TOPICS
TOPICS IN COGNITIVE SCIENCE

Topics in Cognitive Science 0 (2022) 1–33

ISSN: 1756-8765 online
DOI: 10.1111/tops.12633

This article is part of the topic "Building the Socio-Cognitive Architecture of COHUMAIN: Collective Human-Machine Intelligence," Cleotilde Gonzalez, Henny Admoni, Anita W. Woolley and Scott Brown (Topic Editors).

The Role of Adaptation in Collective Human–AI Teaming

Michelle Zhao, Reid Simmons, Henny Admoni

Robotics Institute, Carnegie Mellon University

Received 30 June 2022; received in revised form 17 October 2022; accepted 19 October 2022

Abstract

This paper explores a framework for defining artificial intelligence (AI) that adapts to individuals within a group, and discusses the technical challenges for collaborative AI systems that must work with different human partners. Collaborative AI is not one-size-fits-all, and thus AI systems must tune their output based on each human partner's needs and abilities. For example, when communicating with a partner, an AI should consider how prepared their partner is to receive and correctly interpret the information they are receiving. Forgoing such individual considerations may adversely impact the partner's mental state and proficiency. On the other hand, successfully adapting to each person's (or team member's) behavior and abilities can yield performance benefits for the human–AI team. Under this framework, an AI teammate adapts to human partners by first learning components of the human's decision-making process and then updating its own behaviors to positively influence the ongoing collaboration. This paper explains the role of this AI adaptation formalism in dyadic human–AI interactions and examines its application through a case study in a simulated navigation domain.

Keywords: Human–AI teaming; Human robot interaction; Adaptation; Human–robot collaboration

Correspondence should be sent to Michelle Zhao, Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, Pennsylvania, 15213, USA. E-mail: mzhao2@andrew.cmu.edu

1. Introduction

Robots and artificial intelligence (AI) increasingly serve as collaborative partners to humans. With rapid advancements in machine learning, AI and robot partners have become better equipped to collaborate with humans on joint tasks, such as robot-assisted care (Mertens et al., 2011) and cooking (Koppula, Jain, & Saxena, 2016; Torrey, Fussell, & Kiesler, 2013). Human–AI teaming focuses on interactions between human and AI teammates where a team must jointly accomplish collaborative tasks with shared goals. In a joint activity, members of the team participate to achieve certain dominant goals (Walker, 1997). The actions available to each team member may not be identical. For example, a coach providing guidance to players does not play the game, but is working with the players toward a joint goal of winning the match. Teams comprising of humans and AI agents may also have asymmetric abilities while performing a joint activity.

Human–AI collaboration evolves dynamically as the team progresses through a task (Chilton, Hardgrave, & Armstrong, 2010; Sonnentag & Frese, 2012). The required activities of each member may vary as the team progresses through different stages of the interaction. For example, in a cooking collaboration, team members may first prepare ingredients, then transition to cooking dishes, and later clean up at the end of the interaction. Shared knowledge about a team's history may emerge in collaborations that occur over extended periods of time or repeated interactions (Clark & Wilkes-Gibbs, 1986; Walker, 1997). Learning from shared history may also change the way in which team members interact by way of forming conventions (Hawkins, Frank, & Goodman, 2017; Khani, Goodman, & Liang, 2018). For example, in a task-oriented collaboration, role specialization may evolve as a type of convention, as the team collectively realizes that some members are best suited for specific roles based on their skills (Aggarwal, Woolley, Chabris, & Malone, 2019; Shih, Sawhney, Kondic, Ermon, & Sadigh, 2021). In other cases, AI designed for teaming may not consistently collaborate with the same human partners. Instead, AI or robot agents may need to collaborate ad-hoc with new partners on-the-fly and reason about how their behavior meshes with the behavior of different human teammates in order to promote fluent coordination and effective teaming (Barrett, Rosenfeld, Kraus, & Stone, 2017; Dafoe et al., 2020). In this paper, we will consider collaborative interactions between humans and autonomous agents serving as team members. The agents, having autonomous decision-making processes, may be robots or AI agents, depending on the context and nature of the interaction. As such, we will refer in this article to the autonomous partner as a robot or AI interchangeably.

Human partners may also differ in their preferences (Gombolay, Huang, & Shah, 2015), capabilities (Dunnette, 2014), knowledge (Motowildo, Borman, & Schmit, 1997), intentions (Nicholls, Cheung, Lauer, & Patashnick, 1989), or preferred strategies for accomplishing shared goals (Gobet, Richman, Staszewski, & Simon, 1997). As team members interact, individual differences in preferences, tendencies, and capabilities manifest in behavioral differences (Burke, Stagl, Salas, Pierce, & Kendall, 2006). These behavioral differences require other teammates to adapt their actions in order to better coordinate their behaviors, strategies, or roles. These features can also change over time during a long-term interaction (Koay, Syrdal, Walters, & Dautenhahn, 2007), so adapting the agent's behavior based on a developed and dynamic understanding of the human partner is key for improving collaboration.

In literature on the psychology of human teams, adaptation is defined as a change in team structures or goal-directed actions in response to a salient cue, in order to achieve a functional performance outcome (Burke et al., 2006). In teams with humans and machines, teaming is similarly defined as a context-aware dynamic arrangement of human and AI team members aimed toward efficient pursuit of shared goals (Madni & Madni, 2018). Hence, adaptation in human teams (Burke et al., 2006) and human–AI teams (Madni & Madni, 2018) both rely on cues or cue streams to inform some change within the team structures. In the context of AI adaptation to human partners, the cue for adaptation consists of salient features related to the human. Adaptation involves identifying such features, or cues, that signal an important need for change and responding with a plan that continues to pursue a desired outcome. Collaborative AI and robots will not work in isolation, but will need to be proficient at teaming with different and new human partners, requiring the ability to adapt online to salient features observed in human partners. Reducing the scope of our definition to robot(/AI)-to-human adaptation, we define robot adaptation to human partners in human–robot teaming as an interaction in which a robot changes its behavior, based on understanding of human teammates learned from observation of salient cues not observed a priori to the interaction.

When designing interactions that facilitate robot adaptation to human partners, two assumptions underlie the adaptive human–robot interaction design. First, adapting robot behavior to human partners assumes that humans differ along particular features. We refer to an *adaptation parameter* (AP) as the set of salient and relevant human features that the robot learns and uses to guide its actions. The second assumption is that learning the adaptation parameter will allow the robot to change its behavior in a way that will positively influence some aspect of the collaboration, which may be measured by task performance, collaborative fluency (CF) (Hoffman, 2019), team trust (Hancock et al., 2011; Kwon, Jung, & Knepper, 2016; Lewis, Sycara, & Walker, 2018), engagement (Sidner & Lee, 2003), or another evaluation measure.

The goal of this paper is to present a perspective for formalizing agent adaptation that can be used when designing adaptation in different contexts. We decompose the process of adaptation into two steps: (1) developing understanding of relevant characteristics of human teammates, and (2) adapting agent behavior based on the learned features. The objectives of the human–robot collaboration influence the features of human partners that will be particularly relevant for adaptation. For example, the adaptation parameter learned by the robot can be cognitive factors internal to the human teammate, such as fatigue (Peternel, Tsagarakis, Caldwell, & Ajoudani, 2018) and intentions (Nguyen et al., 2011), but can also be a model of human behavior, such as a learned policy (Barrett et al., 2017). Human decision-making processes have been represented in prior work as partially observable Markov decision processes (POMDP) (Boutilier, Dean, & Hanks, 1999; Hauskrecht, 2000; Kaelbling, Littman, & Cassandra, 1998; Monahan, 1982). While representing humans as POMDP decision makers is common in the literature (Brown & Tellex, 2010; Lam & Sastry, 2014a; Rosenthal & Veloso, 2011a; Woodward & Wood, 2012a), we additionally denote the adaptation cue as a component of the robot's estimate of the human's Markov decision process (MDP), which allows the framework generalize to different adaptation parameters. This formulation interprets robot adaptation as learning and influencing a component of the human's decision process.

2. Related work

In human–AI teaming, the human and AI partners must collaborate on joint tasks and coordinate their actions to achieve a set of shared goals. This collaboration dynamically evolves over the course of an interaction, as the required tasks, collaboration patterns, and team composition may change over time. People may use information they learn about the task, their partner, and the external environment in order to update how they approach the task. AI teammates must learn to adapt to different behaviors and different team members in order to coordinate well (Ahmad, Mubin, & Orlando, 2017). Key to adaptation is understanding people's behavior through observation during an interaction (Nikolaidis, Nath, Procaccia, & Srinivasa, 2017; Patel et al., 2015) or over the course of repeated interactions (Shih et al., 2021). In this section, we examine human–robot or human–AI interactions in which an autonomous agent adapts to some salient feature of the human's behavior. Selection of such a feature implies that the feature to which the agent adapts is relevant to the objective of the interaction. Depending on the context of the interaction, adaptation parameters may be human intentions, physical factors, cognitive features, or learned models of human behavior. While not exhaustive, each paragraph below focuses on one of these categories of adaptation parameter. We examine these different adaptation parameters prior to examining how our formalism (Section 3) can be applied generally across interactions with different adaptation features (Section 3.3).

Adaptation to goals and intentions. AI partners designed for teaming must learn from interactions with human teammates to develop a better understanding of the team members. One way in which agents can understand the behavior of human partners is by inferring their goals. MDPs are a common framework for goal recognition (Sutton & Barto, 2018). Intent inference has been posed as a problem of inverse reinforcement learning (Ng & Russell, 2000) or inverse optimal control (Ziebart et al., 2009). Bayesian approaches are common in goal inference (Baker, Tenenbaum, & Saxe, 2006; Baker, Saxe, & Tenenbaum, 2009; Monfort, Liu, & Ziebart, 2015). Nguyen et al. (2011) introduce a collaborative framework in which an AI serves as an NPC (nonplayer character) in an online game. The AI infers the intention of the human partner, specifically which subtask the human partner is aiming to complete, and assists on a matching subtask. Understanding the intended goals can then inform how a robot partner should update its behavior (Koert et al., 2019; Levine & Williams, 2018). Khoramshahi and Billard (2019) develop an adaptive control framework that allows a robot to learn the intended task a human partner is trying to accomplish by maintaining a belief distribution over each task, and updating its beliefs based on observed human actions. The robot adapts by switching tasks to the one intended by the human partner. Huang, Cakmak, and Mutlu (2015) adapt robot handover strategy in a physical task based on high-level objectives (reaching, placing, and idling) inferred from human actions. The human partner's objectives and intentions can serve as a cue based on which the robot adapts its collaborative behavior.

Adaptation to internal cognitive features. In addition to inferring goals, learning cognitive or internal features of human partners facilitates robot-to-human adaptation. Nikolaidis, Hsu, and Srinivasa (2017) develops a computational formalism for mutual adaptation between a robot and human partner in a collaborative task, using a model that learns how adaptable a human is to their partner. The robot utilizes the measure of human adaptiveness to determine how to best collaborate and whether to guide the human partner toward a better strategy. In a user study, participants were more likely to adapt when working with the mutually adaptive robot, over a baseline cross-trained robot (Nikolaidis & Shah, 2013). Another cognitive feature that influences behavior in a collaboration is the human's awareness of and aversion to risk. In Kwon et al. (2020), humans are modeled as being risk-aware, noting that people may variably perceive differences between positive and negative rewards, and differ in how averse they are to receiving negative rewards. In Görür, Rosman, Sivrikaya, and Albayrak (2018), collaborative agents anticipate and adapt to the state of mind of human partners in two stages. In the first stage, the agent anticipates the human's availability and capability of performing the task, in addition to the intended task itself. In the second stage, the robot selects actions to take, deciding whether it should assist the human or intervene in other ways. Hawkins, Kwon, Sadigh, and Goodman (2020) learns and adapts to human speaking conventions in a repeated conversational task. The approach leverages the observation that humans form new linguistic conventions to more effectively communicate with partners in repeated interactions. In an online task, Rookhuiszen, Obbink, and Theune (2009) infers human experience from click usage, in order to dynamically switch between complexity levels of instruction. Our case study investigates a similar concept of adapting the levels of complexity in task instructions (Section 5).

Adaptation to physical factors. In physical human–robot collaborations, human physical factors may inform robot adaptation. For example, fatigue is an internal factor which influences one's ability to exert effort in a task. Peternel et al. (2018) adapts a robot's assistance online to human motor fatigue. The human muscle activity is measured through electromyography, and when fatigue reaches a predefined threshold, the robot adjusts its stiffness of motion to take over completion of the task, allowing the human partner to exert less physical effort. Zhang, Chen, Zhang, & Jia (2020) considers a human–robot physical collaboration in a manufacturing assembly scheduling problem. Using the key insight that the capability of robots is usually stable, while human capability usually varies, the system periodically observes how well the human partner is performing, measured on performance indices, and estimates the human's capability. The human's capability affects their ability to perform the actions needed for the task, and a scheduler responds accordingly by increasing the tasks assigned to the robot teammate when necessary. Li et al. (2015a) develops a robot adaptation framework that adapts to a human's leading or following actions, where the amount of force exerted by a human partner indicates their intention to lead or follow. When a human persistently takes leadership actions by exerting strong force, the robot yields control and becomes the follower. Nemec, Likar, Gams, and Ude (2018) adapts robot control in a physical human–robot interaction based on measurements of the speed and variance of human demonstrations. In a similar collaborative shared control task, Li et al. (2015b); Li, Yang, and He (2016) updates robot control based on the magnitude of forces exerted by the human partner. The human partner's physical constraints and capabilities are examples of physical adaptation parameters used in the context of human–robot collaborations.

Adaptation to learned human models. Instead of internal features, a learned model of human behavior can also serve as an adaptation cue. Barrett and Stone (2015) performs adaptation to new teammates by reusing knowledge learned from interactions with past teammates. By learning a policy for working with each team in a training set, the agent identifies the most similar player type to the new partner and acts according to the complementary policy. Li et al. (2021) applied a best-response approach to selecting policies from a library of response policies that best match a particular player type. The agent identifies the policy that most closely matches the actions taken by the human as observed by the agent. The agent adapts by acting according to the complementary policy. Prior work (Barrett et al., 2017) proposes a general purpose algorithm that learns policies from previous human teammates, and uses the learned models to quickly adapt to new teammates. Zanchettin, Casalino, Piroddi, and Rocco (2018) predicts the patterns of human activity in order infer when a human will be performing a particular activity and will request a specific collaborative operation with a robot. Predicting the activity and activity's duration allows the robot to adapt its own actions in order to improve the assistance it provides to the human partner.

3. POMDP model of human partners

In this section, we describe a framework for learning salient human features for adaptation. The collaboration's objective and context influence the salient cues related to the human partner learned by the robot for adaptation. As outlined in Section 2, the learned adaptation parameter may be a model of human behavior (i.e., policy), but may also be cognitive or physical features (i.e., adaptability and fatigue). Adapting to some estimated human feature(s) implicitly assumes that the chosen feature(s) affects the human's behavior in the collaboration and that collaboration may be positively influenced by changing robot behavior in response to learning the feature. Hence, we formalize AI learning of particular human features by reducing it to estimation of a component of the human's decision process. If the human feature does not affect their behavior in the collaboration in any way, learning the feature has no bearing on how the robot should change its behavior to benefit the collaboration. We aim to develop a formalism for robot-to-human adaptation that is agnostic to the different types of human adaptation parameter. Our focus is on human–robot dyads, with generalization to many-to-one human–robot teams.

3.1. Human decision process

Human–robot teams operate together to perform a joint activity (Walker, 1997), in which members of the team participate to achieve certain dominant goals. Team members perform joint actions over some period of time, and the actions available to each team member may not be symmetric. The human partner in a human–AI team is also operating under a decision process independent from the robot. Although actions are not decided upon jointly, they are performed jointly. Thus, the robot's actions do not influence the human's decision process directly; the robot's actions influence the state of the world, which subsequently influences the human's decision process. A flexible way of modeling human and robot independent decision-making is with a POMDP (Boutilier et al., 1999; Hauskrecht, 2000; Kaelbling et al., 1998; Monahan, 1982). The POMDP models decision-making for a single-agent acting in an environment. The agent can perform actions that affect the state of the environment, and has the goal of maximizing the expected future rewards that depend on the environment states resulting from its own future actions. The agent cannot observe the true underlying state of the environment, but can make observations that depend on the state (i.e., sensor readings). The agent uses its observations to establish a belief about the true state of the environment that it is in. This belief is based on the history of observations and is expressed as a probability distribution over states. Determining how to best act in a POMDP is learning a policy that prescribes how an agent should act to maximize its expects rewards under the beliefs about the environment. Since the human and robot make decisions about their own actions separately, we model the human and robot as each operating under their own POMDP. In order to inform how it should adapt, the robot, which does not know the human's true decision process model, additionally estimates the human's decision process. In the dyadic collaboration, the human decides on actions according to independent process $\mathcal{M}$, the robot decides on actions according to its own decision process $\mathcal{M}^R$, which relies on an estimate $\mathcal{M}^{H_R}$ that it maintains of the human's decision model (Fig. 1).

The human's true decision process is modeled as a single-agent POMDP defined by tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{O}, \Omega, R \rangle$. $\mathcal{M}$ is the human's actual decision process, which is unknown to the robot. $\mathcal{S}$ is the set of states in the environment. The task objective and actions taken by the robot at a given time are encoded in the state representation. The action space of $\mathcal{A}$ is the set of actions available to the human partner. The transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ represents how the human believes the team will transition between states of the world. Since the robot actions too influence how the environment state changes, the function $\mathcal{T}$ encodes how the human predicts how the robot will act. $R : \mathcal{S} \rightarrow \mathbb{R}$ is the human's reward function. Ω is the set of observations. The POMDP model generalizes to environments where the human may not observe the full state. $\mathcal{O}$ is the set of conditional observation probabilities where $\mathcal{O}(o|s_t, a_t)$ is the probability of the agent receiving observation o given state s_t and action a_t. The human maintains a belief state b, where $b(s) = p(s)$ is a probability distribution over environment states and learns a policy $\pi : b \rightarrow \mathcal{A}$ given their true POMDP $\mathcal{M}$. See Table 1, Section *True Human Decision Process* for a notation reference.

[1] Image source https://www.visualpharm.com/free-icons/person-595b40b85ba036ed117da7ec, https://www.creativefabrica.com/product/robot-icon-9/

Table 1
Notation for POMDP model components

Symbol	Description
$\mathcal{M}$	True human decision process (POMDP internal to human, unknown to robot)
$\mathcal{M}^R$	Robot decision process (POMDP internal to robot, unknown to human)
$\mathcal{M}^{H_R}$	Robot's estimate of human decision process (robot's estimate of the POMDP internal to the human)
True Human Decision Process, $\mathcal{M}$	
$\mathcal{S}$	set of states in $\mathcal{M}$
$\mathcal{A}$	set of actions in $\mathcal{M}$
$\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$	transition function in $\mathcal{M}$
Ω	set of observations in $\mathcal{M}$
R	true human reward function in $\mathcal{M}$
$\mathcal{O}$	conditional observation probabilities given state, updates made by the human
b	belief distribution over states, updates made by the human
Robot Decision Process, $\mathcal{M}^R$	
$\mathcal{S}^R$	set of states in $\mathcal{M}^R$
$\mathcal{A}^R$	set of actions in $\mathcal{M}^R$
$\mathcal{T}^R : \mathcal{S}^R \times \mathcal{A}^R \rightarrow \mathcal{S}^R$	transition function in $\mathcal{M}^R$
Ω^R	set of observations in $\mathcal{M}^R$
R^R	internal robot reward function in $\mathcal{M}^R$
$\mathcal{O}^R$	conditional observation probabilities given state, updates made by the robot
b^R	belief distribution over states, updates made by the robot
Robot's Estimate of Human Decision Process, $\mathcal{M}^{H_R}$	
$\mathcal{S}^{H_R}$	inferred set of states available to human in $\mathcal{M}^{H_R}$
$\mathcal{A}^{H_R}$	inferred set of actions available to human in $\mathcal{M}^{H_R}$
$\mathcal{T}^{H_R} : \mathcal{S}^{H_R} \times \mathcal{A}^{H_R} \rightarrow \mathcal{S}^{H_R}$	inferred transition function in $\mathcal{M}^{H_R}$
Ω^{H_R}	inferred set of observations present to human in $\mathcal{M}^{H_R}$
R^{H_R}	inferred human reward function in $\mathcal{M}^{H_R}$
$\mathcal{O}^{H_R}$	inferred conditional observation probabilities of the human, estimated by the robot
b^{H_R}	inferred belief distribution over states of the human, estimated by the robot

Note. This framework for robot-to-human adaptation relies on three primary decision processes, modeled as POMDPs: the human's internal decision process ($\mathcal{M}$), the robot's internal process ($\mathcal{M}^R$), and the robot's estimate of the human's decision process ($\mathcal{M}^{H_R}$). These decision processes may be modeled explicitly or implicitly, in design of the adaptive interaction.

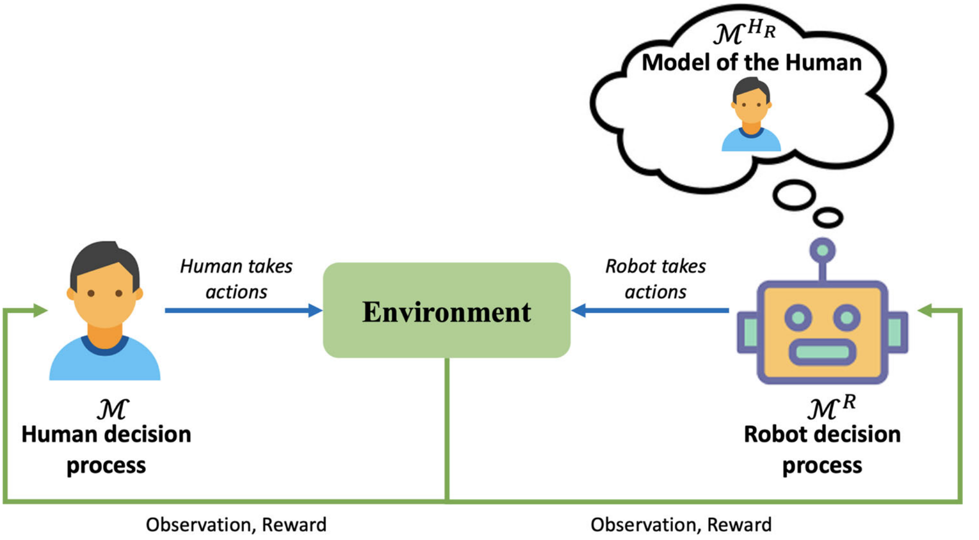

Fig. 1. In a human–robot joint activity, the human and robot select actions based on independent decision processes, $\mathcal{M}$ and $\mathcal{M}^R$, respectively. The human and robot select actions, which affect the environment. In the case where the robot is adapting to some adaptation cue relevant to the human, the robot maintains an estimate of the human's decision process $\mathcal{M}^{H_R}$ to inform its adaptation[1] .

3.2. Robot's estimate of human decision process

$\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{O}, \Omega, R \rangle$ (as defined in Section 3.1) represents the true human POMDP, under which the human partner selects actions according to policy π trained on $\mathcal{M}$. The robot takes actions according to its own POMDP $\mathcal{M}^R$, which may differ and is independent from $\mathcal{M}$. The key observation driving this framework is that learning a human feature for adaptation, implicitly or explicitly, models human behavior from the robot's estimate of the human's decision-making process. Adapting to a selected human feature assumes that the feature influences human behavior in a way relevant to the collaboration, and that adapting robot actions in response can improve the collaboration. The adaptation parameter effectively influences human behavior, and the effects can be predicted with an estimate of the human's decision process. We draw upon insight from human teaming literature that recognizing which particular cues are deemed relevant for adaptation relies on one's long-term memory, contextual priors, and existing mental models (Burke et al., 2006; Endsley, 2017). In the context of robot adaptation to human partners, the selection of a salient adaptation parameter applies a contextual prior provided by the experimenter. For example, in our case study (Section 5), a robot adapts navigation instruction complexity to the inferred knowledge of the human partner. Our implicit assumption is that the human's knowledge will affect their ability to perform the task of following instructions. Our formalism explicitly poses the robot's mental model of the human as an estimate of the human partner's decision process.

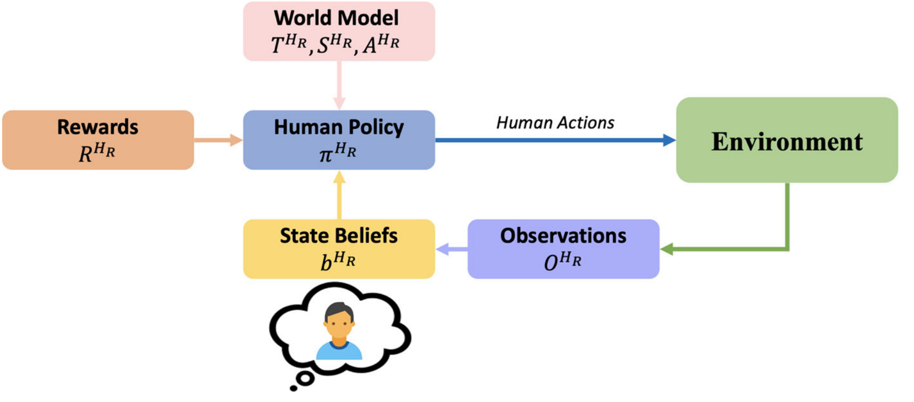

Fig. 2. The robot estimates the human's POMDP model as a single-player POMDP, $\mathcal{M}^{H_R}$. $\mathcal{S}^{H_R}$ is the set of states the robot estimates that the human observes. What the robot does is encoded in states $s \in \mathcal{S}^{H_R}$ from the human's perspective. Under partial observability, the human may not know which state $s \in \mathcal{S}^{H_R}$ they are in, but will observe $o \in \Omega^{H_R}$. Based on the observations, the human forms beliefs over which state they are in, $b^{H_R}(s)$. The action space of $\mathcal{A}^{H_R}$ is the action set the robot believes is available to the human. $\mathcal{T}^{H_R} : \mathcal{S}^{H_R} \times \mathcal{A}^{H_R} \rightarrow \mathcal{S}^{H_R}$ is the estimated transition function, or the robot's estimate of how the human models the world. $R^{H_R} : \mathcal{S}^{H_R} \rightarrow \mathbb{R}^{H_R}$ is the estimated human reward function. The reward function guides the human's policy $\pi^{H_R} : \mathcal{S} \rightarrow \mathcal{A}$ given the full POMDP $\mathcal{M}^{H_R}$[2].

Although the robot does not access the human's true POMDP, the robot estimates the human's POMDP model also as a single-player POMDP $\mathcal{M}^{H_R} = \langle \mathcal{S}^{H_R}, \mathcal{A}^{H_R}, \mathcal{T}^{H_R}, R^{H_R}, \Omega^{H_R}, \mathcal{O}^{H_R}, b^{H_R} \rangle$. The superscript H_R indicates this is the robot's model of the human, which may rely on priors and may not be equivalent to the true decision process of the human. $\mathcal{S}^{H_R}$ is the set of states the robot estimates that the human observes. The robot's behavior is encoded in that set of states. The action space of $\mathcal{A}^{H_R}$ is the action set the robot believes is available to the human. $\mathcal{T}^{H_R} : \mathcal{S}^{H_R} \times \mathcal{A}^{H_R} \rightarrow \mathcal{S}^{H_R}$ is the estimated transition function, or the robot's estimate of how the human models the world. $R^{H_R} : \mathcal{S}^{H_R} \rightarrow \mathbb{R}^{H_R}$ is the estimated human reward function. Ω^{H_R} is the set of observations. $\mathcal{O}^{H_R}$ are the observation probabilities. The estimated human belief state is b^{H_R}, where $b^{H_R}(s) = p(s)$ is a probability distribution over environment states. The human learns a policy $\pi^{H_R} : \mathcal{S} \rightarrow \mathcal{A}$ given the POMDP $\mathcal{M}^{H_R}$. Based on this estimated human POMDP, the robot estimates that the human partner learns and operates under policy π^{H_R}. Importantly, POMDP $\mathcal{M}^{H_R}$ is the robot's model of the human's decision process (Fig. 2). See Table 1, Section *Robot's Estimate of Human Decision Process* for a notation reference.

3.3. *Cognitive parameters as POMDP components*

While representing humans as POMDP decision makers is common in robotics literature (Taha, Miró & Dissanayake, 2011), we additionally observe that learning the

[2] Image source https://www.visualpharm.com/free-icons/person-595b40b85ba036ed117da7ec

adaptation parameter represents learning a component of the estimated human's POMDP $\mathcal{M}^{H_R} = \langle \mathcal{S}^{H_R}, \mathcal{A}^{H_R}, \mathcal{T}^{H_R}, R^{H_R}, \Omega^{H_R}, \mathcal{O}^{H_R}, b^{H_R} \rangle$. In an adaptive interaction, the robot begins with a prior estimate on $\mathcal{M}^{H_R} = \langle \mathcal{S}^{H_R}, \mathcal{A}^{H_R}, \mathcal{T}^{H_R}, R^{H_R}, \Omega^{H_R}, \mathcal{O}^{H_R}, b^{H_R} \rangle$. The adaptation parameter is denoted by α and is an element within the powerset of POMDP components $\alpha \in \mathcal{P}\{\mathcal{S}^{H_R}, \mathcal{A}^{H_R}, \mathcal{T}^{H_R}, R^{H_R}, \Omega^{H_R}, \mathcal{O}^{H_R}, b^{H_R}\}$. Upon obtaining an updated estimate of α', the robot's new estimate of the human becomes $\mathcal{M}^{H_R'} = \alpha' \bigcup \mathcal{M}^{H_R} \setminus \{\alpha\}$. Learning the adaptation parameter improves the estimate of a component of the robot's model of the human POMDP.

We provide several examples of POMDP component representations of adaptation parameters. Examples of adaptation parameters include goals (Baker et al., 2006, 2009; Monfort et al., 2015; Ng & Russell, 2000), intentions (Huang et al., 2015; Koert et al., 2019; Levine & Williams, 2018), or policies (Barrett et al., 2017; Li et al., 2021; Zanchettin et al., 2018). Robot systems also adapt based on cognitive or internal features, such as capabilities (Li et al., 2015a; Peternel et al., 2018; Zhang et al., 2020) or adaptability (Nikolaidis et al., 2017). Learning the goals and intentions of human partners represents learning better estimates of the human's reward function $\alpha = R^{H_R}$. Improving the estimated reward function $R^{H_R'}$ to update $\mathcal{M}^{H_R'}$ provides a better estimate $\pi^{H_R'}$ of the human partner's behavior during the collaboration. The human's world model includes answering questions, such as $\mathcal{T}^{H_R}$: *how does the world work?*, $\mathcal{A}^{H_R}$: *what actions can I take?*, $\mathcal{S}^{H_R}$: *what are the different states of the environment?*. When the robot learns some aspect of the human's world model, they are learning some component(s) in the set $\{\mathcal{S}^{H_R}, \mathcal{A}^{H_R}, \mathcal{T}^{H_R}, \}$. For example, learning the physical fatigue of the human partner in a physical human–robot collaboration can be considered as learning $\alpha = \mathcal{A}^{H_R}$, the action set of the human's POMDP. If a partner cannot exert great force due to fatigue, their ability to exert force is capped, and their action set is essentially reduced to low-force actions. Improving the estimate of $\mathcal{A}^{H_R'}$ better informs the robot of the actions that will be selected by the partner under policy π^{H_R}. If the robot is updating a model of how the human decides actions to take in the current environment, the robot is estimating the human's policy itself; thus, the adaptation parameter in this case would be $\pi^{H_R'}$.

Importantly, modeling of the human's operational POMDP may not be explicit in the algorithms defining robot behavior. For example, Vignolo et al. (2021) develops adaptive behavior in a humanoid robot where the robot slows the speed at which it presents instructions for performing a physical skill when the human asks for clarification. While not explicitly modeled as a POMDP, the robot's implicit intuition is that asking for clarification indicates that the human is uncertain about the required actions and did not understand the instructions at the initial speed. Viewing observations $\mathcal{O}^{H_R}$ as instructions, and states $\mathcal{S}^{H_R}$ as the required actions for the physical skill, the robot aims to learn the belief distribution $b^{H_R'}$: whether the human understands the correct sequence of required actions given the instructions. The human's asking for clarification gives a coarse estimate for $b^{H_R'}$, the human's understanding of the required actions. Our interpretation of this example is one of the various ways the robot's estimate of the human's POMDP can be modeled.

We offer this framework as a way of formalizing robot adaptation to different human parameters. The benefit of thinking about where in the human's decision model the adaptation parameter lies allows designers of an adaptive interaction to reason more explicitly about assumed common knowledge between robot and human partner as priors on the components of the human's POMDP. In Section 5, we examine a case study in which an AI agent adapts to a cognitive feature–human knowledge, and demonstrate how this POMDP model of the human can be applied to formalize the interaction.

4. Adaptive cycle

Adaptation occurs through repeated adaptive cycles, during which adaptive agents assess the salient adaptation parameter related to human partners, and reason about their own behavior accordingly. We simplify the four-component adaptive cycle for human teaming in Burke et al. (2006) for robot-to-human adaptation into a two-process model: (1) a learning process and (2) an influence process. The learning process involves situation assessment. The influence process combines plan formulation and execution. Knowledge from collaboration history informs both processes. Adaptation comprises of assessing a human feature adaptation parameter, whose value for a particular human partner is not known to the robot a priori to the interaction, which in turn allows the robot to respond by changing its behavior in hope of improving the collaboration. The (1) Learning Process represents how the updates to the robot's adaptation parameter estimate are performed. The way in which the robot estimates the set of salient human features (AP) depends on what the features are, requiring the robot to assess the situation for information or observations that will help to approximate the adaptation parameter. Utilizing collaboration history to develop a better partner understanding can contribute to improved estimates of the adaptation parameter. The (2) Influence Process represents how the robot plans and executes actions based on its assessment of the human partner in order to effect a functional outcome. Here, the robot must reason about how its change in response to estimating the adaptation parameter will influence the collaboration and/or the human partner. Collaboration history can inform the robot's predictions of the effects of its actions. These are two algorithmic processes that must be determined when establishing an adaptive interaction. The POMDP human model in particular assists with reasoning about the (2) Influence Process.

4.1. Learning process

The robot's estimate of the human POMDP, $\mathcal{M}^{H_R}$, represents how the robot predicts the human will act in the environment. Given the POMDP components: state belief b^{H_R} update function, rewards R^{H_R}, and environment model $\mathcal{T}^{H_R}, \mathcal{A}^{H_R}, \mathcal{S}^{H_R}$, the robot can predict the human policy π^{H_R}, which determines which actions the human will take. For example, in adaptation to human intentions, where reward $\alpha = R^{H_R'}$ is updated from observation, the robot is able to solve under constant environment model $\mathcal{T}^{H_R}, \mathcal{A}^{H_R}, \mathcal{S}^{H_R}$ and state belief update b^{H_R} an updated prediction of human behavior. Thus, the updated POMDP, $\mathcal{M}^{H_R'}$, consisting of the updated adaptation parameter POMDP component while holding the rest of the components constant, provides a prediction of human behavior through the policy $\pi^{H_R'}$ obtained by solving $\mathcal{M}^{H_R'}$. The learning process is the mechanism by which the adaptation parameter α is learned and updated.

Learning the adaptation parameter entails updating the estimate of a component of the human's POMDP, $\mathcal{M}^{H_R}$. The learning process defines how the robot updates its estimate of the human feature adaptation parameter based on information received or observed during the interaction. Learning the adaptation parameter can be modeled as a regression problem, in which the robot aims to learn a mapping from one random variable, such as environment state, to another random variable: the adaptation parameter estimate. Supervised learning techniques are applied toward learning functions which map relevant state features to the human adaptation parameter estimate, where updates to the estimate based on new observations are learned through gradient descent (Barry & Love, 2021; Kim & Lee, 2020). Bayesian frameworks are another common parameter estimate mechanism, and are especially common in goal inference (Baker et al., 2006, 2009). The Bayesian framework formulates a prior distribution, which incorporates information about the adaptation parameter external to the human partner-specific observations. Bayes theorem obtains a posterior distribution for the parameter given the prior information and observed data relevant to the adaptation parameter from the interaction. Priors on the adaptation parameter can also be leveraged through threshold-based learning of the parameter. For example, (Zhang et al., 2020) updates estimates of human capability by measuring changes in some performance index, where drastic changes that exceed certain thresholds indicate increased or decreased capability. Our case study employs a threshold-based learning process for modeling human task knowledge as the adaption parameter.

4.2. *Influence process*

Once the robot has learned an estimate of the human adaptation parameter, the second component of adaptation is changing the robot's behavior based on the learned feature(s). Designing adaptive robot behavior to individual human partners assumes that changing the robot's behavior based on relevant learned features of the human may improve an aspect of the collaboration. We consider the process by which the robot changes its behavior in order to influence the collaboration based on the estimated adaptation parameter as the influence process. This process involves reasoning about how the robot's actions may affect either the collaboration itself or the human partner directly. This reasoning relies on the robot's objectives when adapting its actions, which can vary from task performance to coordination fluency. During this process, the robot might optionally reason about mutual adaptation (Nikolaidis et al., 2017), where the human might be simultaneously adapting to the robot as well.

Optimizing adaptation for certain goals requires evaluating those objectives and reasoning about how the robot's actions may affect those measures. An adaptation objective, for example, is improving task performance. Task performance is a measure of how efficiently the team completed the task at hand (Burke, Murphy, Riddle, & Fincannon, 2004). In interactions where robots also seek to become more collaborative by adapting, CF metrics measure how effective an agent is at coordinating its behavior with its human partner. CF (Hoffman, 2019) is defined as the coordination of joint activities by members in a team. Adaptation of AI to their human teammates may improve the trust between teammates. In prior analyses of human–robot interactions, trust is most highly influenced by robot performance (Hancock et al., 2011). Human trust in AI may be promoted by confidence in the ability of the AI to meet expectations (Kwon et al., 2016). Engagement (Sidner, Lee, Kidd, Lesh, & Rich, 2005) is another key process that underlies how effectively robots and AI agents can interact with human partners (Holroyd, Rich, Sidner, & Ponsler, 2011; Sidner & Lee, 2003). In our case study (Section 5), one of our outcome measures to evaluate the human–agent interaction is human engagement.

Examples of adaptive behaviors include changing the robot's policy through selection from a policy library (Barrett et al., 2017; Li et al., 2021), or switching tasks based on the inferred goals of a human partner (Huang et al., 2015; Khoramshahi & Billard, 2019). The robot's ability to perform effective adaptation is, however, constrained by the mechanisms by which it can interact with the human partner (Koppol, Admoni, & Simmons, 2021). For example, an agent in a peer-to-peer collaboration, where it can contribute equally to a task, may have a different set of actions available than it would in a coaching collaboration, where the agent may only be able to intervene but not perform the task itself. Actions performed by the robot affect the world states $s \in \mathcal{S}$ experienced by the human. These states may also subsequently affect other components of the human's true POMDP $\mathcal{M}$. The influence process is the mechanism by which the robot selects its actions based on learned understanding of the human partner $\mathcal{M}^{H_R}$, which we can represent as some function $f_{\mathcal{M}^R, \mathcal{M}^{H_R}, \alpha} : \mathcal{S}^R \rightarrow \mathcal{A}^R$ parameterized by the robot's decision process $\mathcal{M}^R$, its estimate of the human $\mathcal{M}^{H_R}$, and the learned adaptation parameter, mapping robot states $\mathcal{S}^R$ to suitable robot actions in $\mathcal{A}^R$.

5. Case study: Adapting navigation instructions to human knowledge

We present a case study in which an AI teammate adapts to its human partner via a learning an adaptation parameter: human knowledge of task-relevant information. The agent adapts by changing its presentation of navigation instructions to the human partner. The collaboration is one in which the AI partner serves as a coach to the human partner on a navigation task, situated in a search-and-rescue (SAR) environment. When guiding someone to perform a task, humans naturally simplify their instructional language if their partner appears confused or fails to understand the information. For instance, teachers systematically move instructional conversation up and down the hierarchy of cognitive difficulty, based on the proficiency of student responses (Kawanaka & Stigler, 1999; Nathan & Kim, 2009). It follows that effective instruction should be suited to a person's preparedness and expertise.

A hierarchy of instruction complexity naturally emerges as one repeatedly simplifies task instructions (Fig. 3). Specifically, complexity is defined by abstraction level. We situate this work in the context of an AI-guided SAR task, which serves as a fitting task domain. An AI guide, equipped with greater familiarity of the environment, would provide instructions to help the human rescuer navigate to all victims as quickly as possible. Importantly, navigation instructions have a natural decomposition into varying levels (modes) of complexity. High complexity directions, like "Go to Office A," are presented at a high level of abstraction and require significant prerequisite knowledge in order to be able to comprehend and follow. Inversely, low complexity instructions, such as "Turn right. Walk 4 steps forward.," require

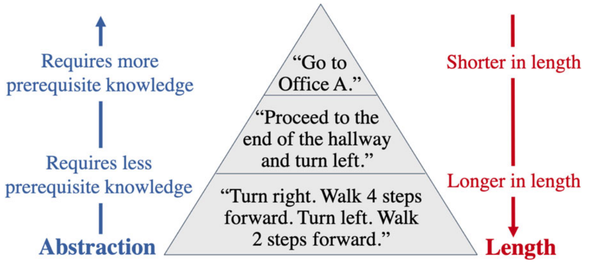

Fig. 3. The abstraction hierarchy of instruction complexity represents a tradeoff between abstraction and length. Directions using greater abstraction are more complex in that they require significant prerequisite knowledge in order to be able to comprehend and follow. For example, the receiver of the instruction "Go to Office A" must know where Office A is located spatially. Inversely, low complexity instructions, such as "Turn right. Walk 4 steps forward.," require little to no prerequisite knowledge, but are more verbose.

little to no prerequisite knowledge, but are more verbose. Thus, there may exist a tradeoff between instruction complexity and length, and providing someone unsuitable instructions may incur a cost in length. In order to measure suitability of different modes of instruction complexity, we introduce the concept of *alignment* between instruction complexity and human knowledge. Alignment occurs when people have sufficient prerequisite knowledge to follow instructions at the provided level of complexity. Misalignment occurs when people are provided instructions that are too complex or too simple for their expertise. In this case study, we answer two cardinal questions: *For navigation instruction-following tasks, should the complexity of AI assistance be adapted to the knowledge of humans? If so, how can we operationalize this type of adaptation?*

5.1. *Preliminaries*

5.1.1. *Task scenario*

An AI-instructor must guide a single human rescuer to victims in a damaged office building (2D gridworld environment). There are 20 injured victims inside of the building who need to be found and rescued. Of these, seven victims are severely injured (denoted in red), and will expire if not treated in time. The rest are moderately injured (denoted in blue), and will persist the duration of the 4-min game. The AI guide will provide onscreen navigation instructions throughout the game to help the player navigate to all victims in time. The task of the human player is to follow the coach's instructions to save all victims (Fig. 4).

5.1.2. *Instruction complexity levels*

Instruction complexity is stratified into three complexity levels: Level 1 ($\mathcal{C}_1$): low-abstraction, Level 2 ($\mathcal{C}_2$): medium-abstraction, and Level 3 ($\mathcal{C}_3$): high-abstraction. The AI instructor controls a single-parameter model of the rescuer. The rescuer is parameterized by β, which represents the complexity level of text-based navigation instructions at which the coach will interact with the rescuer. Each complexity level $\mathcal{C}_i$ is defined by a vocabulary corpus V_i, where each corpus is comprised of a set of actions and objects, $V_i = (A_i, O_i)$. An

Fig. 4. Participants are provided a 2D gridworld view of search and rescue task, with several onscreen tools. The seven severely injured victims are denoted by red circles, and the noncritical victims are in blue. The game screen includes a mini-map in the corner for navigation ease. The instructions from the navigation assistant are displayed at the top of the screen in yellow.

instruction is an (action, object) tuple. `ROOMS` is the set of room names in the building map. `VICTIMS` is the set of victims in the building map. Each instruction level can be generated from a transformation of a neighboring level that either applies higher or lower complexity.

$$A_3 = \{\text{Goto,Triage}\}$$

$$O_3 = \texttt{ROOMS} \cup \{\text{RedVictim,BlueVictim}\}$$

$$A_2 = \{\text{Turn,Proceed,Enter,Exit,ApproachandSave}\}$$

$$O_2 = \mathbb{N} \times \{\text{Room,Hallway,Intersection,left,right,redvictim}$$

$$A_1 = \{\text{Walkforward,Turn,Stop,Utilize}\}$$

$$O_1 = \mathbb{N} \times \{\text{Steps,Left,Right,MedicalEquipment]}\}$$

5.1.3. POMDP estimate formulation

The coaching agent estimates the human's POMDP model also as a single-player POMDP $\mathcal{M}^{H_R} = \langle \mathcal{S}^{H_R}, \mathcal{A}^{H_R}, \mathcal{T}^{H_R}, R^{H_R}, \Omega^{H_R}, \mathcal{O}^{H_R}, b^{H_R} \rangle$. $\mathcal{S}^{H_R}$ is the set of states observed by the human rescuer. The human's state information at time t is combination of the building environment state, denoted s_t, and the content and complexity level of the instructions at t, denoted s_t^i for instruction state. The map state s_t includes the position of the human rescuer $\{x_t^p, y_t^p\}$, the (x, y) location and state $z \in \{\text{triaged,nottriaged}\}$ of all victims in `VICTIMS`, the (x, y) centroid locations of all rooms in `ROOMS`. The instruction state is the content and complexity level of instructions on screen given the state of the building environment $\mathcal{C}_t(s_t)$. $\mathcal{C}_t$ is the instruction complexity level at time t, selected from $\{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\}$. $\mathcal{C}_t(s_t)$ is the content of instructions when applying level $\mathcal{C}_t$ to the current environment state at t, s_t. Combining these, states in the robot's estimated human POMDP $\mathcal{M}^{H_R}$ are $s \in \mathcal{S}^{H_R} = s_t \bigcup \mathcal{C}_t(s_t) = \{x_p, y_p\} \bigcup \{x_i, y_i, z_i\}_{\forall i \in \texttt{VICTIMS}} \bigcup \{x_j, y_j\}_{\forall j \in \texttt{ROOMS}} \bigcup \mathcal{C}_t(s_t)$. Since the robot's action space is the set of three levels of instruction complexity, the robot's behavior is encoded in human's POMDP state.

Ω^{H_R} is the set of observations and $\mathcal{O}^{H_R}$ are the observation probabilities. $\mathcal{O}^{H_R}(o|s_t, a_t)$ is the probability of the agent receiving observation o given state s_t and action a_t, which is a deterministic function of the state s_{t+1}. The observation given s_t, a_t is the radius-5 field-of-view (FOV) the human has at their current position in s_{t+1} within the building. The observation given map state s_t is denoted $o_t = FOV(s_t)$ and consists of only the parts of the environment visible in the human's FOV combined with the onscreen instructions which are always visible $o_t = FOV(s_t) \bigcup \mathcal{C}_t(s_t) = \{x_p, y_p\} \bigcup \{x_i, y_i, z_i\}_{\forall i \in FOV(\texttt{VICTIMS})} \bigcup \{x_j, y_j\}_{\forall j \in FOV(\texttt{ROOMS})} \bigcup \mathcal{C}_t(s_t)$. The action space of $\mathcal{A}^{H_R}$ is the action set available to the human, which in this navigation setting is {move North, move South, move East, move West, triage}. $R^{H_R} : \mathcal{S}^{H_R} \rightarrow \mathbb{R}^{H_R}$ is the estimated human reward function. Under the assumption of compliance with the agent's onscreen navigation instructions, the reward function is assumed to be positive reward for taking actions following the directions given by $\mathcal{C}_t(s_t)$.

The human's state belief, $b^{H_R}(s)$, represents where the human believes all rooms and victims are located in the full building given their partial observations. Formally, $b^{H_R}(s)$ is the probability that the human believes they are in state $s \in \mathcal{S}^{H_R}$. The belief update is $b'(s) = \sum_{s' \in \mathcal{S}^{H_R}} \mathcal{T}^{H_R}(s|a, s') b^{H_R}(s')$. Put simply, the human's state belief is where they think rooms and other objects in the environment are located based on their FOV-restricted observations. Under $\mathcal{M}^{H_R}$, the human is only being rewarded for following the directions of the robot instructor. Thus, the robot estimates that the human partner's policy π^{H_R} is to simply follow the instructions of $\mathcal{C}_t(s_t)$ given their state belief b^{H_R}. $\pi^{H_R}(\mathcal{C}_t(s_t), b^{H_R}) \rightarrow a_t \in \mathcal{A}^{H_R}$. The robot utilizes priors on abstraction complexity and the amount of knowledge needed for understanding instructions at different levels of abstraction. Its prior is that as instruction complexity decreases, the reliance of the human policy on map knowledge decreases. For low complexity instructions at $\mathcal{C}_1$, the action required to follow instructions is the instruction itself, and does not rely on beliefs about locations of rooms in the environment. This means $\pi^{H_R}(\mathcal{C}_1(s_t), b^{H_R}(\cdot)) = \pi^{H_R}(\mathcal{C}_1(s_t))$.

5.2. *Adaptation framework*

In this section, we present a two-step, EVALUATE-and-ADAPT interaction framework for instruction-generation based on dynamic, automatic complexity mode adaptation (Fig. 5). The human rescuer's sole objective is to follow the AI guide's navigation instructions, which ensures human compliance to the provided instructions. First, at time $t = 0$, the AI guide gives the rescuer an instruction at a default complexity mode, β_0, to direct the rescuer to the first victim.

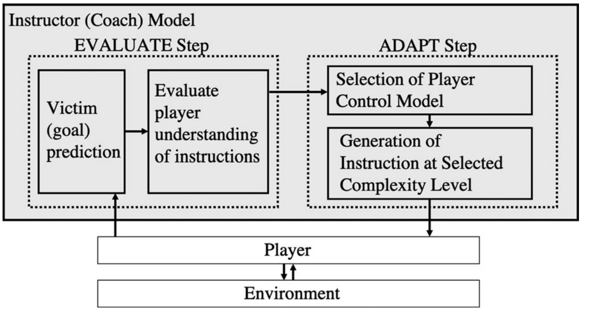

Fig. 5. The AI evaluates understanding, and adapts, generating the next instruction at the appropriate level. In the EVALUATE step, the instructor learns the adaptation parameter (knowledge of the building map) by observing the actions of the rescuer and predicts whether they are headed to the correct victim. In the ADAPT step, the AI dynamically changes its actions based on the adaptation parameter, presenting instructions at a new complexity level suitable for the knowledge of the human partner.

5.2.1. *Learning process*

In the EVALUATE step, the instructor learns the adaptation parameter: knowledge of the building map. The instructor observes the actions of the rescuer and predicts whether they are headed to the correct victim (goal). Assuming the human complies with all instructions, goal prediction serves as a proxy for comprehension at the current complexity level β_0, because failing to reach a goal must thus be attributed to inability. The instructor computes the probability $\mathcal{P}(g^*|\xi)$ that the rescuer is headed to the intended victim g^* given their current trajectory ξ. The set of goals is the set of victims $G = \{g_1, \ldots, g_{20}\}$. We aim to compute $\mathcal{P}(g^*|\xi_{0:t})$: the probability that the rescuer is headed to the instructed victim g^* given their current trajectory $\xi_{0:t} = \{x_1, \ldots, x_t\}$. x_t is the position at time t:

$$\mathcal{P}(g^*|\xi_{0:t}) = \mathcal{P}(g^*|x_0, \ldots, x_t) = \frac{\mathcal{P}(x_0, \ldots, x_t|g^*)\mathcal{P}(g^*)}{\mathcal{P}(x_0, \ldots, x_t)}. \tag{1}$$

Under the POMDP Human Model Framework, the adaptation parameter in this interaction is the human's knowledge of the building environment and understanding of navigation instructions at the current complexity level. The probability that the human rescuer is headed toward the intended goal serves as a proxy for their comprehension. Thus, the component of the human's POMDP α the robot aims to learn is b^{H_R}. The set of possible states in the robot's estimate of the human POMDP is $s \in \mathcal{S}^{H_R} = \{x_p, y_p\} \bigcup \{x_i, y_i, z_i\}_{\forall i \in \texttt{VICTIMS}} \bigcup \{x_j, y_j\}_{\forall j \in \texttt{ROOMS}} \bigcup \mathcal{C}_t(s_t^e)$. which is the space of all possible combinations of player position, victim positions, room positions, and instructions at the three complexity levels. At state s_t, if $b^{H_R}(s_t)$ is small and close to 0, then the human does not have the correct belief of the room and victim locations, and thus cannot comprehend the instructions. If $\mathcal{P}(g^*|\xi) \approx 0$, then $b^{H_R}(s_t)$ is small. If $\mathcal{P}(g^*|\xi) \approx 1$, then $b^{H_R}(s_t) \approx 1$. By using goal prediction to proxy knowledge of the environment, the agent makes a crude estimate of b^{H_R}.

5.2.2. *Influence process*

In the ADAPT step, the AI dynamically changes its actions, presenting instructions at a new complexity level suitable for the knowledge of the human partner. The AI reasons that high complexity instructions can only be understood if the human knowledge of the room names is sufficient. The AI reasons that instructions at complexity levels requiring map knowledge exceeding the knowledge of the human partner will negatively influence the human's ability to perform the task of following the navigation directions. Thus, the instructor makes a threshold-based decision on whether or not the level of instruction must change in order to better fit the human rescuer's comprehension and knowledge. Low values of $\mathcal{P}(g^*|\xi)$ indicate the rescuer is unable to understand directions at the current complexity, and requires simpler instructions in the next interaction. High $\mathcal{P}(g^*|\xi)$ values indicate the rescuer understands instruction at the current complexity mode and is headed to the correct victim. We define two hyperparameters: a threshold τ_G for complexity mode-increase and a threshold τ_B for mode-decrease. If $\mathcal{P}(g^*|\xi_{0:t}) > \tau_G$, the instructor increments the level of instruction complexity. If $\mathcal{P}(g^*|\xi_{0:t}) < \tau_B$, then decrement the instruction mode. Otherwise, the rescuer level remains as is (Fig. 4). Algorithm 1 defines the interaction loop algorithm for adaptive navigation assistance (ANA). Based on sample trials, the thresholds are tuned to $\tau_B = 0.3$, $\tau_G = 0.8$.

This EVALUATE-and-ADAPT process occurs for as many interactions as necessary to guide the rescuer throughout the episode. In this interaction, the agent does not aim to solve the human's estimated POMDP and cannot actually change b^{H_R}: the human's knowledge of the environment. The agent changes its instruction complexity, which is included in the human's observations $\mathcal{O}^{H_R}$, such that the human's belief of the environment layout will not affect their ability to perform the task of following instructions. For example, the agent provides low complexity instructions at $\mathcal{C}_1$ when the human's comprehension $\mathcal{P}(g^*|\xi)$ is low, making their state belief incorrect ($b^{H_R}(s_t)$ is small). The human does not have a correct understanding of what the state of the environment is. Now, when given $\mathcal{C}_1$ level instructions, the human policy does not rely on b^{H_R} since $\pi^{H_R}(\mathcal{C}_1(s_t^e), b^{H_R}(\cdot)) \approx \pi^{H_R}(\mathcal{C}_1(s_t^e))$, allowing the human to follow instructions to reach the victim without having complete knowledge of the environment.

5.3. *User study design*

We ran a mixed-design user study to measure how prerequisite knowledge and information complexity affect human instruction-following and how adaptivity (ANA) affects participant performance and mental state. We investigated three dependent effects: (1) task performance, (2) mental state, and (3) preference. Task performance was measured by the number of victims rescued by the team. Mental state conveys a person's sentiments toward their experience and sense of self-proficiency in the task at hand. Given that the instructions of the task were to follow the onscreen guidance in order to save victims as quickly as possible, the number of victims is a measure of how quickly the participant was able to perform the task of following the navigation instructions. If a participant did not manage to follow instructions accurately, they would be instructed to correct their missteps, as doing so is necessary in order to reach the intended next victim. Our measures of task-oriented mental state were engagement, Algorithm 1 Adaptive Navigation Assistance (ANA)

$\xi_{0:t}, g_t^*, C_{g_t^*}, \mathcal{C}_t \leftarrow$ **Input**: trajectory, current goal, floodfill values of current goal, current level
$\tau_B, \tau_G \leftarrow$ **Input**: Level-Decrement threshold, Level-Increment threshold
$Levels = \{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\}$
Compute $\mathcal{P}(g_t^*|\xi_{0:t})$
if $\mathcal{P}(g_t^*|\xi_{0:t}) < \tau_B$ **then**
  $\mathcal{C}_{t+1} \leftarrow max(1, \mathcal{C}_t - 1)$
else if $\mathcal{P}(g_t^*|\xi_{0:t}) > \tau_G$ **then**
  $\mathcal{C}_{t+1} \leftarrow min(3, \mathcal{C}_t + 1)$
else
  $\mathcal{C}_{t+1} \leftarrow \mathcal{C}_t$
end if
return $\{\mathcal{C}_{t+1}\}$

confidence, frustration, and confusion. Emotional and mental states can be self-reported on a numerical scale (Marcus, Neuman, & MacKuen, 2017) via a slider (Betella & Verschure, 2016), which we use in our study, or dial (Ruef & Levenson, 2007). Following a similar numerical range of self-reported emotion state as in (Eadeh et al., 2021b, 2021a), we measured participant mental state on a subjective 1–100 interval scale questionnaire, where 100 represents maximal expression, and 1 represents the least expression. Preference (3) measures whether a person explicitly prefers instruction at a given mode of complexity. Participant preference was measured by the sum of four Likert-scale questions with responses measured on a 5-point Likert scale (Joshi, Kale, Chandel, & Pal, 2015), using Chronbach's alpha (Tavakol & Dennick, 2011) to test internal consistency of the questions ($\alpha = 0.855$). The four Likert-scale questions asked the participants to rate how they agreed with the following statements: *(1) I found the navigation instructions easy to understand. (2) I found the assistant helpful in navigating the environment. (3) I felt that the instructions were provided at an appropriate difficulty level. (4) I enjoyed performing the search-and-rescue task with the navigation assistance*. The sum of the Likert scale items calculates a composite "Preference" score, which we analyze at the interval measurement scale (Carifio & Perla, 2007; Jr & Boone, 2012). The study simultaneously tests the proposed ANA algorithm against a baseline in which instruction complexity does not change. Static instruction provides directions at only one, immutable mode of complexity, selected from $\{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\}$. Adaptive instruction refers to Algorithm 1. The mixed-design study answers the following questions: **Q1** *How does prerequisite knowledge for a task affect human performance, mental state, and preference when instructed at different modes of complexity?*, and **Q2** *What effect does adaptive versus static instruction have on human performance, mental state, and preference?*

Participants played a 2D gridworld SAR game for two rounds. One of the trials was guided by a static instructor, and the other with the adaptive (ANA) instructor. The static instructor can only provide instruction at a fixed complexity mode, selected randomly from the three modes $\{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\}$. Each trial is divided into three phases: train, pretest, and test (Fig. 6). The between-subjects independent variables are IV_1: *knowledge level* and IV_2: *instruction complexity mode*. There are three levels of IV_1: {LOW-knowledge, MED-knowledge, and HIGH-

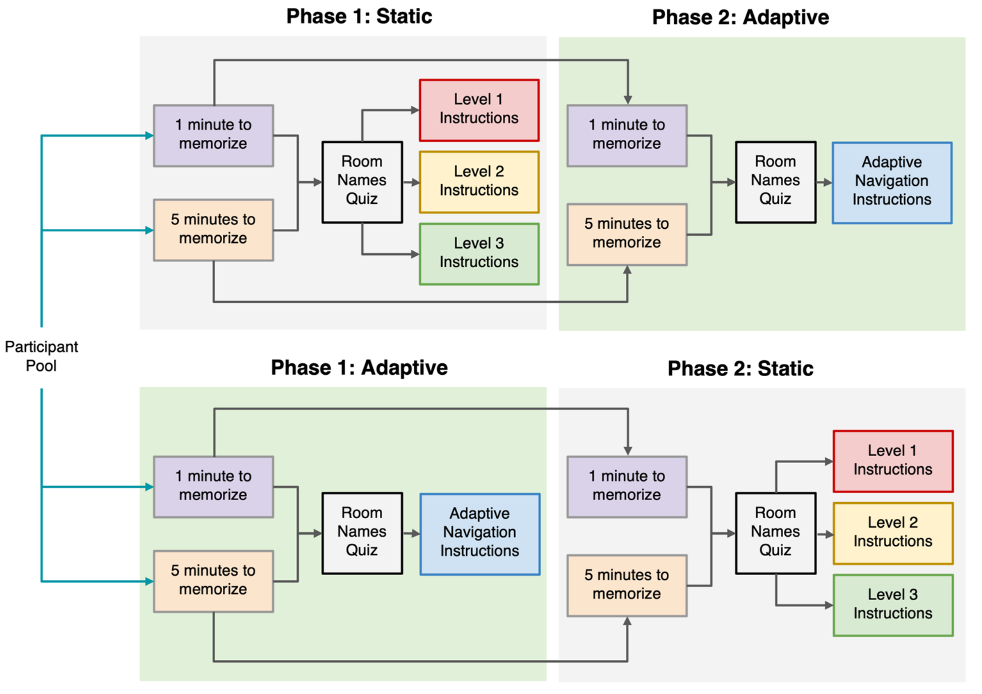

Fig. 6. In the mixed-design user study, participants played a 2D gridworld search-and-rescue game for two rounds. One of the trials was guided by a static instructor, and the other with the adaptive instructor, presented in random order. The human participant was provided a limited time period to memorize the map and room names, the duration was drawn from the set {1 min, 3 min}, in order to ensure a wide, near-uniform spread of participant preparedness. The blue arrows illustrate that participants were first randomly assigned to one of the two memorization times and one of the two instructor type orderings. The amount of time participants had to memorize the map was held constant over both trials. After memorizing, in the pretest phase, the participant was quizzed on labeling the room names. Their score on this room-labeling pretest defined their domain-knowledge score. Following the knowledge assessment, participants perform one of the missions with either the static or adaptive instructor.

knowledge} and three levels of IV_2: $\{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\}$. The within-subjects independent variable, IV_3, is *adaptivity of instruction*, of which there are two conditions: {static, adaptive (ANA)}.

In the **training** phase, the human participant was provided a limited time period to memorize the map and room names. Their memorization time window was drawn from the set {1 min, 3 min}, in order to ensure a wide, near-uniform spread of participant preparedness. The amount of time participants had to memorize the map was held constant over both trials. After memorizing, in the **pretest** phase, the participant was quizzed on labeling the room names. Their score on this room-labeling pretest defined their domain-knowledge score. Next, participants were clustered based on their knowledge scores from the pretest, and categorized into one of three groups representing their level of participant knowledge (LOW, MED, or HIGH). Defining groups on the memorization time condition is not representative of actual knowledge because memorization ability varies significantly. Clustering on the actual pretest knowledge score to define knowledge levels is a more accurate approach and makes use of the natural clustering that occurs in the knowledge of the participant pool. The knowledge group to which a participant was assigned was based on their results on the pretest score from the round in which they were partnered with the static instructor. The pretest knowledge score does not factor into the decision-making of the adaptive instructor, since the adaptive instructor infers knowledge based on the likelihood of the participant heading to intended goals. Since we are primarily interested in the relationship between participant knowledge and level of static instruction, and how this relationship affects reception and efficacy of adaptive instruction, we hold the assigned knowledge group constant from the static instructor round.

In the **test** phase, the participant played the 4-min SAR game to rescue as many victims as possible while following the directions of an AI instructor. In order to mitigate potential ordering effects, participants were counterbalanced between the ordering of the two instructor types. To mitigate memorization effects, two different environments were designed for each mission, and assigned in random order. Participants were asked a series of questions regarding each of the mental state measures (engagement, confidence, frustration, and confusion) after the first mission, either with the static or adaptive instructor. They answered the same set of questions after the mission with the other agent. After participants have worked with both instructors, they were asked which instructor they preferred.

5.4. User study results

We collected data from 108 participants over the age of 18, recruited through Prolific (Palan & Schitter, 2017). Fifty-four participants performed the first mission with the ANA adaptive instructor, followed by the static instructor, and 54 participants performed with the static instructor before the ANA adaptive instructor. Each static instruction complexity mode was played by 36 participants in total. We discarded data from participants who disobeyed the instruction more than five times, which indicated noncompliance with the instruction-following navigation procedure. This left us with 48 participants who experienced adaptive instruction before static, and 52 who experienced static instruction first.

We find that participants with low knowledge levels expressed higher engagement and confidence when their instruction complexity aligned with their knowledge (Fig. 7). LOW-knowledge participants felt significantly more engaged with $\mathcal{C}_1$ than with $\mathcal{C}_3$ instructions ($p = .019$) and were significantly more engaged with $\mathcal{C}_2$ than with $\mathcal{C}_3$ instructions ($p = .028$). HIGH-knowledge participants were significantly more engaged with $\mathcal{C}_2$ instructions than $\mathcal{C}_1$ instructions ($p = .045$). LOW-knowledge participants felt significantly more confident with $\mathcal{C}_1$ than with $\mathcal{C}_2$ instructions ($p = .039$). MED-knowledge participants felt significantly more confident with $\mathcal{C}_1$ instructions than with $\mathcal{C}_3$ instructions ($p = .014$). HIGH-knowledge participants did not express significant differences in confidence at different modes of instruction complexity.

The results also show that misalignment for participants with LOW-knowledge levels causes higher confusion and frustration. But effects of misalignment are not significant for participants with MED- or HIGH-knowledge. LOW-knowledge participants felt significantly more frustrated with $\mathcal{C}_2$ than with $\mathcal{C}_1$ instructions ($p < .001$), and were even more frustrated with $\mathcal{C}_2$ instructions than with $\mathcal{C}_3$ ($p = .003$). There were no significant post-hoc mean dif-

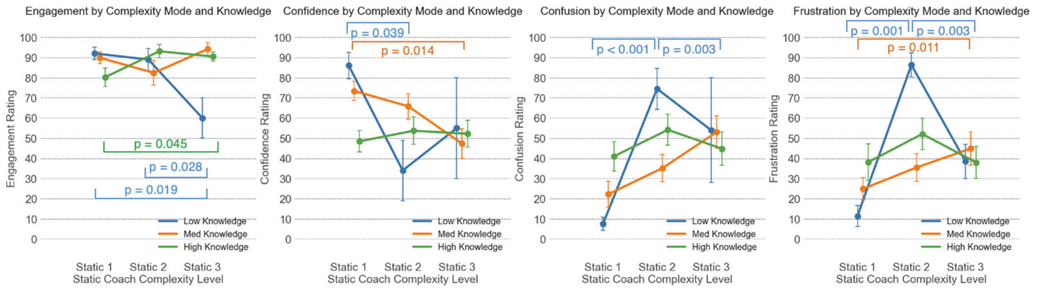

Fig. 7. Participants with aligned complexity mode and knowledge expressed higher engagement and confidence, and lower confusion and frustration. People in the LOW-knowledge condition were particularly affected by misalignment. Error bars represent standard error. p-Values are shown for significant post-hoc pairwise differences. The results also show that misalignment for participants with LOW-knowledge levels causes higher confusion and frustration. But effects of misalignment are not significant for participants with MED- or HIGH- knowledge. LOW-knowledge participants felt significantly more confused with $\mathcal{C}_2$ instructions than with $\mathcal{C}_1$ instructions. MED-knowledge participants felt significantly more confused and less engaged with $\mathcal{C}_3$ instructions than with $\mathcal{C}_1$ instructions.

ferences in the frustration of MED-knowledge nor HIGH-knowledge participants. LOW-knowledge participants felt significantly more confused with $\mathcal{C}_2$ instructions than with $\mathcal{C}_1$ instructions ($p = .001$). MED-knowledge participants felt significantly more confused with $\mathcal{C}_3$ instructions than with $\mathcal{C}_1$ instructions ($p = .011$). HIGH-knowledge participants did not express significant differences in confusion at different modes of instruction complexity (Fig. 7).

We conducted a mixed, repeated-measures ANOVA to evaluate the ANA algorithm for adaptive instruction. We did not find significant effects of the adaptivity of instruction on performance ($\lambda_{Wilks} = 1.00$, $F(1, 91) = 0.009$, $p = .923$), measured by the number of victims saved during the round (Fig. 8). The results of the mixed, repeated-measures ANOVA showed that there was not a significant effect of the adaptivity of instructor type on engagement ($\lambda_{Wilks} = 0.992$, $F(1, 91) = 0.771$, $p = .382$), nor was there a significant effect of the adaptivity of instructor type on confidence ($\lambda_{Wilks} = 1.00$, $F(1, 91) = 0.012$, $p = .912$) (Fig. 9). However, we found a significant interaction effect between instructor adaptivity, static complexity mode, and knowledge level with the static instruction on engagement ($\lambda_{Wilks} = 836$, $F(4, 91) = 4.461$, $p = .002$, $\eta_p^2 = 0.164$). Post-hoc tests show that (HIGH-knowledge,$\mathcal{C}_1$) participants experienced an increase in engagement when instructed adaptively significantly greater ($p = .021$) than that of (LOW-knowledge, $\mathcal{C}_1$) participants, by a 26.44-point increase on average (Fig. 9). Thus, we find that participants with high knowledge misaligned with their static instruction complexity mode, will have a greater increase in engagement with the adaptive instruction, over people who compared adaptive instruction against static instruction aligned with their knowledge.

In order to better understand how the instruction was provided by the adaptive navigation assistant, we dive into different trial instances where participants of varying knowledge collaborated with the adaptive assistant (Fig. 10). We illustrate the switching between levels

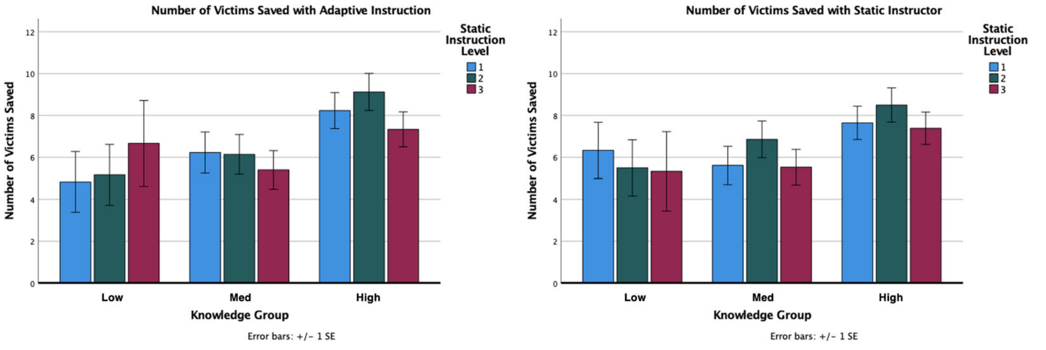

Fig. 8. Performance on the task was measured by the number of victims saved in a 4-min guided search-and-rescue mission. The number of victims saved by participants when coached with adaptive instruction was higher when participants had higher knowledge (left). The number of victims saved with static instructions was similarly higher when participants had higher knowledge (right). However, we did not find significant differences between the number of victims saved with the adaptive instructor and number of victims saved with the static instructor ($\lambda_{Wilks} = 1.00$, $F(1, 91) = 0.009$, $p = .923$). This indicates that our adaptive instruction mechanism did not affect the performance of participants on the task.

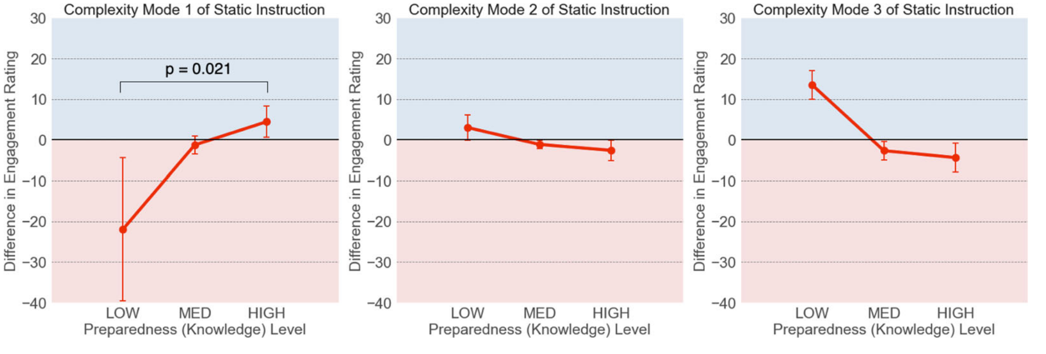

Fig. 9. The adaptive ANA algorithm did not improve engagement of participants across the board. The ANA algorithm improves the engagement of participants whose knowledge with the static instructor was misaligned with the static instruction complexity. The y-axis represents the difference between participant engagement with the adaptive instructor and engagement with the static instructor. Differences below 0, denoted in red, indicate that participant engagement was higher with static instruction. Differences above 0, denoted in blue, indicate that participants were more engaged with adaptive instruction. Error bars represent standard error, and points represent means. The leftmost plot shows that (HIGH-knowledge,$\mathcal{C}_1$) participants experienced an increase in engagement when instructed adaptively significantly greater ($p = .021$) than that of (LOW-knowledge, $\mathcal{C}_1$) participants, by a 26.44-point increase on average. Participants with high knowledge, who were given very basic static instructions, found adaptive instruction to be significantly more engaging. But participants whose knowledge with the static instructor was correctly aligned with the static instruction complexity found the adaptive algorithm less engaging.

of instruction that the adaptive assistance performs to the human rescuer, plotting the complexity level of instruction provided over time. We demonstrate how the instruction complexity changes for a participant in the LOW (blue), MED (red), and HIGH (green) knowledge conditions. During the majority of the mission, the LOW-knowledge participant received

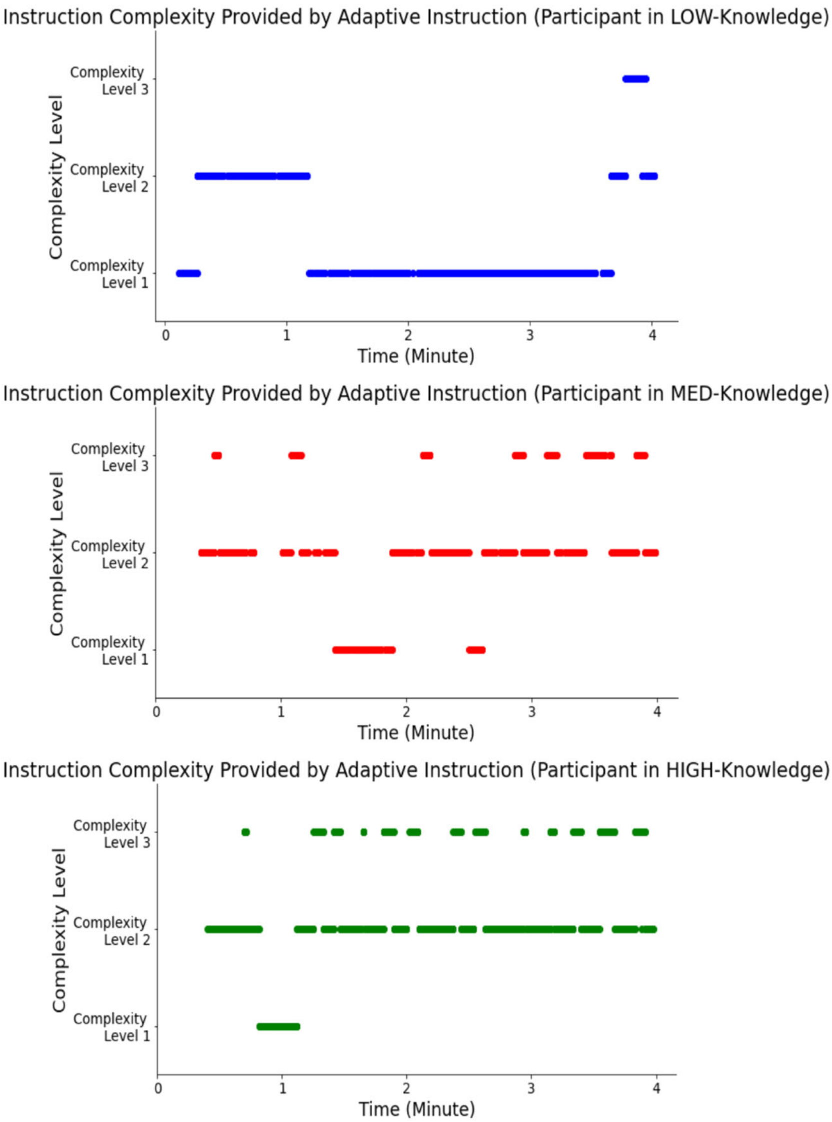

Fig. 10. We demonstrate the levels of navigation instruction complexity provided by the adaptive assistance to the human partner in three instances. The three instances we show are for different participants assigned to each of the knowledge levels. One participant is LOW-knowledge (blue), another is MED-knowledge (red), and the last is HIGH-knowledge (green). During the majority of the mission, the LOW-knowledge participant received mostly simple level $\mathcal{C}_1$ instructions, and received few of the more complex $\mathcal{C}_2$ or $\mathcal{C}_3$ instructions. The MED-knowledge and HIGH-knowledge participants received mostly level $\mathcal{C}_2$ instructions; however, the HIGH-knowledge participant received more high-complexity $\mathcal{C}_3$ instructions than the MED-knowledge participant.

mostly level $\mathcal{C}_1$ instructions, which are fairly simple and do not require prerequisite knowledge. This is an expected behavior of the adaptive algorithm, as it should be able to recognize that the participant requires very basic instructions, as a result of their having little knowledge of the map. The LOW-knowledge participant received some of the more complex $\mathcal{C}_2$ instructions, but very few $\mathcal{C}_3$ instructions, which require prerequisite knowledge. The MED-knowledge and HIGH-knowledge participants received mostly level $\mathcal{C}_2$ instructions. However, the HIGH-knowledge participant received fewer basic $\mathcal{C}_1$ instructions than the MED-knowledge participant and more high-complexity $\mathcal{C}_3$ instructions than the MED-knowledge participant. This is expected behavior from the adaptive algorithm, in that HIGH-knowledge participants would be better equipped to understand $\mathcal{C}_3$ instructions than MED-knowledge participants, because $\mathcal{C}_3$ instructions require people to draw upon their memorization of the map.

5.5. *Case study takeaways*

The results of our user study demonstrated that the alignment of complexity mode to knowledge of human instruction-followers significantly affects their expression of positive mental states. People guided by navigation instructions at an abstraction mode that they are able to sufficiently understand given their domain knowledge are more engaged and more confident than people guided by instructions misaligned to their knowledge. In particular, people in the extremes of the levels of prerequisite knowledge (assigned in the LOW-knowledge or HIGH-knowledge groups) are highly affected by complexity-knowledge alignment. Providing simple $\mathcal{C}_1$ instructions disengages players with high knowledge, likely because the instructions are too simple and verbose. Complex $\mathcal{C}_3$ instruction disengages players with low knowledge, likely because the unprepared players lack sufficient understanding and find $\mathcal{C}_3$ instructions too difficult to comprehend. Furthermore, alignment of complexity mode to knowledge of followers significantly decreases the expression of negative mental states: frustration and confusion. Participants with LOW-knowledge felt significantly more frustrated and confused with $\mathcal{C}_2$ instructions than with $\mathcal{C}_1$ instructions. They were also more frustrated and confused with $\mathcal{C}_2$ instructions than with $\mathcal{C}_3$ instructions. This illuminates a potential shortcoming in our generation of $\mathcal{C}_2$ instructions. Although $\mathcal{C}_2$ instructions occur at a lower level of abstraction than $\mathcal{C}_3$, in that $\mathcal{C}_2$ instructions do not operate at room-level abstraction, $\mathcal{C}_2$ instructions require a different sort of knowledge from the participant: the mapping from object names to their interface renderings. $\mathcal{C}_2$ includes instructions, such as "turn left at the third door on your right" and "proceed to the end of the hallway and turn left." These instructions do not require the person to remember the room names and locations, but they require the person to know what the doors and hallways look like, and potentially count the doors as they pass them. This may have been additional complexity associated with $\mathcal{C}_2$ instructions that LOW-knowledge participants found difficult, frustrating, and confusing. This represents a significant shortcoming of our instruction complexity manipulation, in that instruction complexity was not increased along the dimension of abstraction only. The $\mathcal{C}_2$ instructions introduce complexity along a dimension of interface understanding, which may have impacted our findings on mental states, performance, and preferences, since $\mathcal{C}_2$ instructions may not necessarily be less complex than $\mathcal{C}_3$ instructions. Overall, our analyses suggest that people with high prerequisite knowledge are less susceptible than those with insufficient knowledge to negative mental state changes as a result of instruction complexity. The suitability of abstraction modalities depends on the preparedness of the human.

The study demonstrated that the adaptive mechanism may improve the engagement of participants whose knowledge is misaligned with static instruction complexity. In particular, highly knowledgeable participants who were provided low complexity $\mathcal{C}_1$ instructions felt more engaged with the adaptive instructions. Engagement was the only mental state significantly affected by the new adaptive algorithm. The improvement in engagement may also be due to participants being shown aligned instructions at various times during the mission, as a result of the adaptive mechanism switching between the three levels of complexity, not the algorithm for electing to change levels itself. These results demonstrate that our adaptive algorithm was not particularly effective in assisting the subject. The method also did not offer significant improvements to performance of the SAR task, nor did participants express significant preference for the adaptive instructions.

One possibility is that fixed thresholds for determining when to switch abstraction mode limited the adaptive benefit. User input to perform complexity changes are a way to tune custom thresholds in future work. Fig. 10 visualizes how the complexity level of adaptive instruction changed over the course of the trial for participants in each knowledge condition. For the HIGH-knowledge and MED-knowledge participants, the switching occurred frequently. This frequent switching of instruction style may have caused the participants to become distracted or ignore instructions. Another possibility is that the navigation task was too simple and thus unsuitable for the different complexity levels to impact comprehension significantly. While our proposed method does not significantly improve performance in the SAR task domain, the method boosts the engagement of people who experience complexity-knowledge misalignment. Overall, our evaluation demonstrates that the level of abstraction in navigation instructions matters to human users, and that matching abstraction complexity to human knowledge influences human mental states.

This case study demonstrates an example of how the POMDP human estimate framework can be applied to formalizing the agent's assumptions about human decision-making. While adapting instruction complexity to human knowledge appears natural and intuitive, using only human knowledge as a salient feature for adaptation involves priors in the robot's mental model of the human. An important motivator for the adaptive Algorithm 1 was that lower complexity instructions can be understood with less complete knowledge of the full environment. The POMDP framework illuminates this more explicitly, and translates to the agent having an estimate of the human policy π^{H_R}, which relies less on the human's state beliefs when given instructions at $\mathcal{C}_1$.

6. Discussion

In this work, we present a framework for considering robot adaptation to human partners. Adaptation in human–AI teaming is an interaction in which an AI changes its behavior, based on understanding of human teammates that is not known a priori. Performing adaptation consists of learning some set of relevant human features and changing robot behavior in response in order to influence the collaboration. Our framework is feature-agnostic, in that it can be applied to various adaptation parameters, including human policies, cognitive features (such as knowledge and intentions), as well as other internal factors, such as fatigue and physical capabilities. Under the feature-agnostic framework, learning designated human features reduces to estimation of a component of the human's decision process.

Under this constraint, the formalism also allows for more explicit reasoning about the priors the robot has on the human's decision-making process. Designers of adaptive human–robot interactions select features of the human partner that should be adapted to based on the needs of the collaboration. Components of the human's POMDP that are not the adaptation parameter are implicitly or explicitly determined by the interaction design. For example, adapting to a human partner's goals learns a reward function, while the robot must use the task specifications to determine the actions, states, and observations of the human POMDP as priors. The full human POMDP is not always explicitly estimated in adaptive human–robot interactions; however, viewing the learning and adaptation process under this formalism helps illuminate knowledge about the human partner assumed by the robot and how relevant adaptation parameters influence the human's decision process.

The framework assumes that the human decision-making can be modeled as a POMDP, and consequently, human feature adaptation parameter can be modeled as a component of the human's POMDP. There may be cases in which the salient human feature for adaptation cannot be represented as a component of the human's POMDP, such as when reasoning about collaboration is non-Markovian. In such cases, a POMDP model may not be appropriate for representing the human partner and a different decision process model may be required. The model also assumes that the human and robot partners are both operating at the same timescale. When designing human–robot interactions to facilitate adaptation, the components of the human model which are not learned by the robot are often predefined based on interaction context, which often relies on the design and assumptions of the designer. Assumptions on how the estimated human POMDP is parameterized may not be completely representative of the human's experience or decisions, which is an existing challenge in modeling human behavior. Modeling humans as POMDP decision makers and updating the parameters based on salient cues helps to illuminate more clearly the assumptions used.

7. Conclusion

The ability for AI teammates to adapt to human partners is critical for effective and fluent team collaboration. To better collaborate and understand their teammates, robot and AI teammates must adapt to features related to the human teammates. Modeling human partners is an effective way for robots to build an understanding of how their partners will operate. We explore a framework for modeling the salient features of human partners to which robots adapt as components of the robot's estimate of the human's decision process (POMDP). Through a case study, we demonstrate how this framework can be applied to adapting to a human partner's knowledge in a coached navigation task. Although the results of the case study do not show significant task performance benefits of an adaptive algorithm, the results suggest that when communicating with a partner, an AI should consider the expertise of their partner in order to determine whether their partner can understand the information they are receiving. Forgoing such individual considerations may adversely impact the partner's mental state and proficiency. We view this work as a candidate perspective on defining adaptive interactions between humans and AI. Teaming is dynamic in nature, and developing the mechanisms by which AI build an understanding of and adapt to human partners is an ongoing technical challenge for collaborative AI systems.

Acknowledgments

This research is based upon work supported by the Defense Advanced Research Projects Agency, award number: FP00002636. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of DARPA.

References

Aggarwal, I., Woolley, A. W., Chabris, C. F., & Malone, T. W. (2019). The impact of cognitive style diversity on implicit learning in teams. *Frontiers in Psychology*, *10*.

Ahmad, M. I., Mubin, O., & Orlando, J. (2017). A systematic review of adaptivity in human–robot interaction. *Multimodal Technologies and Interaction*, *1*(3), 14.

Baker, C. L., Saxe, R., & Tenenbaum, J. B. (2009). Action understanding as inverse planning. *Cognition*, *113*(3), 329–349.

Baker, C. L., Tenenbaum, J. B., & Saxe, R. (2006). Bayesian models of human action understanding. *Advances in Neural Information Processing Systems*, *18*, 99.

Barrett, S., Rosenfeld, A., Kraus, S., & Stone, P. (2017). Making friends on the fly: Cooperating with new teammates. *Artificial Intelligence*, *242*, 132–171.

Barrett, S., & Stone, P. (2015). Cooperating with unknown teammates in complex domains: A robot soccer case study of ad hoc teamwork. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Barry, D. N., & Love, B. C. (2021). Human learning follows the dynamics of gradient descent.

Betella, A., & Verschure, P. (2016). The affective slider: A digital self-assessment scale for the measurement of human emotions. *PLoS ONE*, *11*, e0148037.

Boutilier, C., Dean, T., & Hanks, S. (1999). Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, *11*(1), 1–94.

Brown, N. G., & Tellex, S. (2010). Modeling and solving human–robot collaborative tasks using POMDPS.

Burke, C. S., Stagl, K. C., Salas, E., Pierce, L., & Kendall, D. (2006). Understanding team adaptation: A conceptual analysis and model. *Journal of Applied Psychology*, *91*(6), 1189.

Burke, J. L., Murphy, R. R., Riddle, D. R., & Fincannon, T. (2004). Task performance metrics in human–robot interaction: Taking a systems approach. Technical report. University of South Florida, Tampa Center for Robot-Assisted Search and Rescue.

Carifio, J., & Perla, R. (2007). Ten common misunderstandings, misconceptions, persistent myths and urban legends about Likert scales and Likert response formats and their antidotes. *Journal of Social Sciences*, *3*.

Chilton, M. A., Hardgrave, B. C., & Armstrong, D. J. (2010). Performance and strain levels of it workers engaged in rapidly changing environments: A person-job fit perspective. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, *41*(1), 8–35.

Clark, H. H., & Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition*, *22*(1), 1–39.

Dafoe, A., Hughes, E., Bachrach, Y., Collins, T., McKee, K. R., Leibo, J. Z., Larson, K., & Graepel, T. (2020). Open problems in cooperative AI.

Dunnette, M. (2014). Critical concepts in the assessment of human capabilities. *Human performance and productivity: Volumes 1, 2, and 3*.

Eadeh, F., Zhao, M., Nguyen, T. N., Gupta, P., Gonzalez, C., Admoni, H., & Woolley, A. W. (2021a). Can't get you off of my mind: The detrimental effects of anger and rumination for team performance. In *16th Annual Conference for Interdisciplinary Network for Group Research (INGRoup)*.

Eadeh, F., Zhao, M., Nguyen, T. N., Gupta, P., Gonzalez, C., Admoni, H., & Woolley, A. W. (2021b). Does anger help or hurt individual and team performance? In *ACM Collective Intelligence Conference*.

Endsley, M. R. (2017). Toward a theory of situation awareness in dynamic systems. In *Situational awareness* (pp. 9–42). Routledge.

Gobet, F., Richman, H., Staszewski, J., & Simon, H. A. (1997). Goals, representations, and strategies in a concept attainment task: The EPAM model.

Gombolay, M. C., Huang, C., & Shah, J. (2015). Coordination of human–robot teaming with human task preferences. In *2015 AAAI Fall Symposium Series*.

Görür, O. C., Rosman, B., Sivrikaya, F., & Albayrak, S. (2018). Social cobots: Anticipatory decision-making for collaborative robots incorporating unexpected human behaviors. In *Proceedings of the 2018 ACM/IEEE International Conference on Human–Robot Interaction* (pp. 398–406).

Hancock, P. A., Billings, D. R., Schaefer, K. E., Chen, J. Y., de Visser, E., & Parasuraman, R. (2011). A meta-analysis of factors affecting trust in human–robot interaction. *Human Factors: The Journal of Human Factors and Ergonomics Society*, *53*, 517–527.

Hauskrecht, M. (2000). Value-function approximations for partially observable Markov decision processes. *Journal of Artificial Intelligence Research*, *13*, 33–94.

Hawkins, R., Kwon, M., Sadigh, D., & Goodman, N. (2020). Continual adaptation for efficient machine communication. In *Proceedings of the 24th Conference on Computational Natural Language Learning* (pp. 408–419).

Hawkins, R. X., Frank, M., & Goodman, N. D. (2017). Convention-formation in iterated reference games. In *CogSci*.

Hoffman, G. (2019). Evaluating fluency in human–robot collaboration. *IEEE Transactions on Human–Machine Systems*, *49*(3), 209–218.

Holroyd, A., Rich, C., Sidner, C. L., & Ponsler, B. (2011). Generating connection events for human–robot collaboration. In *2011 RO-MAN* (pp. 241–246). IEEE.

Huang, C.-M., Cakmak, M., & Mutlu, B. (2015). Adaptive coordination strategies for human–robot handovers. In *Robotics: Science and systems* (volume 11, pp. 1–10). Rome, Italy.

Joshi, A., Kale, S., Chandel, S., & Pal, D. (2015). Likert scale: Explored and explained. *British Journal of Applied Science and Technology*, *7*, 396–403.

Jr, H., & Boone, D. (2012). Analyzing Likert data. *Journal of Extension*, *50*.

Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, *101*(1), 99–134.

Kawanaka, T., & Stigler, J. (1999). Teachers' use of questions in eighth-grade mathematics classrooms in Germany, Japan, and the United States. *Mathematical Thinking and Learning*, *1*, 255–278.

Khani, F., Goodman, N. D., & Liang, P. (2018). Planning, inference and pragmatics in sequential language games. *Transactions of the Association for Computational Linguistics*, *6*, 543–555.

Khoramshahi, M., & Billard, A. (2019). A dynamical system approach to task-adaptation in physical human–robot interaction. *Autonomous Robots*, *43*(4), 927–946.

Kim, J., & Lee, H. (2020). Adaptive human–machine evaluation framework using stochastic gradient descent-based reinforcement learning for dynamic competing network. *Applied Sciences*, *10*(7), 2558.

Koay, K. L., Syrdal, D. S., Walters, M. L., & Dautenhahn, K. (2007). Living with robots: Investigating the habituation effect in participants' preferences during a longitudinal human–robot interaction study. In *RO-MAN 2007-The 16th IEEE International Symposium on Robot and Human Interactive Communication* (pp. 564–569). IEEE.

Koert, D., Pajarinen, J., Schotschneider, A., Trick, S., Rothkopf, C., & Peters, J. (2019). Learning intention aware online adaptation of movement primitives. *IEEE Robotics and Automation Letters*, *4*(4), 3719–3726.

Koppol, P., Admoni, H., & Simmons, R. G. (2021). Interaction considerations in learning from humans. In *IJCAI* (pp. 283–291).

Koppula, H. S., Jain, A., & Saxena, A. (2016). Anticipatory planning for human–robot teams. In *Experimental robotics* (pp. 453–470). Springer.

Kwon, M., Biyik, E., Talati, A., Bhasin, K., Losey, D. P., & Sadigh, D. (2020). When humans aren't optimal: Robots that collaborate with risk-aware humans. In *2020 15th ACM/IEEE International Conference on Human–Robot Interaction (HRI)* (pp. 43–52). IEEE.

Kwon, M., Jung, M. F., & Knepper, R. A. (2016). Human expectations of social robots. In *2016 11th ACM/IEEE International Conference on Human–Robot Interaction (HRI)* (pp. 463–464).

Lam, C.-P., & Sastry, S. S. (2014). A POMDP framework for human-in-the-loop system. In *53rd IEEE Conference on Decision and Control* (pp. 6031–6036).

Levine, S. J., & Williams, B. C. (2018). Watching and acting together: Concurrent plan recognition and adaptation for human–robot teams. *Journal of Artificial Intelligence Research*, *63*, 281–359.

Lewis, M., Sycara, K., & Walker, P. (2018). *The role of trust in human–robot interaction* (pp. 135–159).

Li, H., Ni, T., Agrawal, S., Jia, F., Raja, S., Gui, Y., Hughes, D., Lewis, M., & Sycara, K. (2021). Individualized mutual adaptation in human–agent teams. *IEEE Transactions on Human–Machine Systems*, *51*(6), 706–714.

Li, Y., Tee, K. P., Chan, W. L., Yan, R., Chua, Y., & Limbu, D. K. (2015a). Continuous role adaptation for human–robot shared control. *IEEE Transactions on Robotics*, *31*(3), 672–681.

Li, Y., Tee, K. P., Chan, W. L., Yan, R., Chua, Y., & Limbu, D. K. (2015b). Role adaptation of human and robot in collaborative tasks. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 5602–5607). IEEE.

Li, Y., Yang, C., & He, W. (2016). Towards coordination in human–robot interaction by adaptation of robot's cost function. In *2016 International Conference on Advanced Robotics and Mechatronics (ICARM)* (pp. 254–259). IEEE.

Madni, A. M., & Madni, C. C. (2018). Architectural framework for exploring adaptive human–machine teaming options in simulated dynamic environments. *Systems*, *6*(4).

Marcus, G. E., Neuman, W. R., & MacKuen, M. B. (2017). Measuring emotional response: Comparing alternative approaches to measurement. *Political Science Research and Methods*, *5*(4), 733–754.

Mertens, A., Reiser, U., Brenken, B., Lüdtke, M., Hägele, M., Verl, A., Brandl, C., & Schlick, C. (2011). Assistive robots in eldercare and daily living: Automation of individual services for senior citizens. In *International Conference on Intelligent Robotics and Applications* (pp. 542–552). Springer.

Monahan, G. E. (1982). A survey of partially observable Markov decision processes: Theory, models, and algorithms. *Management Science*, *28*(1), 1–16.

Monfort, M., Liu, A., & Ziebart, B. D. (2015). Intent prediction and trajectory forecasting via predictive inverse linear-quadratic regulation. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, AAAI'15 (pp. 3672–3678). AAAI Press.

Motowildo, S. J., Borman, W. C., & Schmit, M. J. (1997). A theory of individual differences in task and contextual performance. *Human Performance*, *10*(2), 71–83.

Nathan, M., & Kim, S. (2009). Regulation of teacher elicitations in the mathematics classroom. *Cognition and Instruction*, *27*, 91–120.

Nemec, B., Likar, N., Gams, A., & Ude, A. (2018). Human robot cooperation with compliance adaptation along the motion trajectory. *Autonomous Robots*, *42*(5), 1023–1035.

Ng, A. Y., & Russell, S. (2000). Algorithms for inverse reinforcement learning. In *Proceedings of the 17th International Conference on Machine Learning* (pp. 663–670). Morgan Kaufmann.

Nguyen, T.-H. D., Hsu, D., Lee, W.-S., Leong, T.-Y., Kaelbling, L. P., Lozano-Perez, T., & Grant, A. H. (2011). CAPIR: Collaborative action planning with intention recognition. In *Seventh Artificial Intelligence and Interactive Digital Entertainment Conference*.

Nicholls, J. G., Cheung, P. C., Lauer, J., & Patashnick, M. (1989). Individual differences in academic motivation: Perceived ability, goals, beliefs, and values. *Learning and Individual Differences*, *1*(1), 63–84.

Nikolaidis, S., Hsu, D., & Srinivasa, S. (2017). Human–robot mutual adaptation in collaborative tasks: Models and experiments. *International Journal of Robotics Research*, *36*(5–7), 618–634.

Nikolaidis, S., Nath, S., Procaccia, A. D., & Srinivasa, S. (2017). Game-theoretic modeling of human adaptation in human–robot collaboration. In *Proceedings of the 2017 ACM/IEEE International Conference on Human–Robot Interaction* (pp. 323–331).

Nikolaidis, S., & Shah, J. (2013). Human–robot cross-training: Computational formulation, modeling and evaluation of a human team training strategy. In *2013 8th ACM/IEEE International Conference on Human–Robot Interaction (HRI)* (pp. 33–40). IEEE.

Palan, S., & Schitter, C. (2017). Prolific.ac-A subject pool for online experiments. *Journal of Behavioral and Experimental Finance*, *17*.

Patel, M., Roberts, R. E., Riyaz, M. U., Ahmed, M., Buckwell, D., Bunday, K., Ahmad, H., Kaski, D., Arshad, Q., & Bronstein, A. M. (2015). Locomotor adaptation is modulated by observing the actions of others. *Journal of Neurophysiology*, *114*(3), 1538–1544.

Peternel, L., Tsagarakis, N., Caldwell, D., & Ajoudani, A. (2018). Robot adaptation to human physical fatigue in human–robot co-manipulation. *Autonomous Robots*, *42*(5), 1011–1021.

Rookhuiszen, R., Obbink, M., & Theune, M. (2009). Two approaches to give: Dynamic level adaptation versus playfulness.

Rosenthal, S., & Veloso, M. (2011). Modeling humans as observation providers using POMDPS. In *2011 RO-MAN* (pp. 53–58).

Ruef, A. M., & Levenson, R. W. (2007). Continuous measurement of emotion: The affect rating dial. In J. A. Coan & J. J. B. Allen (Eds.), *Handbook of emotion elicitation and assessment* (pp. 286–297). Oxford University Press.

Shih, A., Sawhney, A., Kondic, J., Ermon, S., & Sadigh, D. (2021). On the critical role of conventions in adaptive human–AI collaboration. *International Conference on Learning Representations (ICLR) 2021*.

Sidner, C. L., & Lee, C. (2003). Engagement rules for human–robot collaborative interactions. In *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme-System Security and Assurance (Cat. No. 03CH37483)*(volume *4*, pp. 3957–3962). IEEE.

Sidner, C. L., Lee, C., Kidd, C. D., Lesh, N., & Rich, C. (2005). Explorations in engagement for humans and robots. *Artificial Intelligence*, *166*(1), 140–164.

Sonnentag, S., & Frese, M. (2012). Dynamic performance.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press.

Taha, T., Miró, J. V., & Dissanayake, G. (2011). A POMDP framework for modelling human interaction with assistive robots. In *2011 IEEE International Conference on Robotics and Automation* (pp. 544–549). IEEE.

Tavakol, M., & Dennick, R. (2011). Making sense of Cronbach's alpha. *International Journal of Medical Education*, *2*, 53–55.

Torrey, C., Fussell, S. R., & Kiesler, S. (2013). How a robot should give advice. In *2013 8th ACM/IEEE International Conference on Human–Robot Interaction (HRI)* (pp. 275–282). IEEE.

Vignolo, A., Powell, H., Rea, F., Sciutti, A., Mcellin, L., & Michael, J. (2021). A humanoid robot's effortful adaptation boosts partners' commitment to an interactive teaching task. *ACM Transactions on Human–Robot Interaction*, *11*(1), 1–17.

Walker, M. (1997). Review of using language by Herbert H. Clark. Cambridge University Press 1996. *Computational Linguistics*, *23*, 625–628.

Woodward, M. P., & Wood, R. J. (2012). Learning from humans as an I-POMDP. *Computing Research Repository (CoRR)*, *abs/1204.0274*.

Zanchettin, A. M., Casalino, A., Piroddi, L., & Rocco, P. (2018). Prediction of human activity patterns for human–robot collaborative assembly tasks. *IEEE Transactions on Industrial Informatics*, *15*(7), 3934–3942.

Zhang, S., Chen, Y., Zhang, J., & Jia, Y. (2020). Real-time adaptive assembly scheduling in human-multi-robot collaboration according to human capability. In *2020 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 3860–3866). IEEE.

Ziebart, B., Ratliff, N., Gallagher, G., Mertz, C., Peterson, K., Bagnell, J., Hebert, M., Dey, A., & Srinivasa, S. (2009). Planning-based prediction for pedestrians (pp. 3931–3936).