Anticipating False Beliefs and Planning Pertinent Reactions in Human-Aware Task Planning with Models of Theory of Mind

Anthony Favier[1,2], **Shashank Shekhar**[1], **Rachid Alami**[1,2*]

[1]LAAS-CNRS, Universite de Toulouse, CNRS, INSA, UPS, Toulouse, France
[2]Artificial and Natural Intelligence Toulouse Institute (ANITI)
{anthony.favier, sshekhar, rachid.alami}@laas.fr

Abstract

It is essential for a collaborative robot to consider Theory of Mind (ToM) when interacting with humans. Indeed, performing an action in the absence of another agent may create false beliefs like in the well-known *Sally & Anne Task* (Wimmer and Perner 1983). The robot should be able to detect, react to, and even anticipate false beliefs of other agents with a detrimental impact on the task to achieve. Currently, ToM is mainly used to control the task execution and resolve in a reactive way the detrimental false beliefs. Some works introduce ToM at the planning level by considering distinct beliefs, and we are in this context. This work proposes an extension of an existing human-aware task planner and effectively allows the robot to anticipate a false human belief ensuring a smooth collaboration through an implicitly coordinated plan. First, we propose to capture the observability properties of the environment in the state description using two observability types and the notion of *co-presence*. They allow us to maintain distinct agent beliefs by reasoning directly on what agents can observe through specifically modeled *Situation Assessment* processes, instead of reasoning of action effects. Then, thanks to the better estimated human beliefs, we can predict if a false belief with adverse impact will occur. If that is the case then, first, the robot's plan can be to communicate minimally and proactively. Second, if this false belief is due to a non-observed robot action, the robot's plan can be to postpone this action until it can be observed by the human, avoiding the creation of the false belief. We implemented our new conceptual approach, prove its soundness and completeness, discuss its effectiveness qualitatively, and show experimental results on three novel domains.

1 Introduction

Human-**R**obot **C**ollaboration (HRC) is a current research focus due to the growing number of robot-assisted applications (Selvaggio et al. 2021; Clodic et al. 2017). Collaborative robots add clear value to real-world domains like household (Unhelkar, Li, and Shah 2020), workshops (Unhelkar et al. 2018), or medical facilities (Jacob et al. 2013).

Consider a shared task scenario where a robot and a human need to cook pasta together, without any prior negotiation about the exact sequence of actions to execute. This

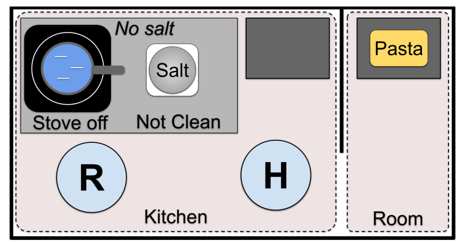

Figure 1: Let us consider cooking pasta as a human-robot shared task. The robot has to turn on the stove (*StoveOn*) and clean the counter (*CounterClean*), but the latter is not a part of the shared task. The human takes care of fetching the pasta while both agents can add salt into the water (*SaltIn*). Before pouring the pasta into the pot the human must know the facts, *StoveOn* and *SaltIn*. Unlike *StoveOn*, the facts *SaltIn* and *CounterClean* are not directly observable. Hence, by acting while the human is away to fetch the pasta, the robot may induce false beliefs which may be detrimental to the shared task (e.g., human adding salt again).

scenario is depicted in Figure 1. In the kitchen, there is already a pot filled with water placed on a stove, but the pasta bag is stored in an adjacent room. This cooking task consists of pouring the pasta into the pot, but only after turning on the stove (*StoveOn*) and after adding salt in the water (*SaltIn*). The robot is in charge of turning on the stove, the human has to fetch the pasta and pour it into the pot, while both agents can add salt to the pot. In addition, the robot has to clean the counter (*CounterClean*) but it is not part of the shared task. The human is free to either first add salt or first fetch the pasta. Depending on these uncontrollable human choices, the robot will perform different actions, which will create different false beliefs. Indeed, consider that the fact *StoveOn* is *observable*, while the facts *SaltIn* and *CounterClean* are not directly observable. Their exact value can only be *inferred* by either performing a dedicated sensing action (e.g., tasting the water and inspecting the counter); or by observing or attending the specific action execution (e.g., salt being added and counter being cleaned).

While the human is away for fetching the pasta in the other room, the robot can perform several actions. Once back in the kitchen, the human will be able to observe whether the robot earlier successfully turned on the stove

*This work has been partially funded by the Agence Nationale de la Recherche through the ANITI ANR-19-PI3A-0004 grant and by the Horizon Europe Framework Programme through the euROBIN Grant 101070596.

since it is *observable*. However, since *SaltIn* is *not observable*, the human agent is likely to believe that no salt has been added, or at least will be uncertain about the status of *SaltIn*. Instead of questioning the robot or tasting the water, it would be appreciated to have a proactive, collaborative robot avoiding this predictable *"uncomfortable"* situation to happen. For the robot to anticipate this situation while planning, it should consider Theory of Mind (ToM) to maintain distinct human beliefs.

Some work (e.g., in (Devin and Alami 2016)) already considers ToM when interacting with humans, but these considerations are attempted *"only"* in the task execution phase. Doing so already produces interesting behaviors but the robot can only be reactive to the human's *unpredictable* absence or inattention (e.g., due to a phone call and some urgent duty). However, considering ToM at the planning level allows a robot to anticipate the *predictable* absence of the human, e.g. the human leaving the kitchen to the adjacent room to bring the pasta packet. Consequently, it makes the robot proactive instead of just reactive. Indeed, the robot could act differently in order to avoid the human missing necessary information or to provide this information through communication when needed or in advance.

For seamless human-robot collaboration, we believe that it is essential not to restrict human behaviors, and hence, we consider the human as an *uncontrollable* agent. Recently proposed planning frameworks, e.g., the one using qualitative task specifications as in (Buisan et al. 2022) and another employing stochastic model as in (Unhelkar, Li, and Shah 2019), can handle uncontrollable human behaviors congruent to the shared task. These offline approaches use human task models to estimate the possible uncontrollable behaviors of the human in various situations. Doing so allows for planning robot's actions accordingly and generating robot plans that are implicitly coordinated and compliant with every possible human action.

In this work, we propose to extend our prior work HATP/EHDA, which stands for *human-aware task planning by emulating human decisions and actions* (Buisan et al. 2022). The framework models and plans with uncontrollable human operators. We first enriched HATP/EHDA's task specifications to explicitly capture the observability properties of the environment. Our new task description is based on the notion of *co-presence*, and on *two* types of facts: First, those that can be observed at any time and, second, those that cannot be observed and can only be inferred while attending the execution of an action that generates them. We then model so-called *"Situation Assessment"* (SA) processes to estimate *offline* the agents' sensing and reasoning capabilities about their surroundings at *run-time*. These processes are inserted into the existing planning workflow to manage the evolution of distinct human beliefs and better estimate the actions they are likely to perform. Finally, we can detect if a false human belief will occur, and if it *concerns* information to the human and is *essential* w.r.t. the shared task. If so, the robot's plan is updated to proactively and minimally communicate to correct the false belief before it has an impact. Moreover, if the false belief is due to a non-observed action causing a fact not inferred, we also try to postpone its execution if possible until the human is anticipated to attend it. With such implicit communication, delaying avoids the false belief to arise.

The paper is structured as follows: A comprehensive amount of related work is provided in the next section, which is followed by the underlying HATP/EHDA framework — briefly described in Section 3. Section 4 describes the formalism used in our planning scheme and how it captures the observability properties of the environment. Section 5 presents two situation assessment processes and how they are used to maintain the estimated human beliefs. Section 6 explains how to detect relevant false human beliefs and how they are corrected. Section 7 provides some formal proofs and properties of our approach. It is followed by Section 8 discussing empirical evaluation, and showing both qualitative and quantitative results. The paper ends with discussion and conclusion sections.

2 Related Work

Theory of Mind in HRC The literature in Human-Robot Collaboration (HRC) uses different variants of ToM in the execution of shared global plans. However, the focus shown is on perspective taking — a robot reasons about what humans can perceive followed by constructing a world from their frame of reference, and hence managing the agents' beliefs accordingly on the fly (Berlin et al. 2006). The framework given by Devin and Alami (2016) allows the robot to estimate the mental state of the human, containing not only their beliefs but also their actions, goals, and plans. It supports the robot's capabilities to do spatial reasoning w.r.t. the humans and tracks their activities. In particular, it manages the execution of shared plans in an object manipulation context and shows how a robot can adapt to human decisions and actions and communicates if needed.

In dynamic settings like ours, an agent may believe something true that no longer holds as a ground truth (Dissing and Bolander 2020), and it must be known to execute the agent's next action (or plan). The framework proposed in (Shvo, Klassen, and McIlraith 2022) uses agents' ToM such that agent reasons over the nested beliefs of other agents to handle misconceptions about the validity of their plans and achieves it by communicating with them or by acting in the real world. To realize their idea, the authors relate it to epistemic planning that combines reasoning and planning based on the beliefs and knowledge of agents (Bolander and Andersen 2011). However, they assume that the agents' plans and (nested) beliefs are given and that the agents are controllable. Certainly, their framework is rich and extendable to cases where agents have possible plans or include a plan recognition technique as in (Cirillo, Karlsson, and Saffiotti 2009), and resolving discrepancy based on, for example, the most probable agents' plans.

Work on human-robot cohabitation with the interest of human-aware planning is explored in (Kulkarni, Srivastava, and Kambhampati 2021; Chakraborti et al. 2015), however, unlike ours, they do not support planning for explicit shared goals/tasks such that humans and robots achieve it while collaborating and cooperating. Moreover, their frameworks allow robots to proactively assist, but only if they improve the human's current plan (and sometimes when humans do not expect such assistance).

Planning Approaches, Solution Plans, and Models Various task models have been realized in the HR (human-robot) collaborative planning context, e.g., hierarchical task networks (HTNs) (Lallement, de Silva, and Alami 2018; Roncone, Mangin, and Scassellati 2017), POMDPs (Unhelkar, Li, and Shah 2019; Roncone, Mangin, and Scassellati 2017; Unhelkar, Li, and Shah 2020), AND/OR graphs (Darvish et al. 2021), etc. A hierarchical network is created using HTNs (abstract and non-abstract tasks) and AND/OR graphs to represent the inner coupling links of the subtasks (Gombolay et al. 2016), and the plan search occurs in a depth-first manner. In (Hörger, Kurniawati, and Elfes 2019), the authors show how uncertainty can be dealt with in the evolution of the environment and agent behavior. The challenge lies with, especially in POMDPs for HRC, the hidden and implied state of the human agent (Unhelkar, Li, and Shah 2020).

The HATP frameworks extending HTNs consider agents controllable (Alami et al. 2006; Lallement, de Silva, and Alami 2018; Alili, Alami, and Montreuil 2009), while in (Roncone, Mangin, and Scassellati 2017), the framework considers planning at multiple abstraction levels (with a single HTN) with humans. It is capable of basic reasoning for role assignment and task allocation. Robots plan under state uncertainty with partially observable human intentions modeled and tackled (mainly) at the primitive task level. But these frameworks assume that a joint task is established prior to planning. Moreover, generally, they produce explicitly coordinated, shared HR plans that are legible and acceptable by humans — they are assumed to be controllable in some sense, such that the techniques rely more on the replanning aspect. In (Cirillo, Karlsson, and Saffiotti 2009; Köckemann, Pecora, and Karlsson 2014), the objectives of the humans around robots define robots' existence and contingent tasks, e.g., do not use the vacuum cleaner when humans go to sleep. However, more importantly, they do not have an explicitly shared task to achieve as a team.

The literature proposes to investigate how to create a reasonable model of humans and how to obtain task knowledge, e.g., (Unhelkar and Shah 2019). HR task planning is also modeled as an optimization problem and solved using multiple cutting-edge research, e.g., mixed integer programming used in the fine work done by (Vats, Kroemer, and Likhachev 2022). Task knowledge can be gathered offline from human psychologists and expert engineers (Levine and Williams 2014; Cirillo, Karlsson, and Saffiotti 2009); or can be learned via human tutors or from demonstrations (Koppula, Jain, and Saxena 2016); or a Markov model for sequential decision-making can also be learned from a partial specification of human behaviors (Unhelkar and Shah 2019). Hierarchical models consist of layered abstractions and are considered suitable or close to human intuitions. They help predict humans' actions, and like ours, they also help emulate human's predictable behaviors and shape robot's decision. Such models can be learned using conjugate task graphs, and to identify the task structure an aggregation algorithm can be used (Hayes and Scassellati 2016).

Communication in HR Collaboration Communication is used to align an agent's belief, clarify its decision or action, fix errors, etc (Tellex et al. 2014). Recent work deals with an explicit usage of communication actions in planning (Nikolaidis et al. 2018; Roncone, Mangin, and Scassellati 2017; Sanelli et al. 2017; Unhelkar, Li, and Shah 2020). E.g., in (Roncone, Mangin, and Scassellati 2017; Unhelkar, Li, and Shah 2020), the authors represent and plan with explicit communication actions, considering them as regular POMDP actions, such that execution policies contain them.

Redundant communication can be annoying and costly. Due to the invisible state of the human operator, their subsequent action is estimated using, e.g., tracking their *attention*, from *decision-making* models, and *motion prediction*. Next, a POMDP can be created and solved by optimizing the overall benefit/cost as in (Unhelkar, Li, and Shah 2020). In this work, we estimate the evolution of the agents' beliefs and decide "if" and "when" belief alignment is required. And, it is achieved via explicit communication actions, but with minimal communication and in a principled way. Moreover, we do not explicitly use these actions for planning (for the deliberation process) like (non-) primitive tasks.

Epistemic Planning Our notion of the "observable-fact" classified into, observable from action and observable from the state, can roughly be seen as a part of the restricted epistemic logic presented and applied in planning applications (Cooper et al. 2021). Our high-level idea of SA (by the robot taking the human's perspective) aligns with the concept of perspective shifts in epistemic multi-agent planning – that extends Dynamic Epistemic Logic (DEL) (Engesser et al. 2017). However, unlike our first-order representation, which is used to maintain agents' distinct beliefs, DEL-based is rich and can model scenarios involving nested perspective-taking (Dissing and Bolander 2020). The concept of perspective shifts is expanded to provide a foundation for producing implicitly coordinated human-robot plans that do not require the agents to negotiate and commit to a joint policy at plan time (Bolander, Dissing, and Herrmann 2021). In specific scenarios, it produces HR policies that are not socially awkward, which is essentially the aim of HRI research. However, the work does not consider humans as uncontrollable agents like ours, so, from what we understand, extending their framework to handle the uncontrollability of human operators is not so clear.

A Quick Summary Putting in a nutshell different topics discussed in this section, we believe that to make an efficient decisional framework needs to integrate various concepts from these specific topics.

Considering the inherent advantages of specifications based on HTNs, we choose it for specifying the HR collaborative problem compactly. The framework to be extended based on our problem specification choice is HATP/EHDA that extends the HATP line of work.

First, to the best of our knowledge, existing approaches like the HATP/EHDA's solver and epistemic planners, despite modeling distinct agent beliefs, do not provide a formal way to manage their evolution when a system plans for a robot while estimating and emulating humans' behavior to achieve a task (shared explicitly), more specifically when they act concurrently — which we intend to support in the next version. Second, the HATP/EHDA's solver relies on cumbersome and domain-specific modeling techniques to update the agent's belief and to also align relevant belief divergences. However, our new approach proposes to both maintain agents' belief (*Situation Assessment* processes) and handle relevant divergences (via *communication* or *delay*) in a principled way within the scheme of the planner, not in the actions' description as in the case of DEL. It will be made clear in later sections.

3 The Underlying Architecture

We briefly describe the HATP/EHDA framework and discuss its ability to capture a broad class of HR collaborative planning scenarios (Buisan et al. 2022). It comprises a *dual* HTN-based task specification model. Based on certain realistic assumptions w.r.t. human operators, it plans for the robot to act in the presence of a human agent even when they do not have to achieve a shared task in the beginning. However, it can ask the human for occasional help to accomplish its task or manage the creation of shared tasks, can handle human reactions modeled explicitly via triggers, etc. Procedural Reasoning System (PRS) is used for supervising and controlling, reports triggers (Ingrand et al. 1996).

Next, we will briefly elaborate on important concepts needed to compactly describe HATP/EHDA.

3.1 Hierarchical Task Networks (HTNs)

Generalized basic terminologies and definitions related to HTNs are presented, e.g. the model, problem definition, and its solution. For more details on how an abstract task is decomposed with the help of available methods and constraints like *precedence*, *before*, and *after* are managed post decomposition, refer to (Ghallab, Nau, and Traverso 2004).

Definition 1. ***(Task Network.)*** *A task network is a 2-tuple $tn = (U, C)$, where $u \in U$ is a task node, while t_u is the task associated to this node. C is the set of constraints that includes strict (partial) orderings between task nodes, variable binding constraints, etc. If $\forall u \in U$, t_u is a primitive task, the task network is primitive; otherwise, it is non-primitive.*

Definition 2. ***(HTN Planning Problem.)*** *The HTN planning problem is a 3-tuple $\mathcal{P} = (s_0, tn_0, D)$ where s_0 is the initial belief state (the ground truth), tn_0 is the initial task network, and D is the HTN planning domain which consists of a set of tasks and methods.*

Definition 3. ***(Domain.)*** *A domain is a 2-tuple $D = (O, M)$ where O is the set of operators and M is the set of methods. An operator $o \in O$ is a primitive task described as $o = (head(o), pre(o), eff(o))$, which corresponds to the operator name (and associated grounded parameters), its precondition, and its effect, respectively.*

Definition 4. ***(HTN Solution Plan.)*** *A sequence of primitive actions $\pi = (o_1, o_2, o_3..., o_k)$, s.t., $\forall o_i, o_i \in O$, is a solution plan for the HTN planning problem $\mathcal{P} = (s_0, tn_0, D)$ iff there exists a primitive decomposition tn_p (of the initial task network tn_0), and π is an instance of it.*

3.2 Important Assumptions

First, let us list all the important assumptions the underlying architecture makes with respect to human operators.

- Humans and robots are not equal. Still, they often choose to collaborate to achieve a (shared) task.
- Humans are uncontrollable agents, their behavior is estimated and emulated. They can be cooperative and rational, but their association, computational capabilities, and tolerance of the shared task vary.
- We have access to the human task/action model describing their capabilities, world dynamics, and their understanding of common ground. This model is available to the robot, which influences its decision-making.
- At this stage, the robot maintains its own belief about the physical world, and a separate belief state to estimate what humans believe by taking their perspective. In the simplest case, both belief states align with the ground truth during planning.

3.3 The HATP/EHDA Framework

We define the general HATP problem for the human-robot team when they have a task to achieve. Here the joint task network captures a fully ordered shared goal and/or the agents' own goals, which is to be fully decomposed.

Definition 5. ***(Human-Aware Task Planning Problem.)*** *The HATP problem, which extends HTN specifications, is a 3-tuple $\mathcal{P}_{rh} = (\langle s_0^r, s_0^h \rangle, \langle tn_0^r, tn_0^h \rangle, \langle D_r, D_h \rangle)$ where s_0^r (i.e., s_0) is the initial belief state of the robot (also the ground truth), while s_0^h is the belief state human beings with which can contain facts do not hold in s_0. Here, tn_0^r is the initial task network that the robot has to solve, similarly tn_0^h —for the human. And D_r represents the domain available for the robot containing its operators and methods, and similarly, D_h represents the domain available for the human.*

The basic underlying structure manipulated by the solver is two *agent* models: the *human* and the *robot*. Each model has its own belief state, action model (HTN), task network, plan, and triggers. The HATP/EHDA existing planning scheme uses agents' action models and beliefs to decompose the agents' task network into legal primitive components. Decomposition updates the current network by inserting new (non) primitive tasks and additional constraints such that the single-agent process is generalized for the two-agent scenario. While doing so, it also updates the belief state of each agent.

Without loss of generality, the planning scheme assumes that a single agent decides to act at a time and, also "which action to execute?", e.g., add salt, cook pasta, and delay, similarly followed by other agents. It uses specific actions to synchronize agents' plans. *IDLE* is inserted into the agent's plan when its task network is empty, and *WAIT* when it does not have regular applicable actions. First, the framework builds the whole search space by considering all possible, feasible decompositions. Then, it can adapt off-the-shelf search algorithms like the well-known algorithms based on A^* and AO^*. Moreover, it considers social cost, plan legibility, and acceptability to search for the robot's policy. Note that our idea is to build an implicitly coordinated (joint-) plan for the agents, such that the planner only builds the robot's policy while estimating and emulating human behaviors w.r.t. the shared task.

Definition 6. ***((Implicitly Coordinated) Joint Solution.)*** *The solution for $\mathcal{P}_{rh}$, is represented as a tree, i.e. $G = (V, E)$. Each vertex ($v \in V$) represents the robot's belief state, starting from the initial belief. Each edge ($e \in E$) represents a primitive task that is either a robot's action o^r, or a human's estimated and emulated action o^h. G gets branched on the possible choices (o^h_1, o^h_2, ..., o^h_m).*

(Considering G a plan tree) each of its branches — from the root to a leaf node — is a sequence of primitive actions. Say, if $\pi = (o^r_1, o^h_2, o^r_3, ..., o^h_{k-1}, o^r_k)$ is a branch trace, it must satisfy all the solution conditions of $\mathcal{P}_{rh}$. Here, each o^h_i represents a choice, often out of several, the human could make. This factor is crucial and decides the robot's execution policy.

Although it is a limitation of the underlying framework, the assumption of a single agent acting at a time comes up with advantages. As humans are uncontrollable, one could think of possible joint actions (the humans' next predicted action could also be *delay* or *leave* among others), this assumption often enables the scheme to explore a broader search space. However, most importantly, our main contribution is agnostic to this limitation of the underlying framework.

In this work, we realize our contributions on top of HATP/EHDA. So, in principle, we tackle the same high-level problem as in Definition 5, and generate a solution similar to that in Definition 6. However, we have enriched the problem description to capture the observability properties of the environment. Moreover, to make the exposition easier, we have adapted the agents' state representation and the definition of the (implicitly coordinated) joint solution. We describe each of them in the coming sections.

4 Augmented Problem Specifications

We consider a *classical planning domain* (*state-transition system*) $\Sigma = (S, A, \gamma)$, s.t., S is a finite set of states in which the system may be, A is a finite set of actions that the actors may perform, $\gamma : S \times A \rightarrow S$ is a state-transition function. Each state $s \in S$ is a description of the properties of various objects in the planner's environment (Ghallab, Nau, and Traverso 2004).

To represent the objects and their properties, we will use two sets B and X: B is a set of names and mathematical constants representing the properties of all objects. X is a set of syntactic terms called state variables, s.t. the value of each $x \in X$ depends solely on the state s.

A *state-variable* over B is a syntactic term $x = sv(b_1, ..., b_k)$, where sv is a symbol called the state variable's name, and each b_i is a member of B and a parameter of x. Each state variable x has a range, *Range*$(x) \subseteq B$, which is the set of all possible values for x.

Here is the description of the sets B and X for the collaborative cooking example:

$$\begin{aligned}&B = Agents \cup Objects \cup Places \cup Booleans \cup \{\mathsf{nil}\}\\ &\quad Agents = \{\mathsf{R}, \mathsf{H}\} \ \backslash\backslash\ \mathsf{R} : robot,\ \mathsf{H} : human\\ &\quad Objects = \{\mathsf{salt}, \mathsf{pasta}, \mathsf{counter}\}\\ &\quad Places = \{\mathsf{kitchen}, \mathsf{room}\}\\ &\quad Booleans = \{\mathsf{true}, \mathsf{false}\}\end{aligned}$$

$$X = \{at(e), saltIn, stoveOn, counterClean \mid e \in Agents \cup Objects\}$$

$$\begin{aligned}&Range(SaltIn \mid StoveOn \mid CounterClean) = Booleans\\ &Range(at(R \mid H \mid pasta)) = Places\\ &Range(at(salt \mid counter)) = \{\mathsf{kitchen}\}\end{aligned}$$

A *variable value assignment* function over X is a function val that maps each $x_i \in X$ into a value $z_i \in$ *Range*(x_i). With $X = \{x_1, ..., x_n\}$, we will often write this function as a set of assertions: $val = \{x_1 = z_1, \ldots, x_n = z_n\}$.

A *variable observability assignment* function over X is a function *obs* that maps each $x_i \in X$ into an observability type $t_i \in \{\texttt{OBS}, \texttt{INF}\}$: *obs* $= \{(x_1, t_1), \ldots, (x_n, t_n)\}$. Respectively, when *obs*$(x_i) = \texttt{OBS}|\texttt{INF}$ then x_i is said to be *observable* | *inferable* in the state s_i.

A *variable location assignment* function over X is a function *loc* that maps each $x_i \in X$ into a $l_i \in Places \cup \{\texttt{nil}\}$: *loc* $= \{(x_1, l_1), ..., (x_n, l_n)\}$. $Places \subseteq B$ captures a group of constant symbols such that each member is a predefined area in the environment. Agents are always either "situated" in a place or moving between two places. More details about how the environment should be divided into places will be given shortly.

The intuition behind these function definitions is to later be able to define two Situation Assessment processes that will maintain the evolution of estimated human beliefs by reasoning on what the human should be able to observe instead of the action effect. These processes are described in the next section.

A *state* $s_i \in S$ is a 6-tuple composed of 4 functions over X and 2 task networks (agendas) s.t. $s_i = (val_i, val^H_i, obs_i, loc_i, tn^R_i, tn^H_i)$. The state of the world from the perspective of the robot is captured by the variable value assignment function val_i, sometimes noted as val^R_i. Similarly, val^H_i represents the estimation of val_i in the perspective of the human, also referred to as the estimated human beliefs. Hence, $\forall s_i \in S$, each $x_j \in X$ is mapped to two *values* (robot perspective and estimation of human's beliefs), an *observability type*, and a *place*. We say that a state $s_i \in S$ contains *false beliefs*, or *belief divergences*, if $\exists x_j \in X, val^H_i(x_j) \neq val^R_i(x_j)$.

For our example, the initial state s_0 would be as follow:

$$s_0 = \{val_0,\ val^H_0,\ obs_0,\ loc_0,\ tn^R_0,\ tn^H_0\}$$

$$\begin{aligned}val_0 = val^H_0 = \{&at(\mathsf{R}) = \mathsf{kitchen}, at(\mathsf{H}) = \mathsf{kitchen},\\ &at(\mathsf{pasta}) = \mathsf{room}, saltIn = \mathsf{F}, stoveOn = \mathsf{F}\}\end{aligned}$$

$$obs_0 = \{(at(e), \texttt{OBS}), (saltIn, \texttt{INF}), (stoveOn, \texttt{OBS}) \mid e \in Agents \cup Objects\}$$

$$loc_0 = \{(at(e), val_0(e)), (saltIn, \textsf{kitchen}), (stoveOn, \textsf{kitchen}) \mid e \in Agents \cup Objects\}$$

$$tn_0^R = \{CookPasta, CleanCounter\}$$

$$tn_0^H = \{CookPasta\}$$

An action is a tuple $\alpha = (head(\alpha), pre(\alpha), eff(\alpha))$ where $head(\alpha)$ is a syntactic expression of the form $act(z_1, ..., z_k)$ where act is a symbol called the *action name* and $z_1, ..., z_k$ are variables called parameters. $pre(\alpha) = \{p_1, ..., p_m\}$ is a set of preconditions, each of which is a literal. And $eff(\alpha) = \{e_1, ..., e_n\}$ is a set of effects, each of which is an expression of the form: $sv(t_1, ..., t_j) \leftarrow t_0$ with t_0 being either the value to assign to the state variable $sv(t_1, ..., t_j)$ or a new location/place for the state variable. We note $agt(\alpha)$ the agent performing the action α.

To estimate the next possible actions that an agent $\varphi \in Agents$ is likely to perform in a state $s_i \in S$, we proceed in the same way as in (Buisan et al. 2022). We refine the agent's agenda tn_φ based on its belief val_i^φ and obtain a *refinement* as follows $ref(tn_i^\varphi, val_i^\varphi) = \{(a_1, tn_1), ..., (a_j, tn_j)\}$. A *refinement* contains a tuple for each estimated possible action a_j and the associated new agenda tn_j after being refined.

In our cooking example, we obtain the following refinement if the starting agent is the human:
$ref(tn_0^H, val_0^H) = \{(add\_salt(), tn_1), (move\_to(\textsf{kitchen}), tn_2)\}$

5 State Transitions and Belief Updates

We now describe how a new state is generated and more precisely how the estimated human beliefs are updated according to our observability models. A transition occurs only if an action a is applicable in a state s_i, i.e. $\gamma(s_i, a) = s_{i+1}$.

Our new formalism provides support only for agents who either know the truth or have a false belief. Moreover, we do not consider cases where the robot's beliefs can diverge, too. Hence, regardless of being co-present, the robot's beliefs are always updated with the action's effects assuming the human only makes deterministic moves when not being observed. (To model, the latter requires richer syntax and semantics than ours.) Finally, $\forall x \in X$, we always have,

$$val_{i+1}(x) = \begin{cases} w, & \text{if } x \leftarrow w \in eff(a) \\ val_i(x), & \text{otherwise} \end{cases}$$

The place associated with a state variable can be modified by the action's effect but, here, we assume that the observability type of each fact is constant during the task. So, $\forall x \in X$,

$$obs_{i+1}(x) = obs_i(x)$$

$$loc_{i+1}(x) = \begin{cases} l, & \text{if } x \leftarrow l \in eff(a) \\ loc_i(x), & \text{otherwise} \end{cases}$$

The new agenda of each agent (tn_{i+1}^R, tn_{i+1}^H) are created by the HTN refinement algorithm, and thus, they are directly retrieved from the obtained refinement. This refinement decomposes abstract tasks in the task network until the first task is a primitive action. To do so, every applicable method is applied leading to a set of possible actions (and refined task networks).

The new estimated human belief val_{i+1}^H is the two-step result of our Situation Assessment processes that models the real-time sensing and reasoning capabilities of the human about their surroundings.

First, let us define the notions of *co-presence* and *co-location* which will be key to maintain the evolution of agents' beliefs as planning progresses.

Definition 7. ***(Co-presence & Co-location.)*** *In a state $s_i \in S$, two agents, φ_1 and φ_2, are considered to be co-present if $val_i(at(\varphi_1)) = val_i(at(\varphi_2))$. This relation is noted $\varphi_1 \curlywedge_i \varphi_2$ in the rest of the paper. Similarly, we say that an agent φ_1 is co-located with a state variable $x \in X$ if $val_i(at(\varphi_1)) = loc_i(x)$, noted $\varphi_1 \curlywedge_i x$.*

Now we can define two SA processes that will maintain human beliefs based on their estimated evolution of perspectives.

Definition 8. ***(Inference Process.)*** *An agent observes the execution of an action by being either co-present with the acting agent,or by being the acting agent. If so, the agent infers the new values of every state variable present in the action's effects.*

Based on the above definition, the human's beliefs are updated as follows when action a is executed in state s_i,

$$val'^H_{i+1}(x) = \begin{cases} w, & \text{if } x \leftarrow w \in eff(a) \text{ and} \\ & (H = agt(a) \text{ or } H \curlywedge_i agt(a) \\ & \text{ or } H \curlywedge_{i+1} agt(a)) \\ val_i^H(x), & \text{otherwise} \end{cases}$$

To change its *place* in the environment, agents would use a dedicated *"move"* action, such that its effect only updates the agent's location.

Definition 9. ***(Observation Process.)*** *An agent observes its surroundings and assesses the exact value of each state variable located in the same place (i.e., each state variable the agent is co-located with).*

After applying the effects of an action to obtain val_{i+1} and the human beliefs val'^H_{i+1} (using the inference process), the observation process is executed. It updates again the estimated human beliefs with the facts currently observable by the human and provides fully updated human beliefs to store in the state s_{i+1}, $\forall x \in X$:

$$val_{i+1}^H(x) = \begin{cases} val_{i+1}(x), & \text{if } H \curlywedge_{i+1} x \text{ and} \\ & obs_{i+1}(x) = \texttt{OBS} \\ val'^H_{i+1}(x), & \text{otherwise} \end{cases}$$

Note that before starting the planning process, the observation process is executed once on the initial state s_0. This allows us to potentially correct the estimated human beliefs with the facts the human should initially be able to observe.

The definition of the set $Places$, i.e. how the environment is divided into different *places*, is guided by the shape of our state transition function. Hence, a $place \in Places$ is an area in the environment such that, when situated in it, agents are aware of each other's activity and they can assess every observable fact located in it. (In literature, they are also sometimes modeled and dealt with as *controlled* observation as in (Kulkarni, Srivastava, and Kambhampati 2021).)

Note that unlike in DEL (Bolander, Dissing, and Herrmann 2021), our formalism is simple and prevents us from expressing agents being *uncertain* about a fact. In line with the classical closed-world assumptions, agents either know the truth or maintain a false belief w.r.t. the ground truth. We consider a straightforward scenario in which the human is *"unaware"* of non-observed changes in the environment. This results in estimated false human beliefs, helping to detect whether a non-observed robot action can disrupt a seamless collaboration.

6 Relevant False Belief: Detection & Solution

In this section, we explain our procedure to detect *when* a false human belief should be corrected and *how*.

6.1 Definition and Detection

The human and the robot carry individual distinct beliefs, while the two can be aligned, or diverging when the human has a false belief. To produce a legal solution plan the robot is fine with such false human beliefs unless they are qualified as *relevant* (Definition 10). In such cases, the relevant false belief needs to be tackled.

Definition 10. *A **relevant false belief** is a false belief that influences the next action(s) the human is likely to perform w.r.t. the shared task, either in terms of number, name, parameters, or effects. This can be written as follows: A state s_i contains a relevant false belief if either (1) or (2) is true:*

$$ref(tn_i^H, val_i^H) \neq ref(tn_i^H, val_i^R) \tag{1}$$

$$\{\gamma(s_i, a) \mid \forall a \in ref(tn_i^H, val_i^H)\} \neq \{\gamma(s_i, a) \mid \forall a \in ref(tn_i^H, val_i^R)\} \tag{2}$$

In this work, we consider that as soon as a false belief has an effect on human action it should be tackled. An interesting future work could be to check in a principled way the overall positive and detrimental impacts of this false belief on the shared task. But it is out of the scope of this work.

6.2 Solved with communication

A state containing a false human belief marked as *relevant* must be handled. The first way to do it is by planning communication actions such that the robot communicates only the required facts to the human. This allows to correct false human beliefs that are relevant, but false beliefs that are *"non-relevant"* will remain.

Modeling Communication Actions We propose a generic communication action schema (ca) in this context. An agent φ_i can *communicate* an assertion $x = z$ (with $x \in X$ and $z \in \text{Range}(x)$) *via* the action $ca_{\varphi_i,\varphi_j}(x, z)$ if $val^{\varphi_i}(x) = z$ and $val^{\varphi_j}(x) \neq z$. The effect of $ca_{\varphi_i,\varphi_j}(x, z)$ corresponds to $val^{\varphi_j}(x) \leftarrow z$. Such actions are considered equally costly and instantaneous.

Communicate Only the Required Facts Definition 10 indicates if there is at least one diverging state variable in the human beliefs causing adverse effects, but without identifying which one(s). Hence, we explain a subroutine below with the three steps, describing how we first identify the pertinent state variables to align, and then how the corresponding communication actions are created and inserted into the robot's plan.

1. *Store* each state variable whose value differs in the human beliefs from the robot beliefs: $X_{diff} = \{x \mid x \in X, val_i^H(x) \neq val_i^R(x)\}$.
2. *Build*, for each stored state variable $x \in X_{diff}$, a communication action $ca_{R,H}(x, val_i^R(x))$, all stored in a set CA_{diff}.
3. *(Breadth-First Search.)* The *source* is s_i. Applying each $ca \in CA_{diff}$ generates a new state by aligning *exactly* one state variable in the human beliefs s.t. $s_i' = \gamma(s_i, ca)$. The search continues until the first state s_i' selected to expand doesn't contain a relevant false belief. The communication actions used from the root until this selected state are *retrieved* in a set CA.

Once the above subroutine finishes, the retrieved communication actions in the set $CA = \{ca_{R,H}(x_1, val_i^R(x_1)), ..., ca_{R,H}(x_j, val_i^R(x_j))\}$ must be inserted in the plan for belief alignment. Thus, Definition 6 is redefined to be sound w.r.t. our approach. An edge can now either be a human action o^h or a robot action o^r with a set of communication action CA. At each step, humans perform *Observation*, while the robot executes each communication action $ca \in CA$, while the human's belief *updates instantaneously*.

The set CA is inserted before the diverging human actions and after the closest state where agents are co-present. But it could be interesting to reason with a better plan evaluation system to find the best place to insert this set.

6.3 Solved by delaying action

So far we relied on communication, but depending on the environment (e.g. noisy), communication can be cognitively demanding. Thus, when the relevant false belief is due to a non-observed robot action, we propose to also consider implicit communication by postponing the pertinent robot action until the human is estimated to be observing its execution. This prevents false beliefs from even occurring.

First, a branch using communication is explored and the state variables concerned by the relevant false beliefs are retrieved (through all $ca \in CA$). Then we check if the divergence is produced by a non-observed action. For now, it is done by checking if the relevant divergence concerns only one inferrable state variable and if it was not present in the initial state. After we identify which action creates the divergence by sequential regressing the current branch/trace. Hence, we can identify when the relevant divergence appears and which action should be delayed. Once identified, we create another branch in the plan just before the identified action. In this new branch, DELAY actions are inserted in the robot's plan until the human is co-present. When the human is co-present again, the identified action is inserted and observed by the human. Then the nominal planning process is resumed.

7 Formal Properties

Our solver shares a similar high-level procedure as the HATP/EHDA solver. However, when discussing its soundness and completeness, dependent on the soundness and completeness of the underlying mechanism, we must distinguish between the problem specifications used by these solvers.

In the worst case, our specifications would pick each assertion $x = z, x \in X$ and $z \in Range(x)$, and generate a new set of primitive propositions for every possible combination of $|\{\texttt{OBS}, \texttt{INF}\}| \times |Places|$. So, the new encoding size for primitive propositions is worst-case *linear* in the size of X.

To support the HATP/EHDA solver's assumptions (Sec. 3.2) with the new specification, a possible encoding is to consider that every $x \in X$ is *inferrable* and is located in one unique dummy place l_{du} s.t. $Places = \{l_{du}\}$. This way, agents are always co-present with each other and co-located with every state variable, making the inference process always update the beliefs of every agent with the action effects. Hence, the new specification captures everything that can be formulated with the HATP/EHDA solver but provides latitude to model more realistic problems.

The following proofs hold under the following conditions: I.e., the high-level assumptions HATP/EHDA makes while supporting turn-taking planning for achieving a shared task tn_0 in $\mathcal{P}_{rh}$, s.t. tn_0 takes a form of a sequential joint-task network. Our paper explicitly considers this setting.

Theorem 7.1. ***(Soundness.)*** *The new solver is sound.*

Proof. (Sketch.) Following Definition 6, we show that the generated robot's policy is executable online (as it is internally coordinated with human possible behaviors): For a branch, it is guaranteed that it meets the *solution conditions*. Moreover, each precondition for an agent's action is achieved in earlier time stamps by either: its own action, *inferred* while observing another agent's actions, another agent *communicated*, or it is *assessed* by observing a situation and reasoning. Of course, another agent's action supplying the precondition, an attribute (in OBS) and its value, cannot be destroyed. Hence this branch/trace is executable.

Finally, since humans can make any choices during execution, and are unknown upfront, any of its branches, by the above argument, will be executed. Implies, the joint solution plan is executable. □

Theorem 7.2. ***(Completeness.)*** *The new planning algorithm is complete, provided the underlying mechanism can exhaustively generate all possible plan elaborations.*

Proof. (Sketch.) Suppose a problem based on our formalism has a solution, say, an (internally coordinated) joint solution tree, τ, and is sound. From τ, remove all the communication and situation assessment steps, and say it generates τ'. Now, relax the original problem, considering all the agents are always co-present and making all the propositions *inferable* everywhere. Technically, the solver will generate τ' for this new problem when it exhaustively generates the whole search space (of course, it may contain redundant actions). We assure it believing in the underlying mechanism and that there is at least one solution (τ) for the original problem. As a result, it is trivial to visualize that τ' is always extendable to generate τ w.r.t. the original problem. After the execution of each step, one can employ the state transition and belief update mechanism described earlier, checking for belief divergences and using communication actions if needed. □

8 Evaluation

To the best of our knowledge, we are not aware of any implemented planning system that considers humans as uncontrollable with distinct beliefs, apart from our previous work: the HATP/EHDA solver. We decided to use the latter as a baseline to help present the results of our approach on three *novel* planning domains that we describe below.

Cooking Pasta Domain: The running example corresponds to a specific problem in this domain. In fact, agents and pasta can initially either be in the kitchen or in the adjacent room, the stove might be on or off and there might be salt or not in the water. In the results, we will focus on the following three state variables from X. Both $StoveOn$ (OBS) and $SaltIn$ (INF) are relevant to the human, unlike $Clean$ (INF) which only concerns the robot.

Preparing Box Domain: A box with a sticker on it and filled with a fixed number of balls is considered prepared and needs to be sent. Both agents can *fill* the box with balls from a bucket, while only the robot can *paste* a sticker and only the human can *send* the box. The bucket can run out of balls, so when one ball is left, the human *moves* to another room to *grab* more balls and *refill* it. The number of balls in the box is *inferable*, while all other variables are *observable*. In the following, three boxes have been considered.

Car Maintenance Domain: The washer fluid (OBS) and engine oil (INF) levels have to be *full* before *storing* the oil gallon in the cabinet (INF). Only the robot can *refill* both the tanks and store the gallon while situated at *Front* of the car. *Front-left* and *Front-right* headlights have to be *checked* and a light-bulb has to be *replaced* at *Rear*. Only the human can check and replace lights, and they can start with either of these two tasks. Both agents start at *Front*. The car's hood needs to be *closed* by the human at last.

8.1 Qualitative Analysis

We discuss in detail the plans obtained with our approach to a problem from the first cooking domain. The problem corresponds to the description given in the introduction. I.e., there is no initial human false belief, agents both start in the kitchen, the pasta is in the adjacent room, the stove is off, and there is no salt in the water. The resulting plans are shown in figure 2 their detailed presentation explains how the approach works in practice. Since human is uncontrollable and has different possible actions, the plan branches and the robot's actions are different in each case.

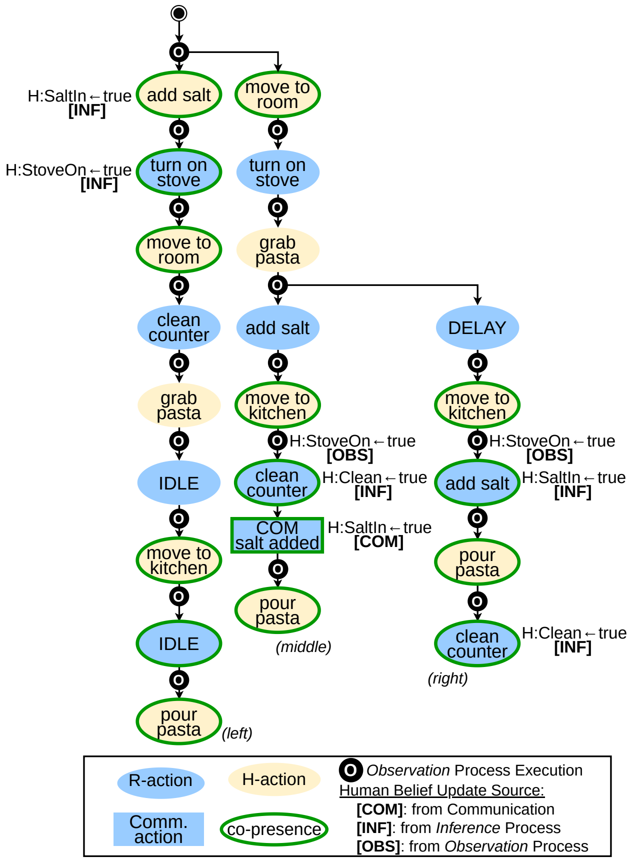

Figure 2: Plan obtained for the cooking scenario. 3 branches. Left: The human starts by adding salt. The only false belief is about *"CounterClean"* which is not relevant for the human agent, hence no comm is added. Middle: While the human is away the robot turns on the stove and adds salt, creating 2 false beliefs. Once the human agent is back we estimate that it will be able to assess the observable fact *"StoveOn"* but not the *"SaltIn"* one. Since the human agent might add salt again due to this false belief, it is relevant and fixed with a communication action. Right: The relevant false belief about *"SaltIn"* is avoided by delaying the robot's action until the human is co-present. Hence no communication is required.

In (*left*) the human first adds salt and then the robot turns on the stove. In both cases, thanks to the inference process, we estimate that the human will be aware of both facts about the salt (*acting*) and the stove (*co-present*). Then while the human is away to fetch the pasta, the robot cleans the counter and since the human isn't co-present their beliefs aren't updated, containing now a false belief. Once back, since *CleanCounter* is not *observable* the observation process does nothing and the false belief remains. However, this false belief doesn't affect human actions (non-relevant), hence, there is no need to align human beliefs.

In (*middle* and *right*) the human first fetches the pasta by leaving the kitchen. Let's first focus on the (*middle*) trace. The robot turns on the stove and adds salt while the human is away, creating two false beliefs. When returning to the kitchen, the observation process updates the human beliefs with the observable facts located in the kitchen. This fixes the false belief about *StoveOn*. The robot then cleans the counter, observed by the human. However, without communication, the human's next action will be either "add salt" or "ask the robot", but considering the ground truth the human could directly pour the pasta. Hence, the false belief on *SaltIn* is relevant and has to be corrected. To do so a communication is inserted in the robot's plan and a "delay" branch is created (*right*). In this delaying branch, the robot delays the add salt action until the human is co-present in order to make it observed (inference process) by the agent. In addition to this implicit communication, like in (*middle*), the human assesses that the stove is on and hence can directly pour the pasta.

Our approach automatically maintains human beliefs and is able to avoid failures due to initial false human beliefs concerning observable facts, as depicted in figure 3.

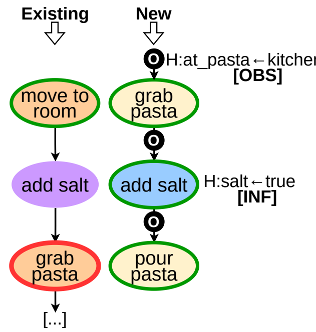

Figure 3: An example where an initial false belief about the pasta location ($val_0(at(Pasta)) = kitchen \neq val_0^H(at(Pasta) = room)$ leads to a non-applicable action (grab pasta) with HATP/EHDA. Our approach, through the observation process, automatically updates human beliefs and leads to a valid plan.

Domain	**HATP/EHDA** *S I.Div.B.*	*S*	**Only Comm** *Comm*	**With Delay** *Comm*
Cooking	6.9%	18.6%	69.5%	65.2%
Box	14.3%	25.0%	79.7%	75.0%
Car	0.0%	12.5%	68.8%	64.1%
Average	7.1%	18.7%	72.6%	68.1%

Table 1: Results were obtained for 512 problems for each of the three domains. For HATP/EHDA, its success ratio with problems with initially diverging beliefs and the overall success ratio are shown respectively under $SI.Div.B$ and S. The ratio of plans including a communication action ($Comm$) is shown for our approach when using only communication ($OnlyComm$) and when only using Delay action ($WithDelay$).

8.2 Experimental Results and Analysis

In each domain, the actions and tasks remain the same. So here, a problem is defined by a starting agent (R or H) and a pair of initial beliefs (val_0^R, val_0^H). Initial ground truth

($val_0 \Leftrightarrow val_0^R$) is defined by setting each state variable to an initial value. But, 5 selected state variables can be set to 2 possible values instead of 1. And among these selected ones, 3 can diverge in human belief. This generates 256 pairs of initial beliefs where 12.5% of them include initially aligned beliefs. Then, considering the starting agent, we obtain 512 problems for each domain. Each of the 1536 generated problems has been solved by HATP/EHDA, *our approach* using first *only communication* and then *delay*. The obtained quantitative results appear in Table 1.

The HATP/EHDA solver always finds legal plans when dealing with aligned beliefs. But, as expected, the low success rate for initially diverging beliefs ($SI.Div.B = 7.1\%$) reflects how the HATP/EHDA solver poorly handles belief divergences without specifically designed action models. Since *our approach* always finds legal plans, we can say it solves a broader class of problems.

Furthermore, considering the initially diverging beliefs and the divergences created along the planning process, more than 87.5% of all problems involve belief divergences. However, when using only verbal communication, only 72.6% of the generated plans include communication actions. This means that *our approach* communicates only when necessary, and not systematically. The amount of communication is even reduced to 68.1% when delaying actions. In the latter case, only the delayed branch that doesn't induce the human to wait is kept.

9 Discussion

Currently, the underlying HATP/EHDA framework allows just a single agent to execute a *"real"* action at a time. However, a post-process can allow the execution of actions concurrently (Crosby, Jonsson, and Rovatsos 2014), however, note that the domain modeler has modeled $\mathcal{P}_{rh}$ as a sequential joint task. Since parallelism is not taken into account during modeling and/or planning it can limit the possible parallelized executions. We are working on extending this framework to systematically allow planning with concurrent actions, in line with (Shekhar and Brafman 2020).

We believe our modeling-level SA proposals could fit in any other planning approach framing multi-party systems having one controllable agent while can only hypothesize remaining agents' behaviors (e.g., human-centered AI).

Agents' SA models only assess new facts and cannot refute a false belief. E.g., assume the human *wrongly* believes that the pasta is in *Kitchen*. The SA does not help refute this false belief, even if the human is in *Kitchen*. The reason is that *NotAt(Pasta)* is not modeled explicitly as a state variable. However, such issues do not affect the completeness and, if necessary, our planning approach *tackles* such cases as relevant false beliefs.

We discussed earlier that DEL is more expressive and flexible in this aspect and can handle knowledge uncertainty, however, it requires an augmented action schema to accurately maintain each agent's beliefs. Think of a specification for *"move"* action manually listing all the environmental facts to be observed by an agent for managing their beliefs. In our case, it is implicitly maintained within a state. We can consider running a set of rules (e.g., *graph-based ontology*) to bring new interesting facts in the state based on a set of known facts. We believe that this aspect opens up new possibilities in the future to integrate human-aware collaborative planning and ontology.

10 Conclusion

We propose an extension to a Human-Aware Task Planner called HATP/EHDA. The planner plans and implicitly coordinates the robot's actions with all estimated possible human (uncontrollable) behaviors that are then emulated by the solver to generate a new state. Our extension and contribution are, first, to integrate a *Situation Assessment* based reasoning system in the planner. This allows for maintaining distinct agents' beliefs based on what they can/should observe. Compared to existing epistemic planners, this simplifies the action descriptions by focusing on their effects on the world, and not how they influence each agent's beliefs. In addition, we propose to detect false human beliefs and tackle only the ones that have a negative impact. First, we propose minimal and proactive communication. Second, if the false belief is due to a non-observed robot action, we propose an implicit communication by postponing the non-observed robot action until the human is co-present to observe it.

References

Alami, R.; Clodic, A.; Montreuil, V.; Sisbot, E. A.; and Chatila, R. 2006. Toward Human-Aware Robot Task Planning. In *AAAI spring symposium: to boldly go where no human-robot team has gone before*.

Alili, S.; Alami, R.; and Montreuil, V. 2009. *A Task Planner for an Autonomous Social Robot*, 335–344. Berlin: Springer Berlin Heidelberg. ISBN 978-3-642-00644-9.

Berlin, M.; Gray, J.; Thomaz, A. L.; and Breazeal, C. 2006. Perspective taking: An organizing principle for learning in human-robot interaction. In *Proc. of AAAI*.

Bolander, T.; and Andersen, M. B. 2011. Epistemic planning for single and multi-agent systems. *J. Appl. Non Class. Logics*, 21(1): 9–34.

Bolander, T.; Dissing, L.; and Herrmann, N. 2021. DEL-based Epistemic Planning for Human-Robot Collaboration: Theory and Implementation. In *Proc. of KR*.

Buisan, G.; Favier, A.; Mayima, A.; and Alami, R. 2022. HATP/EHDA: A Robot Task Planner Anticipating and Eliciting Human Decisions and Actions. In *Proc. of ICRA*.

Chakraborti, T.; Briggs, G.; Talamadupula, K.; Zhang, Y.; Scheutz, M.; Smith, D. E.; and Kambhampati, S. 2015. Planning for serendipity. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015, Hamburg, Germany, September 28 - October 2, 2015*, 5300–5306. IEEE.

Cirillo, M.; Karlsson, L.; and Saffiotti, A. 2009. A Human-Aware Robot Task Planner. In *Proc. of ICAPS 2009*.

Clodic, A.; Pacherie, E.; Alami, R.; and Chatila, R. 2017. *Key Elements for Human-Robot Joint Action*, 159–177. Cham: Springer International Publishing. ISBN 978-3-319-53133-5.

Cooper, M. C.; Herzig, A.; Maffre, F.; Maris, F.; Perrotin, E.; and Régnier, P. 2021. A lightweight epistemic logic and its application to planning. *Artif. Intell.*, 298: 103437.

Crosby, M.; Jonsson, A.; and Rovatsos, M. 2014. A Single-Agent Approach to Multiagent Planning. In *Proc. of ECAI*.

Darvish, K.; Simetti, E.; Mastrogiovanni, F.; and Casalino, G. 2021. A Hierarchical Architecture for Human-Robot Cooperation Processes. *IEEE Trans. Robotics*, 37(2): 567–586.

Devin, S.; and Alami, R. 2016. An implemented ToM to improve human-robot shared plans execution. In *HRI Proc.*

Dissing, L.; and Bolander, T. 2020. Implementing Theory of Mind on a Robot Using DEL. In *Proc. of IJCAI*.

Engesser, T.; Bolander, T.; Mattmüller, R.; and Nebel, B. 2017. Cooperative epistemic multi-agent planning for implicit coordination. In *Proc. of WS on EPTCS*.

Ghallab, M.; Nau, D. S.; and Traverso, P. 2004. *Automated Planning - Theory and Practice*. Elsevier.

Gombolay, M. C.; Jensen, R.; Stigile, J.; Son, S.; and Shah, J. A. 2016. Apprenticeship Scheduling: Learning to Schedule from Human Experts. In *Proc. of IJCAI*.

Hayes, B.; and Scassellati, B. 2016. Autonomously constructing hierarchical task networks for planning and human-robot collaboration. In *Proc. of ICRA*.

Hörger, M.; Kurniawati, H.; and Elfes, A. 2019. Multilevel Monte-Carlo for Solving POMDPs Online. In *Proc. of ISRR*.

Ingrand, F. F.; Chatila, R.; Alami, R.; and Robert, F. 1996. PRS: A high level supervision and control language for autonomous mobile robots. In *Proc. of ICRA*.

Jacob, M. G.; Li, Y.-T.; Akingba, G. A.; and Wachs, J. P. 2013. Collaboration with a robotic scrub nurse. *Communications of the ACM*, 56(5): 68–75.

Köckemann, U.; Pecora, F.; and Karlsson, L. 2014. Grandpa Hates Robots - Interaction Constraints for Planning in Inhabited Environments. In Brodley, C. E.; and Stone, P., eds., *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, 2293–2299. AAAI Press.

Koppula, H. S.; Jain, A.; and Saxena, A. 2016. Anticipatory planning for human-robot teams. In *Experimental robotics*, 453–470. Springer.

Kulkarni, A.; Srivastava, S.; and Kambhampati, S. 2021. Planning for Proactive Assistance in Environments with Partial Observability. *CoRR*, abs/2105.00525.

Lallement, R.; de Silva, L.; and Alami, R. 2018. HATP: hierarchical agent-based task planner. In *Proc. of AAMAS*.

Levine, S. J.; and Williams, B. C. 2014. Concurrent plan recognition & execution for human-robot teams. In *ICAPS*.

Nikolaidis, S.; Kwon, M.; Forlizzi, J.; and Srinivasa, S. 2018. Planning with verbal communication for human-robot collaboration. *ACM Transactions on HRI*, 7(3): 1–21.

Roncone, A.; Mangin, O.; and Scassellati, B. 2017. Transparent role assignment and task allocation in human robot collaboration. In *Proc. of ICRA*.

Sanelli, V.; Cashmore, M.; Magazzeni, D.; and Iocchi, L. 2017. Short-term human-robot interaction through conditional planning and execution. In *Proc. of ICAPS*.

Selvaggio, M.; Cognetti, M.; Nikolaidis, S.; Ivaldi, S.; and Siciliano, B. 2021. Autonomy in physical human-robot interaction: A brief survey. *IEEE Robotics Autom. Lett.*

Shekhar, S.; and Brafman, R. I. 2020. Representing and planning with interacting actions and privacy. *AIJ*, 278.

Shvo, M.; Klassen, T. Q.; and McIlraith, S. A. 2022. Resolving Misconceptions about the Plans of Agents via Theory of Mind. In *Proc. of ICAPS*.

Tellex, S.; Knepper, R. A.; Li, A.; Rus, D.; and Roy, N. 2014. Asking for Help Using Inverse Semantics. In *Proc. of RSS*.

Unhelkar, V. V.; Lasota, P. A.; Tyroller, Q.; Buhai, R.-D.; Marceau, L.; Deml, B.; and Shah, J. A. 2018. Human-aware robotic assistant for collaborative assembly: Integrating human motion prediction with planning in time. *IEEE Robotics and Automation Letters*, 3(3): 2394–2401.

Unhelkar, V. V.; Li, S.; and Shah, J. A. 2019. Semi-Supervised Learning of Decision-Making Models for Human-Robot Collaboration. In *Proc. of CoRL*.

Unhelkar, V. V.; Li, S.; and Shah, J. A. 2020. Decision-Making for Bidirectional Communication in Sequential Human-Robot Collaborative Tasks. In *Proc. of HRI*.

Unhelkar, V. V.; and Shah, J. A. 2019. Learning models of sequential decision-making with partial specification of agent behavior. In *Proc. of AAAI*.

Vats, S.; Kroemer, O.; and Likhachev, M. 2022. Synergistic Scheduling of Learning and Allocation of Tasks in Human-Robot Teams. In *Proc. of ICRA*.

Wimmer, H.; and Perner, J. 1983. Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children's understanding of deception. *Cognition*, 13(1): 103–128.