having the human provide control input within these information-rich modes, the possibility of inferring the human's intent accurately and unambiguously is increased. This, in turn, allows the autonomy to step in and *assist the human achieve their desired goal earlier*. This is important, because when the intent inference mechanism infers the incorrect goal (or is simply too slow), the user and autonomy often end up competing instead of collaborating. This can decrease user satisfaction and result in significantly worse performance than either the human or autonomy would produce on their own. In this paper, we address the problem of accurate intent inference by characterizing control modes according to the information content, and proposing a mode switch assistance paradigm that selects the control mode in which a user-initiated motion will *maximally disambiguate* their intent. We formalize the problem of intent disambiguation within the framework of information theory. Specifically, we make use of the information theoretic notions of entropy and Kullback-Leibler (KL) divergence to characterize and quantify the information content—with respect to intent disambiguation—of each control mode.

In Section II we present an overview of relevant research. Section III introduces our set theoretic treatment of control modes, followed by intent disambiguation and intent inference in Section IV. The study design and experimental methods are discussed in Section V followed, by results in Section VI. Discussion and conclusions are presented in Sections VII and VIII respectively.

II. Related Work

This section presents an overview of related works in information acquisition in robotics, intent inference in human-robot interaction and robot assistance for modal control.

Information theoretic approaches are widely utilized in the field of machine learning and robotics for optimal experiment design, for efficient data collection processes and for informing search strategies. Robot assistance schemes that seek to elicit more informative cues *from* the human to clarify the underlying intent can be thought of as an information acquisition problem. Intent acquisition can leverage the underlying synergies and shared intentionality [24] of human-robot teams and can be an active process in which the robot performs actions (for example, selecting a control mode or executing a robot motion) that will nudge the human to reveal her/his intent more clearly [19, 20]. Information theoretic metrics such as KL divergence can be utilized to identify regions of the sample space that will maximize information gain [25] and subsequently guide the data sampling process. Sensing robots designed for automated exploration and data acquisition tasks can benefit from exploring more information-rich regions in the environment [1]. If the spatial distribution of information density is known *a priori*, information maximization can be accomplished by maximizing the ergodicity of the robot's trajectory with respect to the underlying information density map [13, 14].

For an assistive machine to provide appropriate kinds of assistance accurately and at the right time, it needs to have a good estimate of the human's underlying intent. Intent inference algorithms, therefore, play a vital role in the success of an assistive system. Intent inference and recognition can be classified into two broad categories: heuristic approaches and model-based approaches. Heuristic approaches are often simpler and computationally light-weight and seek to find direct mappings between various task relevant features (such as motion cues) and the human's underlying intention [2, 3]. On the other hand, in model-based approaches the system maintains a model of how a human maps states to control actions. The model can either be learned from data or can be hand-designed based on domain knowledge. For example, the human can be modeled within the Partially Observable Markov Decision Process (POMDP) [7, 23] framework and is assumed to behave according to a control policy that maps states to actions. However, in model-based approaches incorporating the entire history of states requires estimation of joint probability distributions over past states which can become computationally expensive and intractable quickly.

When there is a mismatch between the dimensionality of the problem and the control interface users have to continuously shift their focus from the task at hand to the choice of control mode during task execution, thereby resulting in a higher cognitive load. To solve this problem, various mode switching assistance paradigms have been proposed to alleviate task effort. For example, an automatic time-optimal mode switch assistance has been proposed which has shown to significantly improve user satisfaction [9]. In the area of myoelectric prosthetics, dynamic switching approaches that learn the most effective control option during task execution using temporal difference and reinforcement learning have also been proposed [18].

III. Mathematical Notation

This section describes the mathematical notation used in our intent disambiguation algorithm that computes a control mode that maximally disambiguates between the various goals. We develop this algorithm specifically for robotic manipulation scenarios in which the user is controlling a robotic arm to reach for and interact with various discrete objects in the environment.

A. Probability Distribution Over Goals

In assistive robotic manipulation, intent inference is most commonly the process of estimating the user's intended goal out of a set of discrete objects in the environment [4]. The set of all candidate goals is denoted by $\mathcal{G}$ with $n_g = |\mathcal{G}|$ and let g^i refer to the i^{th} goal with $i \in [1, 2, \ldots, n_g]$. At any time t, the likelihood of goal g being the desired goal follows a categorical probability distribution whose sample space is $\mathcal{G}$. That is,

$$g \sim \textbf{Cat}(n_g|\boldsymbol{p}(t)) = \prod_{i=1}^{n_g} p^i(t)^{[g=g^i]}$$

where, $[g = g^i]$ is the Iverson bracket and evaluates to 1 if $g = g^i$, 0 otherwise. $\boldsymbol{p}(t)$ denotes the probability distribution over goals such that $\boldsymbol{p}(t) = [p^1(t), p^2(t), \ldots, p^{n_g}(t)]^T$ where $p^i(t)$ denotes the probability associated with goal g^i. The probability $p^i(t)$ also represents the robot's *confidence* that goal g^i is the human's intended goal.

B. Set Theoretic Treatment of Control Modes

The low dimensionality of the control interfaces necessitates the control space to be partitioned into control modes. Let $\mathcal{K}$ be the set of all controllable dimensions of the robot and k^i represent the i^{th} control dimension where $i \in [1, 2, \ldots, n_k]$ with $n_k = |\mathcal{K}|$. The number of controllable dimensions (n_k) depends on the robotic platform; for example, a 2D point robot that operates in $\mathbb{R}^2$ has $n_k = 2$ whereas $n_k = 6$ for a 6-DoF robotic manipulator.

Let $\mathcal{M}$ denote the set of all control modes with $n_m = |\mathcal{M}|$. Additionally, let m^i refer to the i^{th} control mode where $i \in [1, 2, \ldots, n_m]$. Each control mode m^i is a subset of $\mathcal{K}$ such that $\bigcup_{i=1}^{n_m} m^i = \mathcal{K}$. The cardinality of each mode $m \in \mathcal{M}$, denoted by $|m|$ indicates the number of dimensions that can be controlled when operating in m.[1] Furthermore, the user can only operate in one of the n_m control modes at any given time t. That is, for each $m \in \mathcal{M}$, the subspace of $\mathcal{R}^{n_k}$ that is accessible corresponds to $\mathcal{R}^{|m|}$ whose orthonormal basis vectors are given by $\boldsymbol{e}^k \ \forall \ k \in m$. Maximum velocity limits along each dimension impose further constraints on the set of control commands that are available in each mode. This constrained set of control commands available in mode m, denoted by $\mathcal{U}^m$, can be written as

$$\mathcal{U}^m = \{\boldsymbol{u} | \boldsymbol{u} \in \mathcal{R}^{|m|} \ and \ \|\boldsymbol{u}\|_\infty \leq v_{lim}\}$$

[1] Note that a dimension $k \in \mathcal{K}$ can be an element of multiple control modes.

where v_{lim} denotes the maximum velocity along any dimension k and $\|\cdot\|_\infty$ denotes the L_∞ norm.

IV. Intent Disambiguation

This section describes our two proposed approaches for intent disambiguation using information theoretic concepts and the different intent inference schemes that work in conjunction with the disambiguation algorithm.

A. Need For Intent Disambiguation

Our aim is to develop a metric that will capture the "disambiguation capability" of a control dimension/mode. Subsequent user operation of the robot in the control mode with maximum disambiguation capability will likely help the system to perform better intent inference and may result in the autonomy providing more appropriate assistance.

At any given time t, the probability distributions over goals encodes the human's underlying intent. The time evolution of the probability distribution, however, depends on the choice of intent inference scheme and the various task relevant features and parameters that contribute to it. If the evolution is sensitive to the user control command (denoted by $\boldsymbol{u_h}$) then it is likely that the goal probabilities evolve differently as the user operates the robot in different control modes. User-generated motion along certain control dimensions therefore might reveal the human's intent with less ambiguity. That is, motion within maximally disambiguating control dimensions serves as a mechanism to enhance the accuracy of intent inference. Various types of intent inference schemes used in conjunction with the proposed disambiguation algorithm in our paper are described in detail in Section IV-E.

B. Forward Projection of $\boldsymbol{p}(t)$

The first step towards the computation of the disambiguation metric D_m for each $m \in \mathcal{M}$ is the forward projection of the probability distribution $\boldsymbol{p}(t)$ from current time t_a to t_b such that $t_a < t_b$. The exact computation of the projected probability distribution will depend on the underlying intent inference computation—for example, whether it depends on the robot position ($\boldsymbol{x_r}$), which can be computed from the human control command (denoted by $\boldsymbol{u}_h$) applied to the robot kinematics model or $\boldsymbol{u}_h$. All parameters and features which affect the computation of $\boldsymbol{p}(t)$ are denoted as $\boldsymbol{\Theta}$. If the inference mechanism depend on the control command, the projected probability distribution at time t_b given by $\boldsymbol{p}(t_b)$ depends on $\boldsymbol{u_h}$. For the purposes of computation of the disambiguation metric, in each mode $m \in \mathcal{M}$, we only consider a subset of control commands from $\mathcal{U}^m$, and is denoted by $Vert(\mathcal{U}^m)$. The set $Vert(\mathcal{U}^m)$ is given by

$$Vert(\mathcal{U}^m) = \{\boldsymbol{u} \in \mathcal{U}^m \mid \boldsymbol{u} = \underset{\boldsymbol{u}\in\mathcal{U}^m}{\operatorname{argmax}} \|\boldsymbol{u}\|_2\}$$

where $\|\cdot\|_2$ is the Euclidean norm.[2]

[2] $Vert(\mathcal{U}^m)$ corresponds to the "vertices" of the $|m|$-dimensional hypercube that is the space of control commands accessible from mode m.

The probabilities are forward projected for each of the possible control commands available in $Vert(\mathcal{U}^m)$ and a weighted average of the disambiguation metric computations on each of the projected probabilities is used to characterize the control mode m (Algorithm 1, Lines 3-11). In the following subsections we present two different methods to compute the disambiguation metric for a given mode m.

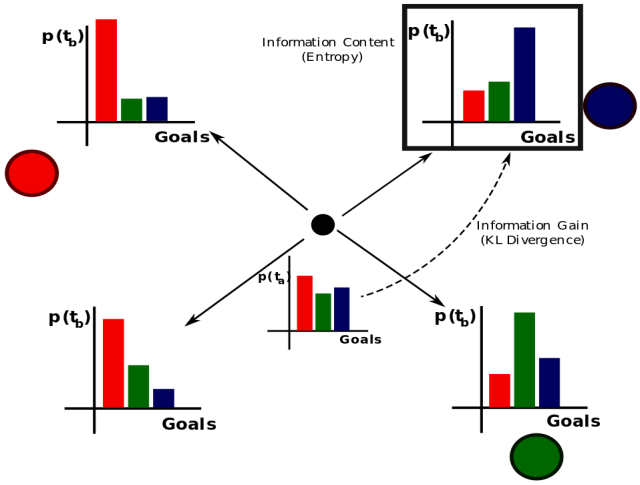

Fig. 1. Illustration of computation of D_m in $\mathbb{R}^2$. The goals are shown in red, green and blue and the robot in black. The probability distributions are forward projected for all $\boldsymbol{u_h} \in Vert(\mathcal{U}^m)$. The change in the overall shape of the probability distribution upon evolving from $\boldsymbol{p}(t_a)$ to $\boldsymbol{p}(t_b)$ amounts to the information *gain* and is captured by the *KL divergence*. The entropy of $\boldsymbol{p}(t_b)$ captures the information *content* in the projected distribution.

C. Entropy Disambiguation Metric

The entropy of a probability distribution captures the average information content of a stochastic source of data. Lower entropy indicates higher certainty in the value of the random variable and vice-versa. Therefore, in the context of intent disambiguation entropy of the projected probability distribution, $\boldsymbol{p}(t_b)$, can be used as a measure of how confident the system is in its prediction of human intent. That is, entropy can be used a measure of disambiguation. Lower the entropy better the disambiguation due to higher certainty in the human's intended goal. For a discrete random variable X with possible values $\{x_1, x_2, \ldots x_n\}$, the Shannon entropy is defined as

$$ENT(P(X)) = -\sum_{i=1}^{n} P(x_i) log_2 P(x_i)$$

where $P(X)$ denotes the probability mass function. The disambiguation capability of a control mode m is characterized by computing a weighted average of the entropies of projected probability distributions for all possible control commands $\boldsymbol{u_h} \in Vert(\mathcal{U}^m)$. That is

$$D_m = \frac{1}{|Vert(\mathcal{U}^m)|} \sum_{\boldsymbol{u_h}\in Vert(\mathcal{U}^m)} ENT(\boldsymbol{p}(t_b; \boldsymbol{u_h}))$$

Algorithm 1 Calculate $\boldsymbol{p}(t_b)$, D_m

Require: $\boldsymbol{p}(t_a), \boldsymbol{x}_r(t_a), \Delta t, t_a < t_b, \boldsymbol{\Theta}$

1: **for** $m = 1 \dots n_m$ **do**
2: Initialize $D_m = 0$
3: **for** $\boldsymbol{u_h} \in Vert(\mathcal{U}^m)$ **do**
4: Initialize $t = t_a, \boldsymbol{x_r}(t) = \boldsymbol{x_r}(t_a), \boldsymbol{p}(t) = \boldsymbol{p}(t_a)$
5: **while** $t \leq t_b$ **do**
6: $\boldsymbol{p}(t + \Delta t) \leftarrow$ UpdateIntent$(\boldsymbol{p}(t), \boldsymbol{u}_h; \boldsymbol{\Theta})$
7: $\boldsymbol{x}_r(t + \Delta t) \leftarrow$ SimulateKinematics$(\boldsymbol{x}_r(t), \boldsymbol{u}_h)$
8: $t \leftarrow t + \Delta t$
9: **end while**
10: $D_m \leftarrow D_m + \frac{1}{|Vert(\mathcal{U}^m)|} ENT(\boldsymbol{p}(t_b))$
or
11: $D_m \leftarrow D_m + \frac{1}{|Vert(\mathcal{U}^m)|} KL(\boldsymbol{p}(t_b)||\boldsymbol{p}(t_a))$
12: **end for**
13: **end for**

where $\boldsymbol{p}(t_b; \boldsymbol{u_h})$ denotes the projected probability distribution at time t_b when the control command used for forward projection is $\boldsymbol{u_h}$.

D. KL Divergence Disambiguation

Although entropy can capture information *content*, intent disambiguation can likely benefit from the quantification of the information *gain* regarding the human's intended goal as a result of user-initiated motion in a control mode m. KL divergence, also known as relative entropy, measures how a probability distribution differs from another distribution. KL divergence is widely used in the context of Bayesian inference to compute the information gain when the prior is updated to the posterior in the light of new evidence. In the context of disambiguation we can treat the projected probability distribution at time t_b as the posterior and the distribution at time t_a to be the prior. KL divergence can then be used to characterize the *information gain* regarding the human's intended goal as a result of the application of $\boldsymbol{u_h}$. For a discrete random variable X with possible values $\{x_1, x_2, \dots x_n\}$ the KL divergence is defined by

$$KL(P||Q) = -\sum_{i=1}^{n} P(x_i) log_2 \frac{Q(x_i)}{P(x_i)}$$

where $P(X)$ and $Q(X)$ are two different probability mass distributions. More specifically, the disambiguation capability of control mode m is computed by averaging the information gain for all possible control commands $\boldsymbol{u_h} \in Vert(\mathcal{U}^m)$. That is the disambiguation metric can be computed as

$$D_m = \frac{1}{|Vert(\mathcal{U}^m)|} \sum_{\boldsymbol{u_h} \in Vert(\mathcal{U}^m)} KL(\boldsymbol{p}(t_b; \boldsymbol{u_h})||\boldsymbol{p}(t_a)) \quad .$$

The control mode with highest disambiguation capability m^* is given by $\text{argmax}_m D_m$. Disambiguation mode m^* is the control mode that system chooses *for* the human to better estimate their intent. Any control command issued by the human when operating in m^* is likely to be more beneficial for the system to determine the human's intended goal.

E. Intent inference

The disambiguation power of our algorithm is closely linked to the inference power of different choices of inference mechanisms. The update rule for the probability distribution over goals is determined by the choice of inference schemes (Algorithm 1, Line 6) as a result of which the projected probability distributions, $\boldsymbol{p}(t_b)$ and the disambiguation metric D_m, likely are going to be different for different choices. This section describes the different types of intent inference schemes that work in conjunction with the disambiguation algorithms proposed in Section IV.

1) Heuristic Approaches: Heuristic approaches based on *confidence* functions [7] seek to find direct mappings between instantaneous cues and underlying human intentions. For every goal $g \in \mathcal{G}$, the system maintains an associated set of confidences denoted by $\mathcal{C}$. The system designer has the freedom to choose the set of features that will inform the confidence functions. For example, a simple proximity-based confidence function used extensively in literature is

$$c(\boldsymbol{x_r}, \boldsymbol{x_g}) = max\Big(0, 1 - \frac{\|\boldsymbol{x_r} - \boldsymbol{x_g}\|}{r}\Big)$$

where $\boldsymbol{x_r}$ is the robot position, $\boldsymbol{x_g}$ is the goal position, r is the radius beyond which the confidence function is always 0 and $\|\cdot\|$ is an appropriate distance metric. A slightly more information-rich variant that aims to capture the "directedness" of the human control command to a particular goal position is

$$c(\boldsymbol{x_r}, \boldsymbol{x_g}, \boldsymbol{u}_h) = \boldsymbol{u}_h \cdot (\boldsymbol{x}_g - \boldsymbol{x_r})$$

where $\boldsymbol{u_h}$ is the human control command. These confidence functions rely on instantaneous features and therefore are amnesic and can exhibit chatter behavior [6].

2) Bayesian Approaches: Bayesian approaches for intent inference consist of iteratively updating of the belief (probability distribution over goals) as new evidence arrives at every time-step. The Bayesian update equation for the probability distribution over goals is given by

$$\underbrace{p(g_t|\boldsymbol{u_h}^{1:t}, \boldsymbol{\Theta})}_{posterior} = \eta \underbrace{p(\boldsymbol{u_h}^{1:t}|g_t, \boldsymbol{\Theta})}_{likelihood} \underbrace{p(g_{t-1})}_{prior}$$

where η is the normalization factor, $\boldsymbol{u_h}^{1:t}$ is the entire history of control commands and $\boldsymbol{\Theta}$ denoted all other task-relevant and environment features that inform the inference process. The likelihood function is related to how the human teleoperates the robot in order to accomplish a goal and can be either learned from data, or be hand-designed utilizing domain knowledge. Under first-order Markovian assumption the update equation can be simplified by only considering the control command at the current time step t.

3) Dynamic Neural Field Approaches: Dynamic neural fields are differential equations in time that governs the time evolution of dynamical state variables, with some specific properties such as recurrent interactions between the state variables, robustness to external noise and memory. Dynamic neural fields were originally conceived to explain cortical population neuronal dynamics, based on the hypothesis that

the excitatory and inhibitory neural interactions between local neuronal pools form the basis of cortical information processing [21]. When applied to the problem of intent inference, the individual goal probabilities are treated as constrained dynamical state variables whose time evolution is determined by a dynamic neural field such that $p^i(t) \in [0,1]$ and $\Sigma_1^{n_g} p^i(t) = 1$ [17].

The full specification of the neural field is given by

$$\frac{\partial \boldsymbol{p}(t)}{\partial t} = \frac{1}{\tau}\Bigg[-\mathbb{I}_{n_g \times n_g} \cdot \boldsymbol{p}(t) + \underbrace{\frac{1}{n_g} \cdot \mathbb{1}_{n_g}}_{\text{rest state}} \Bigg] + \underbrace{\boldsymbol{\lambda}_{n_g \times n_g} \cdot \sigma(\boldsymbol{\xi}(\boldsymbol{u}_h; \boldsymbol{\Theta}))}_{\text{excitatory + inhibitory}} \quad (1)$$

where time-scale parameter τ determines the memory capacity of the system, $\mathbb{I}_{n_g \times n_g}$ is the identity matrix, $\mathbb{1}_{n_g}$ is a vector of dimension $n_g \times 1$ containing all ones, $\boldsymbol{\lambda}$ is the control matrix that controls the excitatory and inhibitory aspects, $\boldsymbol{\xi}$ is a function that encodes the nonlinearity through which human control commands and task features affect the time evolution, $\boldsymbol{\Theta}$ represents all other task-relevant features and parameters that affect the time-evolution of the probability distribution, and σ is a biased sigmoidal nonlinearity given by $\sigma(\boldsymbol{\xi}) = \frac{1}{1+e^{-\boldsymbol{\xi}}} - 0.5$. Given the initial probability distribution at time t_a Equation 1 can be solved numerically from $t \in [t_a, t_b]$ using a simple Euler algorithm with a fixed time-step Δt. The design of $\boldsymbol{\xi}$ is informed by what features of the human control input and environment capture the human's underlying intent most effectively. We rely on the *directedness* of the human control commands towards a goal, the *proximity* to a goal and the *agreement* between the human commands and robot autonomy.

V. Experimental Evaluation

We evaluate the efficacy of the proposed disambiguation algorithm on simulated point robots and a simulated robotic manipulator that operates in a *shared-control* setting. This section describes the shared-control paradigm used in our experiments and the details of the simulation experiment protocol.

A. Shared Control

Shared autonomy in assistive robotics aims to reduce the human's physical and cognitive burden during task execution without having the user to completely cede manual control. Some common shared-control approaches include (a) hierarchical paradigms in which the choice of a high-level goal is entrusted to the user and the autonomy generates low-level control [11] and (b) signal-level paradigms in which the user and robot's control commands are blended together [15]. Shared control systems often require a good estimate of the human's intent to determine when and how to share control. Therefore, intent inference and disambiguation play a vital role in the seamless operation and success of a shared control system. In our experimental evaluation, control sharing is achieved with a blending-based paradigm in which the final control command issued to the robot is a weighted linear combination of the user control command and robot autonomy.

1) Robot Autonomy: The robot autonomy command $\boldsymbol{u}_{r,g}$ is generated using a simple potential field which is defined in all parts of the state space [10]. Every goal g is associated with a potential field γ_g which treats g as an attractor and all the other goals in the scene as repellers. That is,

$$\dot{\boldsymbol{x}}_r^{attract} = \boldsymbol{x}_g - \boldsymbol{x_r}$$

where $\dot{\boldsymbol{x}}_r$ indicates the velocity of the robot, $\boldsymbol{x}_g$ is the location of goal g. Each dimension of $\dot{\boldsymbol{x}}_r^{attract}$ can be interpreted as the magnitude of the *attractive pull* towards the goal along that dimension. The repeller velocity is given by

$$\dot{\boldsymbol{x}}_r^{repel} = \sum_{i \in \mathcal{G} \setminus g} \frac{\boldsymbol{x_r} - \boldsymbol{x}_{g^i}}{\mu(\left\|\boldsymbol{x_r} - \boldsymbol{x}_{g^i}\right\|^2)}$$

μ controls the magnitude of the repeller velocity. The autonomy command $\boldsymbol{u}_{r,g}$ is computed as a summation of the attractor and repeller velocities.

2) Blending Paradigm: The final control command $\boldsymbol{u_f}$ issued to the robot then is given by

$$\boldsymbol{u_f} = \alpha \cdot \boldsymbol{u}_{r,g^*} + (1-\alpha) \cdot \boldsymbol{u}_h$$

where g^* is the most confident goal and corresponds to the mode of $\boldsymbol{p}(t)$. The blending parameter, α is a piecewise linear function of the $p(g^*)$ and is specified as

$$\alpha = \begin{cases} 0 & p(g^*) \leq \rho_1 \\ \frac{\rho_3}{\rho_2 - \rho_1} \cdot p(g^*) & \text{if} \quad \rho_1 < p(g^*) \leq \rho_2 \\ \rho_3 & p(g^*) > \rho_2 \end{cases}$$

with $\rho_i \in [0,1] \ \forall \ i \in [1,2,3]$ and $\rho_2 > \rho_1$. In our implementation, we empirically set $\rho_1 = \frac{1.2}{n_g}, \rho_2 = \frac{1.4}{n_g}$ and $\rho_3 = 0.7$.

3) Maximum Potential Mode Switch Assistance: As a baseline for the simulation experiments described in Section V-B we utilize a simple greedy mode switch assistance scheme. At any given time t, the system selects the control mode which has the *maximum attractive pull* or *maximum potential* towards the predicted goal. For each control dimension k, the magnitude of the k^{th} dimension of $\dot{\boldsymbol{x}}_r^{attract}$ is a measure of the attractive potential along k. The attractive potential of each control mode $m \in \mathcal{M}$ is computed as the total attractive potential of all control dimensions $k \in m$.

B. Point Robot Simulation Setup

1) Robot Workspace: Our simulations were performed on point robots that reside in $\mathbb{R}^2$, $\mathbb{R}^3$, $\mathbb{SE}(2)$ and $\mathbb{SE}(3)$ spaces. The translational workspace limits were set at $[-0.6m, 0.6m]$ along each translational dimension and the orientation limits were set at $[0, 2\pi]$ along each rotational axis. Table I lists all the factors there were manipulated for each trial.

Variable	Range
n_g	$[2, 6]$
$\mathcal{M}$	Randomly selected from Table II conditioned on the space
Intent Inference	['Confidence', 'Bayes', 'Dynamic Neural Field']
Initial Robot Position	Randomized within workspace limits
Goal Positions	Randomized within workspace limits
Intended Goal	Randomly chosen from $\mathcal{G}$

TABLE I
RANDOMIZED FACTORS FOR EACH SIMULATION TRIAL

2) Mode Switch Schemes: Four different mode switching schemes were activated during each simulation trial, three of which performed intent disambiguation and one which served as a baseline for comparison. The disambiguation algorithms were entropy disambiguation metric (Section IV-C), KL divergence disambiguation metric (Section IV-D) and a heuristic approach developed in a related work [8]. We used the mode switching scheme described in Section V-A3 as the baseline for comparison.

3) Simulated $\boldsymbol{u_r}$ and $\boldsymbol{u_h}$: The simulated robot autonomy $\boldsymbol{u_r^{sim}}$ is generated using a repeller-free potential field. In a given control mode $m \in \mathcal{M}$, the simulated human control command denoted by $\boldsymbol{u_h^{sim}}$ is given by

$$\boldsymbol{u_h^{sim}} = \underset{\boldsymbol{u_h} \in Vert(\mathcal{U}^m)}{\operatorname{argmax}} \; \boldsymbol{u_h} \cdot (\boldsymbol{x_g}^{intended} - \boldsymbol{x_r})$$

where $\boldsymbol{x_g}^{intended}$ is the position of the human's intended goal which is known only to the human and the system tries to infer using the inference mechanism. This model of simulated human control command assumes that the user issues control commands at all times and always tries to optimize for distance and time. The final control command issued to the point robot was determined by the blending-based shared control scheme described in Section V-A2.

4) Metrics: The following metrics were used to evaluate the efficacy of the disambiguation algorithm compared to the baseline mode switch assistance:

- *Initial Onset of Assistance*: Earliest time (normalized with respect to individual trial time) at which assistance was triggered towards the intended goal. This measure captures how *early* the robot was able to infer the correct goal *and* provide assistance.
- *Total Amount of Assistance* : Fraction of the total trial time for which assistance towards the intended goal was present.
- *Inference Accuracy* : Fraction of the total trial time for which the intent inference mechanism correctly inferred the human's intended goal. This is a measure of overall confidence of the system.

5) Simulation Protocol: 5000 simulation runs were performed for each of the scenarios. The maximum trial duration was set at $12s$ or equivalently 120 time-steps with $\Delta t = 0.1$. The mode switch assistance algorithm was activated once every 4 time-steps. The trial was ended prematurely if the robot reached within the intended goal as determined by a pre-specified distance threshold. Figure 2 illustrates an example of the simulation for a point robot in $\mathbb{R}^2$.

Space	n_k	Control Mode Sets
$\mathbb{R}^2$	2	$\{[1], [2]\}$
$\mathbb{R}^3$	3	$\{[1], [2], [3]\}, \{[1, 2], [3]\}$
$\mathbb{SE}(2)$	3	$\{[1], [2], [3]\}, \{[1, 2], [3]\}$
$\mathbb{SE}(3)$	6	$\{[1], [2], [3], [4], [5], [6]\}$, $\{[1, 2, 3], [4, 5, 6]\}$, $\{[1, 2], [1, 3], [4, 5], [6]\}$, $\{[1, 2], [1, 3], [4, 5], [6]\}$

TABLE II
PREDEFINED CONTROL SPACE PARTITION SETS FOR EACH OF THE SIMULATION SPACES.

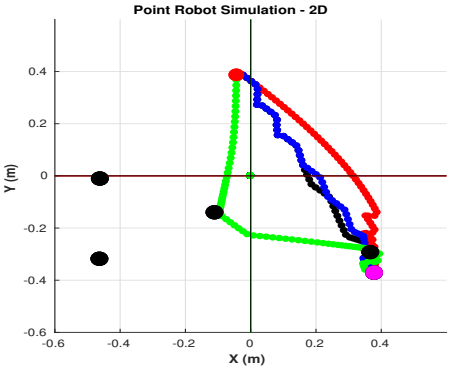

Fig. 2. Example of point robot simulation in $\mathbb{R}^2$. The goals are represented in black, the initial robot position in red and the human's intended goal in magenta. The different colored trajectories correspond to different mode switching schemes. Greedy (Black), Entropy (Red), KL Divergence (Blue) and Heuristic (Green).

C. Robotic Arm Simulation Setup

We also evaluated our system in a physics-based simulation of a 6-DoF robotic arm (MICO robotic arm from Kinova Robotics). Although the end effector of the robot lives in $\mathbb{SE}(3)$, the physical constraints of the robot make the kinematics highly nonlinear compared to a point robot in $\mathbb{SE}(3)$.

We evaluated the system on two reaching tasks: 1) Four goals with same orientation and 2) five goals with different orientations. Two mode switching schemes were tested: KL divergence disambiguation and the maximum potential mode switching. During each trial, the mode switching algorithm was activated every $5s$ and the maximum trial duration was set at $40s$. The robot autonomy was generated using a potential field as described in Section V-A1 and the control interface mapping was 1D and discrete. Additionally, a *sparsity* factor was added to the simulated human control command $\boldsymbol{u_h^{sim}}$ to capture the variability in command timing exhibited by humans during robot teleoperation. This sparsity factor captured the *idle time* users have during task execution when teleoperating a robot and was randomly set between 20% and 80% as informed by the analysis of teleoperation data from previous experiments conducted in our lab.

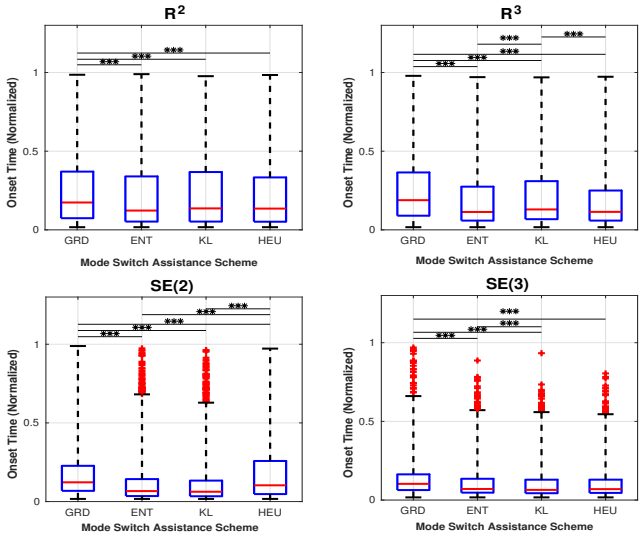

Fig. 3. Initial onset of assistance. Group differences between mode switch assistance schemes (GRD, ENT, KL, HEU), in four operational spaces $\mathbb{R}^2$, $\mathbb{R}^3$, $\mathbb{SE}(2)$ and $\mathbb{SE}(3)$. Box plots show the median and the quartiles.

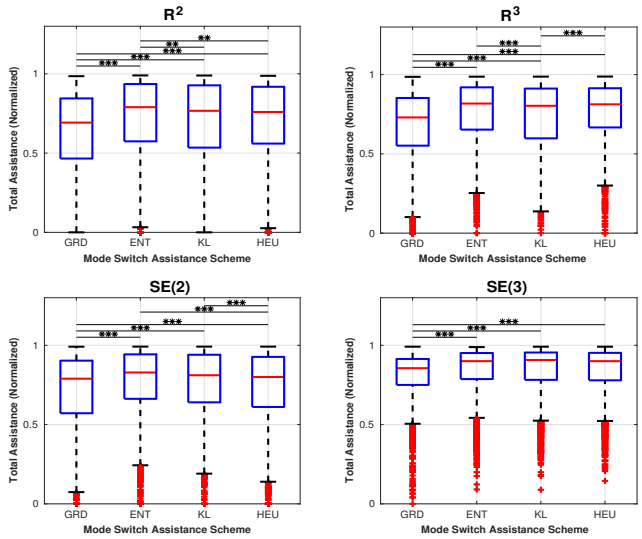

Fig. 4. Total amount of assistance. Group differences between mode switch assistance schemes (GRD, ENT, KL, HEU), in four operational spaces $\mathbb{R}^2$, $\mathbb{R}^3$, $\mathbb{SE}(2)$ and $\mathbb{SE}(3)$. Box plots show the median and the quartiles.

VI. Results

In this section we present results from our simulation studies conducted on a simulated 6-DoF robotic arm and point robots in $\mathbb{R}^2$, $\mathbb{R}^3$, $\mathbb{SE}(2)$ and $\mathbb{SE}(3)$.

A. Point Robot Simulation

One way analysis of variance was performed using Kruskal-Wallis procedure to test for group effects between the four mode switch assistance schemes: greedy (GRD), entropy (ENT), KL-Divergence (KL) and heuristic (HEU). Post-hoc analysis between groups was conducted using a multiple comparison test that used Tukey's HSD criterion. All analysis was performed in MATLAB. In all the data plots, (*) indicates $p < 0.05$, (**) indicates $p < 0.01$ and (***) indicates $p < 0.001$.

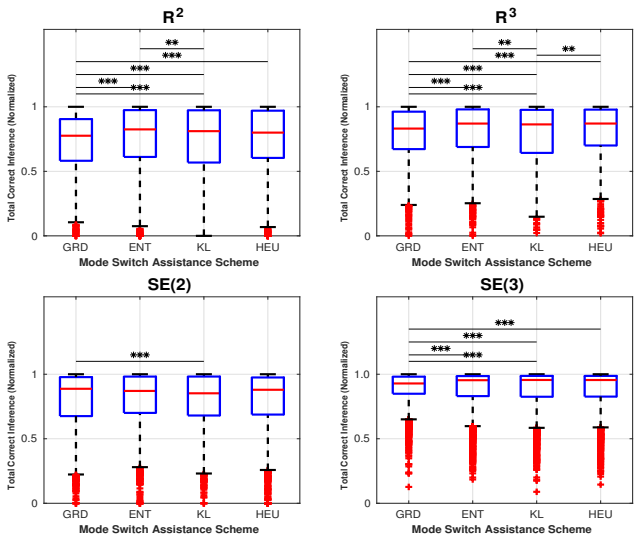

Fig. 5. Inference Accuracy. Group differences between mode switch assistance schemes (GRD, ENT, KL, HEU), in four operational spaces $\mathbb{R}^2$, $\mathbb{R}^3$, $\mathbb{SE}(2)$ and $\mathbb{SE}(3)$. Box plots show the median and the quartiles.

1) Initial Onset of Assistance: The Kruskal-Wallis test reveals that the group ranks are different for simulations in all operational spaces. Post-hoc analysis also reveals statistically significant differences between the baseline mode switch assistance scheme and at least one of the disambiguation schemes for all operational spaces (Figure 3). This shows that when the disambiguation scheme is used, the autonomy is able to provide assistance *earlier* during task execution. This is likely due to the fact that with disambiguation the system is able to elicit more informative control commands from the user and narrow down its prediction to the correct goal quicker during task execution. As a result, the confidence in the prediction becomes higher and the robot assistance is triggered earlier during a trial.

2) Amount of Total Assistance: For all operational spaces, Figure 4 shows a statistically significant increase in the total amount of assistance when disambiguation schemes are active compared to the baseline. The shared control paradigm used in our setup is designed in such a way that the robot assistance gets triggered when the confidence associated with the goal prediction crosses a predefined threshold. These results indicate that with disambiguation the system is able to maintain the confidence in its prediction of the correct goal above the threshold for a longer duration during the trial, resulting in greater overall assistance during the course of the task.

3) Inference Accuracy: As shown in Figure 5, for all operational spaces, the accuracy of intent inference is higher for at least one disambiguation scheme compared to the baseline. This is likely because of the fact that robot operation in disambiguating modes results in higher information gain regarding the underlying intent thereby helping the system to perform accurate intent inference.

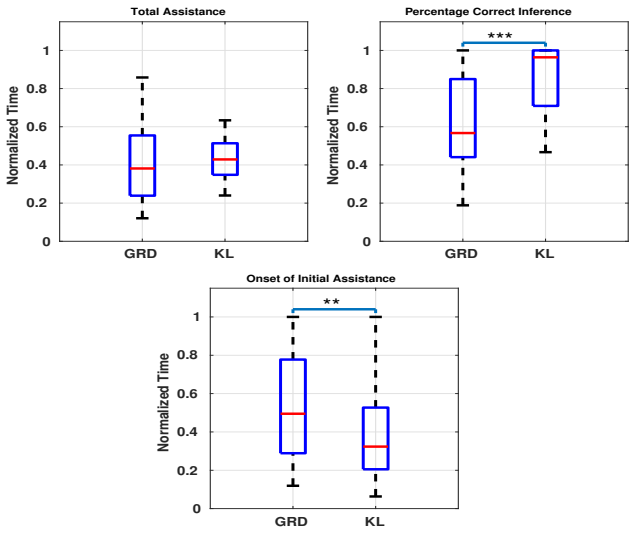

Fig. 6. Results with the simulated 6-DoF robotic arm. Top Left: Fraction of total time for which assistance towards the intended goal was engaged. Top Right: Accuracy of intent inference procedure measured in fraction of total trial time. Bottom: Onset of initial assistance as a fraction of total trial time. Statistical significance was computed using the Wilcoxon Rank-Sum test.

B. Robotic Arm Simulation

Figure 6 reveals that for the simulations conducted on a simulated robotic arm, the onset of initial assistance towards the correct goal is earlier compared to that of the baseline ($p < 0.05$). Similarly, the accuracy of the intent inference procedure is higher when disambiguation algorithm is deployed ($p < 0.001$). With disambiguation the total amount of assistance provided by the system also is slightly higher, though not significantly so, compared to the baseline mode-switching scheme. These results indicate that even with a low-dimensional and discrete control interface that exhibits signal dropouts (due to the sparsity factor), robot operation in control modes selected by the disambiguation algorithm helps the autonomy to elicit more intent-revealing control commands thereby helping the system predict the intended goal more accurately and provide assistance earlier.

VII. Discussion

The effectiveness of an assistive machine in a shared control setting is closely related to its ability to infer the user's intent accurately and unambiguously. Accurate goal inference can improve the trust and communication between the human and the robot and can result in seamless and fluid human robot interaction. On the other hand, incorrect goal inference can likely lead to frustrating user experience and can often have a detrimental effect on task performance. Our results indicate that disambiguation helps to enhance the accuracy of the intent inference procedure and results in an increase in overall assistance provided to the user. Yet another factor that is crucial for the success of a shared control system, especially for safety-critical applications, is *when* during task execution the assistance gets activated. For example, in the shared control of complex systems that exhibit highly nonlinear and unstable dynamics typically the role of autonomy is to maintain stability and safety. In such situations, if the user operation leads to the destabilization of the system, failure to intervene earlier during task execution can lead to catastrophic results. From our simulation results, we can see that intent disambiguation enhances the inference capabilities of the system and helps the autonomy to provide assistance earlier.

The utility value of a mode-switching scheme can vary during the time course of task execution. For instance, in the beginning stages of task execution when the intent is unclear, disambiguation algorithms are likely to be more useful. Once task execution gets underway and intent is clarified, a greedy time-optimal mode switching scheme can be used to lead the user towards successful task completion. That is, a more optimal strategy might be one which is adaptive and chooses different schemes depending on the context. This could possibly be cast within a control theoretic framework as a mode-scheduling problem in which the question of *when* to deploy *which* mode-switching paradigm can be solved concurrently.

High fidelity models for how a user teleoperates a robot and robot kinematics can be learned from data using machine learning tools such as deep neural networks [16], Gaussian processes [26]. With sufficient data these techniques have the capability to capture the inherent nonlinearities in human-robot systems. Characterization of the disambiguation capabilities of different control modes can benefit a great deal from more accurate models of robot kinematics and user teleoperation. Furthermore, the disambiguation algorithm can be customized to specific users using real-time learning algorithms that capture individual teleoperation characteristics.

VIII. Conclusion

In this paper, we formalized the problem of intent disambiguation within the framework of information theory. We introduced two different methods for intent disambiguation using the information theoretic concepts of entropy and KL divergence. We also present results from an extensive simulation-based study for point robots and a physics-based simulation of a robotic arm. Results from the study indicate that, compared to the baseline, the proposed disambiguation algorithms facilitated faster intent disambiguation which in turn allowed the autonomy to assist earlier during task execution. Goal inference was more accurate and the total amount of assistance engaged was also higher. In our future work, we plan to perform an in-depth analysis of the data to expose the interactions between various factors. Knowledge of how different components interact and influence each other is critical for the design of a successful shared control system. Furthermore, as informed by the simulation results, we plan to evaluate our system with an extensive user study.

Acknowledgments

Funding source omitted for review.

References

[1] Nikolay Atanasov, Jerome Le Ny, Kostas Daniilidis, and George J. Pappas. Information acquisition with sensing robots: Algorithms and error bounds. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[2] Chris L. Baker, Joshua B. Tenenbaum, and Rebecca R. Saxe. Goal inference as inverse planning. In *Proceedings of the Cognitive Science Society*, 2007.

[3] Chris L. Baker, Rebecca R. Saxe, and Joshua B. Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3):329–349, 2009.

[4] Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. The YCB object and model set: Towards common benchmarks for manipulation research. In *Advanced Robotics (ICAR), 2015 International Conference on*, pages 510–517. IEEE, 2015.

[5] DKEA Croft. Estimating intent for human-robot interaction. In *IEEE International Conference on Advanced Robotics*, pages 810–815. Citeseer, 2003.

[6] Anca D. Dragan and Siddhartha S. Srinivasa. *Formalizing assistive teleoperation*. MIT Press, 2012.

[7] Anca D. Dragan and Siddhartha S. Srinivasa. A policy-blending formalism for shared control. *The International Journal of Robotics Research*, 32(7):790–805, 2013.

[8] Deepak Gopinath and Brenna Argall. Mode switch assistance to maximize human intent disambiguation. In *Robotics: Science and Systems*, 2017.

[9] Laura V. Herlant, Rachel M. Holladay, and Siddhartha S. Srinivasa. Assistive teleoperation of robot arms via automatic time-optimal mode switching. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016.

[10] Oussama Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1):90–98, 1986.

[11] Dae-Jin Kim, Rebekah Hazlett-Knudsen, Heather Culver-Godfrey, Greta Rucks, Tara Cunningham, David Portee, John Bricout, Zhao Wang, and Aman Behal. How autonomy impacts performance and satisfaction: Results from a study with spinal cord injured subjects using an assistive robot. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 42(1):2–14, 2012.

[12] Mitchell P LaPlante et al. Assistive technology devices and home accessibility features: prevalence, payment, need, and trends. *Advance Data from Vital and Health Statistics*, 1992.

[13] Lauren M. Miller and Todd D. Murphey. Trajectory optimization for continuous ergodic exploration. In *American Control Conference (ACC)*, 2013.

[14] Lauren M. Miller, Yonatan Silverman, Malcolm A. MacIver, and Todd D. Murphey. Ergodic exploration of distributed information. *IEEE Transactions on Robotics*, 32(1):36–52, 2016.

[15] Katharina Muelling, Arun Venkatraman, Jean-Sebastien Valois, John E Downey, Jeffrey Weiss, Shervin Javdani, Martial Hebert, Andrew B Schwartz, Jennifer L Collinger, and J Andrew Bagnell. Autonomy infused teleoperation with application to brain computer interface controlled manipulation. *Autonomous Robots*, pages 1–22, 2017.

[16] Anusha Nagabandi, Gregory Kahn, Ronald S Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. *arXiv preprint arXiv:1708.02596*, 2017.

[17] Authors Omitted. Disambiguation of human intent through control space selection. *Autonomous Robots*, Under Review.

[18] Patrick M Pilarski, Michael R Dawson, Thomas Degris, Jason P Carey, and Richard S Sutton. Dynamic switching and real-time machine learning for improved human control of assistive biomedical robots. In *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on*, pages 296–302. IEEE, 2012.

[19] Dorsa Sadigh, S Shankar Sastry, Sanjit A Seshia, and Anca Dragan. Information gathering actions over human internal state. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 66–73. IEEE, 2016.

[20] Dorsa Sadigh, Shankar Sastry, Sanjit A Seshia, and Anca D Dragan. Planning for autonomous cars that leverage effects on human actions. In *Robotics: Science and Systems*, 2016.

[21] Gregor Schöner. Dynamical systems approaches to cognition. *Cambridge Handbook of Computational Cognitive Modeling*, pages 101–126, 2008.

[22] Tyler Simpson, Colin Broughton, Michel JA Gauthier, and Arthur Prochazka. Tooth-click control of a hands-free computer interface. *IEEE Transactions on Biomedical Engineering*, 55(8):2050–2056, 2008.

[23] Tarek Taha, Jaime Valls Miró, and Gamini Dissanayake. A POMDP framework for modelling human interaction with assistive robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 544–549. IEEE, 2011.

[24] Michael Tomasello and Malinda Carpenter. Shared intentionality. *Developmental Science*, 10(1):121–125, 2007.

[25] Simon Tong and Daphne Koller. Active learning for parameter estimation in Bayesian networks. In *Advances in neural information processing systems*, pages 647–653, 2001.

[26] Zhikun Wang, Katharina Mülling, Marc Peter Deisenroth, Heni Ben Amor, David Vogt, Bernhard Schölkopf, and Jan Peters. Probabilistic movement modeling for intention inference in human–robot interaction. *The International Journal of Robotics Research*, 32(7):841–858, 2013.