



**HAL**  
open science

# Models and Algorithms for Human-Aware Task Planning with Integrated Theory of Mind

Anthony Favier, Shashank Shekhar, Rachid Alami

► **To cite this version:**

Anthony Favier, Shashank Shekhar, Rachid Alami. Models and Algorithms for Human-Aware Task Planning with Integrated Theory of Mind. IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Aug 2023, Busan, South Korea. 10.1109/RO-MAN57019.2023.10309437 . hal-04210921

**HAL Id: hal-04210921**

**<https://laas.hal.science/hal-04210921>**

Submitted on 19 Sep 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Models and Algorithms for Human-Aware Task Planning with Integrated Theory of Mind

Anthony Favier<sup>1,2</sup>, Shashank Shekhar<sup>1</sup> and Rachid Alami<sup>1,2,\*</sup>

**Abstract**—It is essential for a collaborative robot to consider the 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* [1]. 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, discuss its effectiveness qualitatively, and show experimental results on three novel domains.

## I. INTRODUCTION

Human-Robot Collaboration (HRC) is a current research focus due to the growing number of robot-assisted applications [2]. Collaborative robots add clear value to real-world domains like household [3], workshops [4], or medical facilities [5].

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 scenario is depicted in Fig. 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

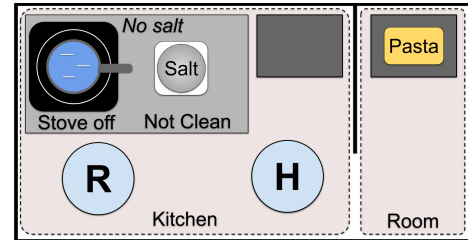


Fig. 1. Let us consider cooking pasta as a human-robot shared task. The robot has to turn on the stove (*stoveOn*) and clean the counter (*counterClean*), but the latter is not a part of the shared task. The human takes care of fetching the pasta while both agents can add salt into the water (*saltIn*). 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).

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 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 successfully turned on the stove 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 this fact. 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 [6]) already considers ToM when interacting with humans, but only during the task execution. Thus, the robot can only be reactive to human’s absence or inattention, even if they are predictable. We propose to bring such ToM considerations at the planning time. This makes the robot proactive and able to act differently to avoid the human missing necessary information, or to communicate when needed or in advance.

For seamless human-robot collaboration, we believe that

<sup>1</sup> LAAS-CNRS, Universite de Toulouse, CNRS, Toulouse, France

<sup>2</sup> Artificial and Natural Intelligence Toulouse Institute (ANITI)

\* Contacts: {anthony.favier, sshakhar, rachid.alami}@laas.fr

\*\*This work was partially supported by ANITI under grant agreement No: ANR-19-PI3A-0004

it is essential not to restrict human behaviors, and hence, we consider the human as an *uncontrollable* agent. Recent offline planning frameworks [7], [8] use human task models to estimate and handle uncontrollable human behaviors congruent to the shared task. 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 [7] which stands for **H**uman-**A**ware **T**ask **P**lanner **E**mulating **H**uman **D**ecisions and **A**ctions. We first enriched its task specifications to explicitly capture the observability properties of the environment. We rely on the notion of *co-presence*, and on *two* types of facts: First, those that can be observed directly in the environment, second, those that can only be inferred while attending the execution of an action. We then model so-called “*Situation Assessment*” (SA) processes to estimate *offline* the agents’ sensing and reasoning *run-time* capabilities about their surroundings. 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 *essential* to the human 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, we also try to postpone its execution until the human is anticipated to attend it. Such implicit communication 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 some background about the HATP/EHDA framework — briefly described in Section III. Section IV describes the formalism used in our planning scheme and how it captures the observability properties of the environment. Section V presents two situation assessment processes and how they are used to maintain the estimated human beliefs. Section VI explains how to detect relevant false human beliefs and how they are corrected. It is followed by Section VII discussing empirical evaluation, and showing both qualitative and quantitative results. The paper ends with discussion and conclusion sections.

## II. RELATED WORK

*a) 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 [9]. The framework given by [6] allows the robot to estimate at execution time, the mental state of the human, containing not only their beliefs but also their actions, goals, and plans. 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.

The framework proposed in [10] uses agents’ ToM. Here, an 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 [11]. 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 [12], 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 [13], [14], 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).

*b) Planning Approaches, Solution Plans, and Models:* Various task models have been realized in the Human-Robot (HR) collaborative planning context, e.g., hierarchical task networks (HTNs) [15], [16], POMDPs [8], [16], [3], AND/OR graphs [17], 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 [18], and the plan search occurs in a depth-first manner. In [19], 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 [3].

The HATP frameworks extending HTNs consider agents controllable [20], [15], while in [16], the framework considers planning at multiple abstraction levels (with a single HTN) with humans. 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 [12], [21], 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 investigated how to create a reasonable model of humans and how to obtain task knowledge, e.g., [22]. 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 robots’ decisions. Such models can be learned using conjugate task graphs, and to identify the task structure an aggregation algorithm can be used [23].

c) *Communication in HR Collaboration*: There is a line of work dealing with an explicit usage of communication actions in planning [24], [25], [16], [26], [3]. E.g., in [16], [3], the authors represent and plan with explicit communication actions, considering them as regular POMDP actions, such that execution policies contain them.

d) *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 [27]. 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) [28]. 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. In [29], 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. 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.

e) *A Quick Summary*: 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, which extends the HATP line of work. To the best of our knowledge, existing approaches like HATP/EHDA and epistemic planners, despite modeling distinct agents’ beliefs, do not provide a formal way to manage their evolution during planning while collaborating on an explicitly shared task. Moreover, these existing approaches rely on cumbersome and domain-specific modeling techniques, especially conditional action effect, to update the agent’s belief and to also align relevant belief divergences. Our new approach proposes to both maintain agents’ beliefs and handle relevant divergences in a principled way within the scheme of the planner, not in the actions’ description.

### III. BACKGROUND

To better introduce our approach, we provide a brief background on the underlying framework called HATP/EHDA, but first, let us discuss some relevant assumptions.

#### A. Important Assumptions

- Humans and robots are not equal. Still, they may collaborate to achieve a (shared) task.
- Humans are naturally uncontrollable agents and prefer their actions not to be imposed. So, their behavior can only be estimated and emulated. They are assumed to be cooperative, rational, and congruent but their involvement, computational capabilities, and tolerance regarding the shared task can 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.

#### B. The HATP/EHDA Framework

The HATP/EHDA framework [7] comprises a *dual* HTN-based task specification model. For more details about HTN refer to [30]. The framework plans the robot’s actions while estimating the possible human behaviors thanks to a given human model. This way, the generated plan is implicitly coordinated with all uncontrollable human actions.

This framework tackles problems where a human-robot team has to achieve a task together.

**Definition 1: (Human-Aware Task Planning Problem.)** The HATP problem 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 initial human belief state, which can contain facts that do not hold in  $s_0$ . Here,  $tn_0^r$  is the initial task network that the robot has to solve, respectively  $tn_0^h$  for the human. The two task networks are fully ordered and can contain similar tasks, hence considered as shared tasks. 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.

Each agent has its own belief state, action model, task network (agenda), plan, and triggers. The framework uses agents’ action models and beliefs to decompose agents’ task networks into primitive tasks (actions). The planning scheme assumes that a single agent decides to act at a time and which action it performs, and uses specific actions to synchronize the agent’s plans. *IDLE* is inserted in the agent’s plan when its task network is empty, and *WAIT* when it does not have any regular applicable action. First, it builds the whole search space by considering all possible, feasible decompositions. Then, considering plan evaluation with action and social costs, it can adapt off-the-shelf search algorithms to determine the best robot policy.

**Definition 2: ((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  $(o_1^h, o_2^h, \dots, o_m^h)$ .

Each branch in the solution tree is a sequence of primitive actions, say  $\pi = (o_1^r, o_2^h, o_3^r, \dots, o_{k-1}^h, o_k^r)$ , that must satisfy all the solution conditions of  $\mathcal{P}_{rh}$ . Here, each  $o_i^h$  represents a choice, often out of several, the human could make. This factor is crucial and decides the robot’s execution policy.

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 1, and generate a solution similar to that in Definition 2. 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.

#### IV. 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 [30].

To represent the objects and their properties, we will use two sets  $B$  and  $X$ :  $B$  is a set of names for all the objects, plus any mathematical constants representing properties of those 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  $x = sv(b_1, \dots, b_k)$ , 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:

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

$$\begin{aligned} X &= \{at(e), saltIn, stoveOn, counterClean \mid e \in Entities\} \\ Range(saltIn \mid stoveOn \mid counterClean) &= Booleans \\ Range(at(R \mid H \mid pasta)) &= Places \\ Range(at(salt \mid counter)) &= \{\text{kitchen}\} \end{aligned}$$

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  $X = \{x_1, \dots, x_n\}$ , we will often write this function as a set of assertions:  $val = \{x_1 = z_1, \dots, x_n = z_n\}$ .

A *variable observability assignment* function over  $X$  is a function  $obs$  that maps each  $x_i \in X$  into an observability type  $t_i \in \{\text{OBS}, \text{INF}\}$ :  $obs = \{(x_1, t_1), \dots, (x_n, t_n)\}$ . Respectively, when  $obs(x_i) = \text{OBS} \mid \text{INF}$  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  $l_i \in Places \cup \{\text{nil}\}$ :  $loc = \{(x_1, l_1), \dots, (x_n, l_n)\}$ .  $Places \subseteq B$  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. We consider  $x_i$  to be located in every  $place \in Places$  if  $loc(x_i) = \text{nil}$ . More details about how the environment should be divided into places will be given shortly.

A *state*  $s_i \in S$  is a 6-tuple composed of 4 functions over  $X$  and 2 task networks (agendas) s.t.  $s_i = (val_i, val_i^H, obs_i, loc_i, tn_i^R, tn_i^H)$ . The state of the world from the perspective of the robot is captured by the variable

value assignment function  $val_i$ , sometimes noted as  $val_i^R$ . Similarly,  $val_i^H$  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  $\exists x_j \in X, val_i^H(x_j) \neq val_i^R(x_j)$ .

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

$$\begin{aligned} s_0 &= \{val_0, val_0^H, obs_0, loc_0, tn_0^R, tn_0^H\} \\ val_0 &= val_0^H = \{at(R) = \text{kitchen}, at(H) = \text{kitchen}, \\ &\quad at(\text{pasta}) = \text{room}, saltIn = \text{false}, stoveOn = \text{false}\} \\ obs_0 &= \{(at(e), \text{OBS}), (saltIn, \text{INF}), (stoveOn, \text{OBS}) \\ &\quad (counterClean, \text{INF}), \mid e \in Entities\} \\ loc_0 &= \{(at(e), val_0(e)), (counterClean, \text{kitchen}), \\ &\quad (saltIn, \text{kitchen}), (stoveOn, \text{kitchen}), \mid e \in Entities\} \\ tn_0^R &= \{\text{CookPasta}, \text{CleanCounter}\} \\ tn_0^H &= \{\text{CookPasta}\} \end{aligned}$$

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

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 [7]. We refine the agent's agenda  $tn_\varphi$  based on its belief  $val_i^\varphi$  and obtain a *refinement* as follows  $ref(tn_\varphi^\varphi, val_i^\varphi) = \{(a_1, tn_1), \dots, (a_j, tn_j)\}$ . 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:

$$ref(tn_0^H, val_0^H) = \{(add\_salt(), tn_1), (move\_to(\text{kitchen}), tn_2)\}$$

#### V. 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. Thus,  $\forall x \in X$ , 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,  $\forall x \in X$ ,

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

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

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 human’s real-time sensing and reasoning capabilities about their surroundings.

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

**Definition 3: (Co-presence & Co-location.)** In a state  $s_i \in S$ , two agents,  $\varphi_1$  and  $\varphi_2$ , are considered to be *co-present* if  $val_i(at(\varphi_1)) = val_i(at(\varphi_2))$ . This relation is noted  $\varphi_1 \lambda_i \varphi_2$  in the rest of the paper. Similarly, we say that an agent  $\varphi_1$  is *co-located* with a state variable  $x \in X$  if  $val_i(at(\varphi_1)) = loc_i(x)$ , noted  $\varphi_1 \lambda_i x$ .

Now we can define two SA processes that will maintain the estimated human beliefs.

**Definition 4: (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  $s_i$ ,

$$val_{i+1}^H(x) = \begin{cases} w, & \text{if } x \leftarrow w \in \text{eff}(a) \text{ and} \\ & (H = \text{agt}(a) \text{ or } H \lambda_i \text{agt}(a) \\ & \text{or } H \lambda_{i+1} \text{agt}(a)) \\ val_i^H(x), & \text{otherwise} \end{cases}$$

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 5: (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  $val_{i+1}^H$  (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  $s_{i+1}$ ,  $\forall x \in X$ :

$$val_{i+1}^H(x) = \begin{cases} val_{i+1}(x), & \text{if } H \lambda_{i+1} x \text{ and} \\ & obs_{i+1}(x) = \text{OBS} \\ val_i^H(x), & \text{otherwise} \end{cases}$$

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.

Note that unlike in DEL [29], our knowledge representation 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 have 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.

## VI. RELEVANT FALSE BELIEF: DETECTION & SOLUTION

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

### A. 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 6). In such cases, the relevant false belief needs to be tackled.

**Definition 6:** A **relevant false belief** is a false belief that influences the next action(s) the human is likely to perform, 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:

$$ref(tn_i^H, val_i^H) \neq ref(tn_i^H, val_i^R) \quad (1)$$

$$\begin{aligned} & \{\gamma(s_i, a) \mid \forall a \in ref(tn_i^H, val_i^H)\} \\ & \neq \{\gamma(s_i, a) \mid \forall a \in ref(tn_i^H, val_i^R)\} \quad (2) \end{aligned}$$

We consider that as soon as a false belief has an effect on human actions 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 collaboration. But it is out of the scope of this work.

### B. 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.

**1) Modeling Communication Actions:** We propose a generic communication action schema ( $ca$ ) in this context. An agent  $\varphi_i$  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.

2) *Communicate Only the Required Facts*: Definition 6 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  $CA = \{ca_{R,H}(x_1, val_i^R(x_1)), \dots, ca_{R,H}(x_j, val_i^R(x_j))\}$  must be inserted in the plan for belief alignment. Thus, Definition 2 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  $ca \in CA$ , making the human’s belief to *update 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.

### C. 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 inferable 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.

## VII. EVALUATION

Referring to the related work section, we are not aware of an implemented planning system that can be used as a baseline. Hence, we use the HATP/EHDA solver to help present our approach’s results on three *novel* planning domains.

1) *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.

2) *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.

3) *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.

### A. Qualitative Analysis

Considering the cooking domain, we discuss in detail the plans obtained with our approach to a problem corresponding 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 Fig. 2 and 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.

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 *counterClean* is not *observable* the observation process does nothing and the false belief remains. However, this false

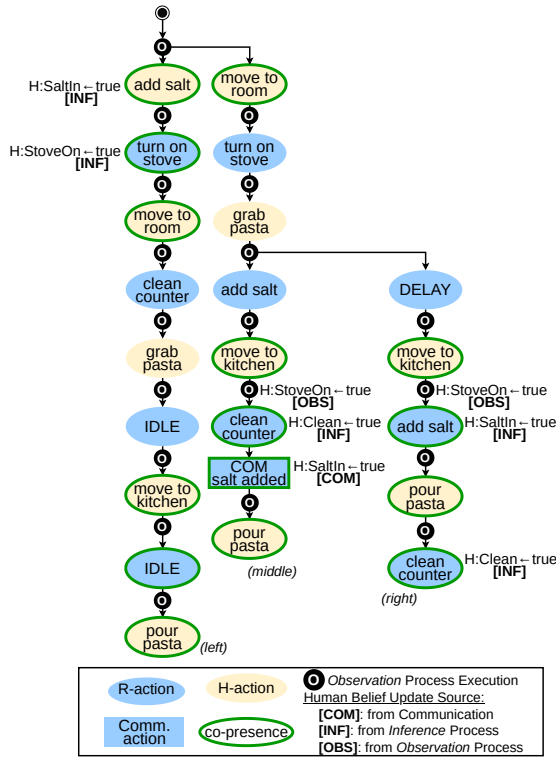


Fig. 2. Plan obtained for the cooking scenario. 3 branches. Left: The human starts by adding salt. The only false belief is about “counterClean” which is not relevant for the human agent, hence no comm is added. Middle: While the human is away the robot turns on the stove and adds salt, creating 2 false beliefs. Once back, we estimate that the human agent will be able to assess the observable fact “stoveOn” but not “saltIn”. 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.

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.

### B. 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  $H$ )

TABLE I

SUCCESS AND COMMUNICATION RATIO OF DIFFERENT APPROACHES.

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

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. 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, by our approach using first only communication and then using also delay. The obtained quantitative results appear in TABLE I.

The overall success rate ( $S$ ) and the one for initially diverging beliefs ( $S I.Div.B.$ ) are shown for the HATP/EHDA solver. As expected, this solver always finds legal plans when dealing with initially aligned beliefs, but the low value of  $S I.Div.B.$  reflects how poorly it handles belief divergences without specifically designed action models. Our approach always finds legal plans so we omitted its success rates in the last two columns, and we can say that 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 delayed branches that do not imply the human to wait are kept.

## VIII. DISCUSSION

The underlying scheme allows just a single agent to execute a “real” action at a time. However, a post-process can allow the execution of actions concurrently [31], however, note that the domain modeler has modeled  $\mathcal{P}_{rh}$  as a sequential joint task. Parallelism is not considered in the current modeling and planning process, which limits the potential for concurrent executions. However, we are working on extending the framework to enable systematic planning with concurrent actions, aligning with [32].

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 cannot simply refute a false belief, they can only assess new true facts to correct them. E.g.,



assume the human *wrongly* believes that the pasta is in kitchen. The SA does not help refute this when the agent is in kitchen because appropriate knowledge reasoning w.r.t. *NotAt(Pasta)* in kitchen is not taken into account. However, such issues do not affect the completeness and, if necessary, our approach *tackles* such cases as relevant false beliefs.

We have planned a user study for the future to conform our framework with reality and validate the approach.

We discussed earlier that DEL knowledge representation is more expressive and flexible, and can handle 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.

## IX. 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 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 necessary ones in a principled way. First, we propose minimal and proactive explicit communication. Second, when pertinent, we propose an implicit communication by postponing the non-observed robot action until the human is co-present to observe it.

The relevance of false belief, when to optimally communicate and parallelization are interesting future works, and we aim at conducting a user study to validate the benefits of the proactive robot behavior that our approach permits.

## REFERENCES

- [1] H. Wimmer and J. Perner, "Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children's understanding of deception," *Cognition*, vol. 13, no. 1, pp. 103–128, 1983. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0010027783900045>
- [2] M. Selvaggio, M. Cagnetti, S. Nikolaidis, S. Ivaldi, and B. Siciliano, "Autonomy in physical human-robot interaction: A brief survey," *IEEE Robotics Autom. Lett.*, 2021.
- [3] V. V. Unhelkar, S. Li, and J. A. Shah, "Decision-making for bidirectional communication in sequential human-robot collaborative tasks," in *Proc. of HRI*, 2020.
- [4] V. V. Unhelkar, P. A. Lasota, Q. Tyroller, R.-D. Buhai, L. Marceau, B. Deml, and J. A. Shah, "Human-aware robotic assistant for collaborative assembly: Integrating human motion prediction with planning in time," *IEEE RAL*, vol. 3, no. 3, pp. 2394–2401, 2018.
- [5] M. G. Jacob, Y.-T. Li, G. A. Akingba, and J. P. Wachs, "Collaboration with a robotic scrub nurse," *Communications of the ACM*, vol. 56, no. 5, pp. 68–75, 2013.
- [6] S. Devin and R. Alami, "An implemented ToM to improve human-robot shared plans execution," in *HRI Proc.*, 2016.
- [7] G. Buisan, A. Favier, A. Mayima, and R. Alami, "HATP/EHDA: A Robot Task Planner Anticipating and Eliciting Human Decisions and Actions," in *Proc. of ICRA*, 2022.
- [8] V. V. Unhelkar, S. Li, and J. A. Shah, "Semi-supervised learning of decision-making models for human-robot collaboration," in *Proc. of CoRL*, 2019.
- [9] M. Berlin, J. Gray, A. L. Thomaz, and C. Breazeal, "Perspective taking: An organizing principle for learning in human-robot interaction," in *Proc. of AAAI*, 2006.
- [10] M. Shvo, T. Q. Klassen, and S. A. McIlraith, "Resolving misconceptions about the plans of agents via theory of mind," in *ICAPS*, 2022.
- [11] T. Bolander and M. B. Andersen, "Epistemic planning for single and multi-agent systems," *J. Appl. Non Class. Logics*, vol. 21, no. 1, pp. 9–34, 2011.
- [12] M. Cirillo, L. Karlsson, and A. Saffiotti, "A human-aware robot task planner," in *Proc. of ICAPS 2009*, 2009.
- [13] A. Kulkarni, S. Srivastava, and S. Kambhampati, "Planning for proactive assistance in environments with partial observability," *CoRR*, vol. abs/2105.00525, 2021.
- [14] T. Chakraborti, G. Briggs, K. Talamadupula, Y. Zhang, M. Scheutz, D. E. Smith, and S. Kambhampati, "Planning for serendipity," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015, Hamburg, Germany, September 28 - October 2, 2015*. IEEE, 2015, pp. 5300–5306.
- [15] R. Lallement, L. de Silva, and R. Alami, "Hatp: hierarchical agent-based task planner," in *Proc. of AAMAS*, 2018.
- [16] A. Roncone, O. Mangin, and B. Scassellati, "Transparent role assignment and task allocation in human robot collaboration," in *Proc. of ICRA*, 2017.
- [17] K. Darvish, E. Simetti, F. Mastrogiovanni, and G. Casalino, "A hierarchical architecture for human-robot cooperation processes," *IEEE Trans. Robotics*, vol. 37, no. 2, pp. 567–586, 2021.
- [18] M. C. Gombolay, R. Jensen, J. Stigile, S. Son, and J. A. Shah, "Apprenticeship scheduling: Learning to schedule from human experts," in *Proc. of IJCAI*, 2016.
- [19] M. Hörger, H. Kurniawati, and A. Elfes, "Multilevel monte-carlo for solving pomdps online," in *Proc. of ISRR*, 2019.
- [20] R. Alami, A. Clodic, V. Montreuil, E. A. Sisbot, and R. Chatila, "Toward human-aware robot task planning," in *AAAI spring symposium: to boldly go where no human-robot team has gone before*, 2006.
- [21] U. Köckemann, F. Pecora, and L. Karlsson, "Grandpa hates robots - interaction constraints for planning in inhabited environments," in *AAAI*. AAAI Press, 2014, pp. 2293–2299.
- [22] V. V. Unhelkar and J. A. Shah, "Learning models of sequential decision-making with partial specification of agent behavior," in *Proc. of AAAI*, 2019.
- [23] B. Hayes and B. Scassellati, "Autonomously constructing hierarchical task networks for planning and human-robot collaboration," in *Proc. of ICRA*, 2016.
- [24] S. Tellex, R. A. Knepper, A. Li, D. Rus, and N. Roy, "Asking for help using inverse semantics," in *Proc. of RSS*, 2014.
- [25] S. Nikolaidis, M. Kwon, J. Forlizzi, and S. Srinivasa, "Planning with verbal communication for human-robot collaboration," *ACM Transactions on HRI*, vol. 7, no. 3, pp. 1–21, 2018.
- [26] V. Sanelli, M. Cashmore, D. Magazzeni, and L. Iocchi, "Short-term human-robot interaction through conditional planning and execution," in *Proc. of ICAPS*, 2017.
- [27] M. C. Cooper, A. Herzig, F. Maffre, F. Maris, E. Perrotin, and P. Régnier, "A lightweight epistemic logic and its application to planning," *Artif. Intell.*, vol. 298, p. 103437, 2021.
- [28] T. Engesser, T. Bolander, R. Mattmüller, and B. Nebel, "Cooperative epistemic multi-agent planning for implicit coordination," in *Proc. of WS on EPTCS*, 2017.
- [29] T. Bolander, L. Dissing, and N. Herrmann, "DEL-based Epistemic Planning for Human-Robot Collaboration: Theory and Implementation," in *Proc. of KR*, 2021.
- [30] M. Ghallab, D. S. Nau, and P. Traverso, *Automated Planning - Theory and Practice*. Elsevier, 2004.
- [31] M. Crosby, A. Jonsson, and M. Rovatsos, "A single-agent approach to multiagent planning," in *Proc. of ECAI*, 2014.
- [32] S. Shekhar and R. I. Brafman, "Representing and planning with interacting actions and privacy," *AIJ*, vol. 278, 2020.