Proceedings of the Thirty-Second International Conference on Automated Planning and Scheduling (ICAPS 2022)

Resolving Misconceptions about the Plans of Agents via Theory of Mind*

Maayan Shvo, Toryn Q. Klassen, Sheila A. McIlraith

Department of Computer Science, University of Toronto, Toronto, Canada
Vector Institute, Toronto, Canada
Schwartz Reisman Institute for Technology and Society, Toronto, Canada
{maayanshvo,toryn,sheila}@cs.toronto.edu

Abstract

For a plan to achieve some goal – to be valid – a set of sufficient and necessary conditions must hold. In dynamic settings, agents (including humans) may come to hold false beliefs about these conditions and, by extension, about the validity of their plans or the plans of other agents. Since different agents often believe different things about the world and about the beliefs of other agents, discrepancies may occur between agents' beliefs about the validity of plans. In this work, we explore how agents can use their Theory of Mind to resolve such discrepancies by communicating and/or acting in the environment. We appeal to an epistemic logic framework to allow agents to reason over other agents' nested beliefs, and demonstrate how epistemic planning tools can be used to resolve discrepancies regarding plan validity in a number of domains. Our work shows promise for human decision support as demonstrated by a study that showcases the ability of our approach to resolve misconceptions held by humans.

1 Introduction

"Planning is the art of thinking before acting" (Haslum 2014), but a problem with thinking before acting is that the validity of the resultant plan is predicated on *beliefs* about the way the world is, rather than ground truth, and even if those beliefs are correct at the time of planning (and they may not be!), the actual state of the world may change prior to plan execution, invalidating the plan, sometimes unbeknownst to various agents. Moreover, agents may perceive discrepancies between their own beliefs and other agents' beliefs about the validity of plans (e.g., Alice believes that Bob's plan is not valid but that *he* believes it is).

Here we wish to allow agents to contemplate each others' plans, realize when agents hold misconceptions about the validity of their plans or the plans of other agents, and resolve discrepancies pertaining to the validity of these plans by communicating and/or acting in the environment. For example, a robot could communicate to its human teammate that the conditions necessary to the success of her plan do not hold or, alternatively, the robot could act in the world to ensure that those conditions hold.

To contemplate another agent's beliefs and plans, agents must employ their Theory of Mind which, according to Premack and Woodruff (1978), is exercised when an agent imputes mental states (e.g., plans, goals, beliefs) to itself and others. To enable agents to employ their Theory of Mind, we appeal to epistemic logic and propose a framework that allows agents to identify and resolve discrepancies between their beliefs and the beliefs of others regarding plan validity. Importantly, our framework allows agents to be aware of and reason about the mental states of human counterparts and offer assistance by resolving perceived discrepancies.

Recent work in Explainable AI Planning (XAIP) has stressed the need to consider the possibly incomplete and incorrect perspective of other agents when resolving misconceptions pertaining to various properties of plans (e.g., optimality and validity). For instance, the *model reconciliation* literature has investigated how to enable planning agents to resolve discrepancies between the planning models of the planning agent and the observing human(s) (Sreedharan, Chakraborti, and Kambhampati 2021). In Section 7 we elaborate and survey additional related work.

Our work goes beyond extant work by supporting a unique variety of settings requiring complex Theory of Mind reasoning. In particular, the expressive nature of our framework supports (1) nested belief attribution (e.g., in order to resolve Mary's misconception about Bob's beliefs about the validity of his plan, Alice may inform Mary that Bob holds a false belief about some fact relevant to the plan's success); and (2) reasoning about threats to the achievement of *epistemic goals* (e.g., if Bob's epistemic goal is for Mary to know ϕ without Eve knowing ϕ and Bob's robot teammate knows that (unbeknownst to Bob) Eve is within earshot, then the robot could inform Bob of Eve's proximity). To realize our approach, we establish a relationship between our proposed formulation of discrepancy resolution and *epistemic planning* which is focused on generating plans to achieve epistemic goals in the context of agents' beliefs and knowledge (Petrick and Bacchus 2002; Bolander and Andersen 2011; Kominis and Geffner 2015; Muise et al. 2015b; Huang et al. 2017; Le et al. 2018; Fabiano et al. 2020, 2021).

The contributions of our paper are as follows:

1. We propose a formulation of discrepancy resolution that appeals to a multi-agent epistemic logic (Sec. 3).
2. We present an algorithm that resolves discrepancies via

*Associated technical appendix appears in Shvo et al. (2022).

epistemic planning and establish its soundness (Sec. 4).
3. We demonstrate that epistemic planning tools can be used to resolve discrepancies via different modalities (i.e., with communicative and/or world-altering actions) in various domains and evaluate the impact of the depth of nested belief on the runtime of our algorithm (Sec. 5).
4. We conduct a user study which indicates that our approach can effectively resolve misconceptions held by humans pertaining to plan validity (Sec. 6).

2 Preliminaries

KD45$_n$. We briefly discuss the multi-agent modal logic KD45$_n$ which we appeal to in this work (Fagin et al. 2004). Let Ag and $\mathcal{P}$ be finite sets of agents and atoms, respectively. ϕ and ψ are used to represent formulae. $\top$ and $\bot$ represent *true* and *false*, respectively. The language $\mathcal{L}$ of multi-agent modal logic is generated by the following BNF:

$$\phi ::= p \mid \neg\phi \mid \phi \wedge \phi' \mid B_i\phi$$

where $p \in \mathcal{P}$, $i \in Ag$, $\phi \in \mathcal{L}$ and $B_i\phi$ means that *"agent i believes ϕ."* The semantics for formulae in $\mathcal{L}$ is given by Kripke models (Fagin et al. 2004) which are triplets, $M = \langle W, R, V\rangle$, containing a set of worlds, accessibility relations between the worlds for each of the agents ($R = \{R_i \mid i \in Ag\}$), and a valuation map, $V\colon W \rightarrow 2^P$. The set of worlds an agent i (at world $w \in W$) considers possible is given by M and the accessibility relations in R_i pertaining to w. R_i is a binary relation on W and is a subset of $W \times W$. A formula ϕ is true in a world w of a Kripke model $M = \langle W, R, V\rangle$, written $M, w \models \phi$, under these conditions:

$M, w \models p$ for an atom p, *iff* $p \in V(w)$,
$M, w \models \neg\phi$, *iff* $M, w \not\models \phi$,
$M, w \models \phi \wedge \psi$, *iff* both $M, w \models \phi$ and $M, w \models \psi$,
$M, w \models B_i\phi$, *iff* $M, w' \models \phi \quad \forall w' \in W$ s.t. $R_i(w, w')$

ϕ is satisfiable if there is a Kripke model M and a world w of M s.t. $M, w \models \phi$. ϕ is said to entail ψ, written $\phi \models \psi$, if for any Kripke model M, $M, w \models \phi$ entails $M, w \models \psi$. We are interested in a set of properties of belief and assume a number of constraints on Kripke models to achieve this (Fagin et al. 2004). In particular, Kripke models are:

Serial - $\forall w\, \exists v\, R(w, v)$
Transitive - $R(w, v) \wedge R(v, u) \Rightarrow R(w, u)$
Euclidean - $R(w, v) \wedge R(w, u) \Rightarrow R(v, u)$

with the resulting properties of belief:

$B_i\phi \wedge B_i(\phi \Rightarrow \psi) \Rightarrow B_i\psi$ (K - Distribution)
$B_i\phi \Rightarrow \neg B_i\neg\phi$ (D - Consistency)
$B_i\phi \Rightarrow B_iB_i\phi$ (4 - Positive Introspection)
$\neg B_i\phi \Rightarrow B_i\neg B_i\phi$ (5 - Negative Introspection)

This is the KD45$_n$ system (n is the number of agents in the environment) that is defined by these properties of belief.

Epistemic planning combines automated planning and reasoning over the beliefs and knowledge of agents. We appeal to a multi-agent epistemic planning formulation to represent the beliefs of different agents in a dynamic setting.

Definition 1 (MEP Problem) *A* **Multi-agent Epistemic Planning Problem** *is a tuple $\langle Q, \mathcal{I}, G\rangle$ where $Q = \langle \mathcal{P}, \mathcal{A}, Ag\rangle$ is the **domain** comprising sets of atoms $\mathcal{P}$, actions $\mathcal{A}$, and agents Ag, together with the problem instance description comprising the initial state, $\mathcal{I} \in \mathcal{L}$, and the goal condition $G \in \mathcal{L}$, where $\mathcal{L}$ is the language of multi-agent modal logic corresponding to $\mathcal{P}$ and Ag.*

$\mathcal{A}$ is a set of actions where each action $a \in \mathcal{A}$ is a tuple $\langle \text{Pre}, \{(\gamma_1, \epsilon_1), ..., (\gamma_k, \epsilon_k)\}\rangle$, where Pre $\in \mathcal{L}$ is the precondition of a (written Pre(a)), $\gamma_i \in \mathcal{L}$ is the condition of a conditional effect, and $\epsilon_i \in \mathcal{L}$ is the effect of a conditional effect. To model how the state of the world and agents' beliefs change following the execution of an action, we rely on the definition and realization of *progression* by the epistemic planner we use in this work, RP-MEP (Muise et al. 2015b). RP-MEP's definition of a progression operator, rather than progressing arbitrary formulae in $\mathcal{L}$, operates over syntactically restricted formulae – Proper Epistemic Knowledge Bases (PEKBs) (Lakemeyer and Lespérance (2012) building on Liu, Lakemeyer, and Levesque (2004) and Levesque (1998) with further work by Muise et al. (2015a)). A PEKB is defined as a set of restricted formulae called *restricted modal literals* (RMLs) (Lakemeyer and Lespérance 2012). An RML is obtained from the following grammar:

$$\phi ::= p \mid B_i\phi \mid \neg\phi$$

where $p \in \mathcal{P}$ and $i \in Ag$, and as a consequence PEKBs do not contain disjunctive formulae. For more details on PEKBs and RMLs, see Appendix A and (Muise et al. 2022). Muise et al. (2022) define a progression operator over a restricted MEP setting where states, preconditions, and conditional effects are PEKBs. *In the remainder of the paper, we make use of this restricted form of MEP.*

Definition 2 (PROG (Muise et al. 2022, Definition 4)) *Given a PEKB state ϕ and an action $a = \langle Pre, \{(\gamma_1, \epsilon_1), ..., (\gamma_k, \epsilon_k)\}\rangle$ where Pre and each γ_i are PEKBs and each ϵ_i is an RML, the **progression** of ϕ wrt a, a PEKB state labelled* $\text{PROG}(a, \phi)$*, is*

$$\text{PROG}(a, \phi) = (\phi \blacklozenge (R \cup U)) \diamond Q$$
$$Q = \bigcup_{1 \le i \le k} \{\psi \mid \gamma_i \subseteq \phi \text{ and } \epsilon_i \models \psi\}$$
$$R = \bigcup_{1 \le i \le k} \{\psi \mid \gamma_i \subseteq \phi \text{ and } \epsilon_i \models \psi\}$$
$$U = \bigcup_{1 \le i \le k} \{\neg\psi \mid \overline{\gamma_i} \cap \phi = \emptyset \text{ and } \neg\epsilon_i \models \neg\psi\}$$

where $\overline{P}$ is the PEKB that contains the negation of every RML in some PEKB P and $\blacklozenge$ and $\diamond$ are belief erasure and belief update operators[1]*, respectively. In case a is not executable in ϕ, i.e. Pre $\not\subseteq \phi$,* $\text{PROG}(a, \phi)$ *is undefined. Finally, Q defines the set of literals to be added, R defines the set*

[1]Belief update and erasure for PEKBs have been defined and shown to be polynomial time operations (Miller and Muise 2016).

of literals to be deleted, and U defines the set of uncertain firing[2] literals to be deleted.

For additional details on RP-MEP's progression, we refer readers to (Muise et al. 2022). We use the shorthand $\text{PROG}([a_1, ..., a_n], \phi)$ or $\text{PROG}(\pi, \phi)$ to denote the progression of ϕ wrt a sequence of actions, a *plan*, $\pi = [a_1, ..., a_n]$. We use $\mathcal{I}$ and S to denote PEKB states. Finally, when talking about a tuple $\vec{v} = \langle v_1, \ldots, v_n \rangle$ of agents from Ag, we may use $B_{v_1 \ldots v_n}$ or $B_{\vec{v}}$ to stand for the belief operator sequence $B_{v_1} \ldots B_{v_n}$. In case $\vec{v}$ is empty, $B_{\vec{v}}p$ represents p.

3 Resolving Discrepancies

In this section, we propose a formulation of discrepancy resolution for plan validity that appeals to a multi-agent epistemic logic. For a plan to achieve some goal – to be valid – a set of sufficient and necessary conditions must hold.

Definition 3 (Plan Validity) *Given a domain $\langle \mathcal{P}, \mathcal{A}, Ag \rangle$, a state S, and a goal G, a plan π is valid for achieving (or simply achieves) G from S if $\text{PROG}(\pi, S) \models G$.*

We want a formula $\text{VAL}(\pi, G)$ that captures plan validity in the sense that $\text{PROG}(\pi, S) \models G$ if and only if $S \models \text{VAL}(\pi, G)$. We characterize $\text{VAL}(\pi, G)$ by appealing to regression rewriting (Waldinger 1977; Reiter 2001; Rintanen 2008; Fritz and McIlraith 2007), a form of preimage computation that takes a formula and an action and returns the condition that is necessary to hold in the current state for the formula to hold in the state resulting from performing the action. Regression can be applied repeatedly to compute the condition that must be true in the initial state for the goal to hold in the state resulting from the execution of the actions in a sequential plan. Here we suppose that we have (given a domain) a *regression* operator REG which maps a formula ϕ and action sequence π to a formula $\text{REG}(\pi, \phi)$ which satisfies the property that for any state S, $S \models \text{REG}(\pi, \phi)$ if and only if $\text{PROG}(\pi, S) \models \phi$. Finally, we say that $S \models \text{VAL}(\pi, G)$ if and only if $S \models \text{REG}(\pi, G)$. Importantly, since $\text{VAL}(\pi, G)$ is a formula, we can talk about agents' beliefs about it, *which we can interpret as indicating their beliefs about whether π is a valid plan.*

Definition 4 (Subjective Plan Validity) *Given a domain $\langle \mathcal{P}, \mathcal{A}, Ag \rangle$, a state S, and a goal G, agent i believes that a plan π is valid if $S \models B_i \text{VAL}(\pi, G)$.*

Agents can also hold beliefs about other agents' beliefs (about other agents' beliefs...) about the validity of a plan and perceive *discrepancies* between their beliefs and the beliefs of other agents about plan validity.

Definition 5 (Discrepancy) *Given a domain $\langle \mathcal{P}, \mathcal{A}, Ag \rangle$, agents $i, j \in Ag$, and a (possibly empty) tuple $\vec{v} = \langle v_1, \ldots, v_n \rangle$ of agents in Ag, we say that a formula ϕ is a **discrepancy** perceived by agent i in state S between its beliefs and those of agent j (about the beliefs of agent v_1 about the beliefs of ... about the beliefs of agent v_n) if the following condition is entailed by S:*

$$\neg(B_{i,\vec{v}}\phi \wedge B_{i,j,\vec{v}}\phi) \wedge \neg(B_{i,\vec{v}}\neg\phi \wedge B_{i,j,\vec{v}}\neg\phi)$$

We will be interested in discrepancies about formulae like $\text{VAL}(\pi, G)$, i.e., in discrepancies about the validity of plans, and in enabling agents to resolve such discrepancies by changing the environment or other agents' beliefs. To this end, we cast the task of resolving a discrepancy perceived by agent i between its beliefs and those of agent j as an *epistemic goal*, where agent i needs to either change j's beliefs to align with its own, or change its own beliefs to align with j's beliefs. The following definition is of a plan that achieves this goal and ensures that in the end, $\text{VAL}(\pi, G)$ will *not* be a discrepancy perceived by agent i between its beliefs and those of agent j (about the beliefs of agent v_1 about the beliefs of ... about the beliefs of agent v_n).

Definition 6 ((Plan Validity) Discrepancy Resolving Plan) *Given a domain $Q = \langle \mathcal{P}, \mathcal{A}, Ag \rangle$, agents $i, j \in Ag$, a (possibly empty) tuple $\vec{v} = \langle v_1, ..., v_n \rangle$ of agents in Ag, initial state $\mathcal{I}$, a plan π, and a goal G, a **plan validity discrepancy resolving plan** (henceforth discrepancy resolving plan) for $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G \rangle$ is a plan π' that achieves the following goal from $\mathcal{I}$:*

$$(B_{i,j,\vec{v}}\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\text{VAL}(\pi, G)) \vee$$
$$(B_{i,j,\vec{v}}\neg\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\neg\text{VAL}(\pi, G)).$$

There are many ways to resolve a discrepancy, some of which are trivial or undesirable. For instance, suppose that agent i is a planning system trying to explain the validity of its own plan, π, to agent j (Bob, the human user of the system). The system believes that π is valid while believing also that Bob believes that π is not valid. In this case, a valid discrepancy resolving plan, π', would be for the system to render π invalid which resolves the discrepancy – the system now believes that π is not valid and also believes that Bob believes it is not valid. However, this is an undesirable solution since the system's intention was to convince Bob of π's validity. We therefore often wish to resolve discrepancies under certain conditions by constraining the discrepancy resolution epistemic goal specified in Definition 6.

Definition 7 (Constrained Discrepancy Resolving Plan) *Given a domain $Q = \langle \mathcal{P}, \mathcal{A}, Ag \rangle$, agents $i, j \in Ag$, a (possibly empty) tuple $\vec{v} = \langle v_1, ..., v_n \rangle$ of agents in Ag, initial state $\mathcal{I}$, a plan π, a goal G, and a logical formula Φ representing additional constraints, a **constrained discrepancy resolving plan** for $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G \rangle$ is a plan π' that achieves the following goal from $\mathcal{I}$:*

$$[(B_{i,j,\vec{v}}\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\text{VAL}(\pi, G)) \vee$$
$$(B_{i,j,\vec{v}}\neg\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\neg\text{VAL}(\pi, G))] \wedge \Phi.$$

Note that as written here, Φ imposes a constraint on what states the plan can end in. It might also be desirable to constrain the plan trajectory (e.g., to require some condition

[2]Uncertain firing occurs when an agent is unsure whether a cond. effect is true and should therefore not believe the effect but must also not believe the opposite (Muise et al. 2022).

holds throughout the entire plan), as has been explored in the literature on temporally extended goals (e.g., Baier and McIlraith 2006). Our definition could be extended to do that, though we will not pursue that further here.

Each of the following definitions specifies different conditions involving Φ, resulting in two conceptually distinct ways to resolve discrepancies: (1) by changing agent j's beliefs to align with i's beliefs, and (2) by changing agent i's beliefs to align with j's beliefs.

Definition 8 (i-aligned Discrepancy Resolving Plan) *Given a domain $Q = \langle \mathcal{P}, \mathcal{A}, Ag \rangle$, agents $i, j \in Ag$, a (possibly empty) tuple $\vec{v} = \langle v_1, ..., v_n \rangle$ of agents in Ag, initial state $\mathcal{I}$, a plan π, and a goal G, an **i-aligned discrepancy resolving plan** for $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G \rangle$ is a constrained discrepancy resolving plan π' where the constraint Φ satisfies these conditions:*

1. *If $\mathcal{I} \models B_{i,\vec{v}}\text{VAL}(\pi, G)$ then*
   $\Phi \models B_{i,j,\vec{v}}\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\text{VAL}(\pi, G)$.
2. *If $\mathcal{I} \models B_{i,\vec{v}}\neg\text{VAL}(\pi, G)$ then*
   $\Phi \models B_{i,j,\vec{v}}\neg\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\neg\text{VAL}(\pi, G)$.

While discrepancy resolving plans need not contain only actions performed by agent i, one useful form of i-aligned discrepancy resolving plans involves agent i communicating salient information to agent j either *implicitly* (e.g., by opening a box in front of agent j, demonstrating that it is unlocked) or *explicitly* (e.g., by telling agent j the box is unlocked). i-aligned discrepancy resolving plans are important for a variety of settings. For instance, recall the aforementioned undesirable discrepancy resolving plan π' that rendered the planning system's plan invalid. While π' is a valid discrepancy resolving plan, it is not a valid i-aligned discrepancy resolving plan. Therefore, to avoid such undesirable solutions in the plan explanation setting, we would generate i-aligned discrepancy resolving plans, thus preserving the validity of the plan π. In contrast, discrepancies can be resolved by changing agent i's beliefs to align with j's beliefs.

Definition 9 (j-aligned Discrepancy Resolving Plan) *Given a domain $Q = \langle \mathcal{P}, \mathcal{A}, Ag \rangle$, agents $i, j \in Ag$, a (possibly empty) tuple $\vec{v} = \langle v_1, ..., v_n \rangle$ of agents in Ag, initial state $\mathcal{I}$, a plan π, and a goal G, a **j-aligned discrepancy resolving plan** for $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G \rangle$ is a constrained discrepancy resolving plan π' where the constraint Φ satisfies these conditions:*

1. *If $\mathcal{I} \models B_{i,j,\vec{v}}\text{VAL}(\pi, G)$ then*
   $\Phi \models B_{i,j,\vec{v}}\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\text{VAL}(\pi, G)$.
2. *If $\mathcal{I} \models B_{i,j,\vec{v}}\neg\text{VAL}(\pi, G)$ then*
   $\Phi \models B_{i,j,\vec{v}}\neg\text{VAL}(\pi, G) \wedge B_{i,\vec{v}}\neg\text{VAL}(\pi, G)$.

One form of j-aligned discrepancy resolving plans involves agent i changing the environment to align with j's beliefs. For example, agent i could place some item where she believes agent j falsely believes it to be, in order to make j's plan valid. Such plans facilitate assistance which does not require coordination or communication with agent j.

Finally, Definition 6 does not consider the plans of the other agents in Ag. Therefore, it is possible that a valid discrepancy resolving plan will introduce new discrepancies pertaining to the validity of other agents' plans (e.g., making some agent's plan invalid while they believe it is valid). Φ can be specified appropriately such that discrepancy resolving plans preserve the validity of other agents' plans.

3.1 Example

Using an example, we illustrate the concepts discussed in this section. Consider a search and rescue scenario with three agents, Alice (virtual assistant or robot), Bob (human), and Mary (human), where all agents are aware that Bob's goal is to obtain a particular medical kit (Kit1). Alice believes that Bob falsely believes that Kit1 is in room A (Alice herself believes that the medical kit is in room B). Alice also believes that Mary falsely believes that Bob believes that Kit1 is in room B. We partially model this scenario:

$$Ag = \{\text{Alice, Mary, Bob}\} \quad (1)$$
$$\mathcal{I} \models B_{\text{Alice}}\textit{at}(\text{Bob, Hall}) \quad (2)$$
$$\mathcal{I} \models B_{\text{Alice}}\textit{at}(\text{Kit1, RoomB}) \quad (3)$$
$$\mathcal{I} \models B_{\text{Alice}}\neg\textit{at}(\text{Kit1, RoomA}) \quad (4)$$
$$\mathcal{I} \models B_{\text{Alice,Mary}}\textit{at}(\text{Kit1, RoomB}) \quad (5)$$
$$\mathcal{I} \models B_{\text{Alice,Bob}}\textit{at}(\text{Kit1, RoomA}) \quad (6)$$
$$\mathcal{I} \models B_{\text{Alice,Bob}}\neg\textit{at}(\text{Kit1, RoomB}) \quad (7)$$
$$\mathcal{I} \models B_{\text{Alice,Mary,Bob}}\textit{at}(\text{Kit1,RoomB}) \quad (8)$$
$$\mathcal{I} \models B_{\text{Alice,Mary,Bob}}\neg\textit{at}(\text{Kit1,RoomA}) \quad (9)$$

Let us assume that Bob's goal G is *holding*(Bob,Kit1) and that Alice predicts[3] that Bob's plan to achieve G is

[*move*(Bob,Hall,RoomA), *pckUp*(Bob,Kit1,RoomA)].

We refer to Alice's prediction about Bob's plan as π_{AliceBob}. Moreover, let us assume that Alice can reason that Mary predicts that Bob's plan is

[*move*(Bob,Hall,RoomB),*pckUp*(Bob,Kit1,RoomB)].

We refer to Alice's prediction about Mary's prediction about Bob's plan as $\pi_{\text{AliceMaryBob}}$. The actions in π_{AliceBob} are:

move(Bob,Hall,RoomA) = $\langle$*at*(Bob, Hall),
$\{(\top, \textit{at}(\text{Bob, RoomA})), (\top, \neg\textit{at}(\text{Bob, Hall}))\}\rangle$
pckUp(Bob,Kit1,RoomA) =
$\langle$*at*(Kit1, RoomA) $\wedge$ *at*(Bob, RoomA),
$\{(\top, \textit{holding}(\text{Bob, Kit1})), (\top, \neg\textit{at}(\text{Kit1, RoomA}))\}\rangle$

Actions in $\pi_{\text{AliceMaryBob}}$ are identical with RoomB replacing RoomA. Agents are 'aware' that an action has been performed if they are in the same location in which the action is performed. For example, if Bob picks up Kit1 in RoomA and Mary is also there, then Mary will believe that Bob is holding Kit1. The planner we use in this work, RP-MEP, achieves this by automatically generating conditional effects from an

[3] Plan recognition techniques can be used to predict or recognize other agents' plans. See Section 8 for a brief discussion.

action's existing set of conditional effects. For more details, see (Muise et al. 2022) and Appendix E. Next, suppose that $\text{VAL}(\pi_{\text{AliceBob}}, G)$ and $\text{VAL}(\pi_{\text{AliceMaryBob}}, G)$ are

$$\textit{at}(\text{Kit1, RoomA}) \wedge \textit{at}(\text{Bob, Hall}) \text{ and}$$
$$\textit{at}(\text{Kit1, RoomB}) \wedge \textit{at}(\text{Bob, Hall}),$$

respectively. That is, for π_{AliceBob} to be valid, Bob must initially be in the hallway and Kit1 must be in RoomA. Given entailments (2)-(9) (and assuming that all agents believe (that all agents believe) *at*(Bob, Hall)), the following holds pertaining to agents' beliefs about the validity of π_{AliceBob} and $\pi_{\text{AliceMaryBob}}$:

$$\mathcal{I} \models B_{\text{Alice}} \neg \text{VAL}(\pi_{\text{AliceBob}}, G) \quad (10)$$
$$\mathcal{I} \models B_{\text{Alice,Bob}} \text{VAL}(\pi_{\text{AliceBob}}, G) \quad (11)$$
$$\mathcal{I} \models B_{\text{Alice,Mary,Bob}} \neg \text{VAL}(\pi_{\text{AliceBob}}, G) \quad (12)$$
$$\mathcal{I} \models B_{\text{Alice}} \text{VAL}(\pi_{\text{AliceMaryBob}}, G) \quad (13)$$
$$\mathcal{I} \models B_{\text{Alice,Bob}} \neg \text{VAL}(\pi_{\text{AliceMaryBob}}, G) \quad (14)$$
$$\mathcal{I} \models B_{\text{Alice,Mary,Bob}} \text{VAL}(\pi_{\text{AliceMaryBob}}, G) \quad (15)$$

Alice perceives in $\mathcal{I}$ a number of discrepancies between her beliefs and those of Bob and Mary pertaining to plan validity. In particular, $\text{VAL}(\pi_{\text{AliceBob}}, G)$ is a discrepancy perceived by Alice between her beliefs and those of Bob, where $\vec{v}$ is empty (entailments (10) and (11)). One possible (i-aligned) discrepancy resolving plan is then

$\pi' = [\textit{inform}(\text{Alice,Bob,}\neg\textit{at}(\text{Kit1,RoomA}))]$, such that
$\mathcal{I} \models \text{VAL}(\pi',$
$B_{\text{Alice,Bob}} \neg \text{VAL}(\pi_{\text{AliceBob}}, G) \wedge B_{\text{Alice}} \neg \text{VAL}(\pi_{\text{AliceBob}}, G))$

The grounded inform action in π' is modelled as follows:

$$\textit{inform}(\text{Alice,Bob,}\neg\textit{at}(\text{Kit1,RoomA})) =$$
$$\langle B_{\text{Alice}} \neg \textit{at}(\text{Kit1, RoomA}),$$
$$\{(\top, B_{\text{Alice,Bob}} \neg \textit{at}(\text{Kit1, RoomA}))\}\rangle.$$

The plan π' consists of Alice informing Bob that Kit1 in not in RoomA. This resolves Alice's perceived discrepancy about the validity of π_{AliceBob}. That is, Alice believes that after Bob learns that Kit1 is not in RoomA, he will believe that π_{AliceBob} is not valid. In Section 4 we discuss how to leverage epistemic planning to compute discrepancy resolving plans.

Modelling the inform action in this way enforces truthful communication, since its precondition is that Alice believe $\neg$*at*(Kit1, RoomA). Moreover, we assume that agents believe that other agents find their communications trustworthy (see discussion of trust by Fabiano et al. (2021)).

The i-aligned discrepancy resolving plan π' aligns Bob's beliefs with Alice's beliefs via a communication action. In contrast, the world-altering, j-aligned discrepancy resolving plan π'' resolves Alice's perceived discrepancy pertaining to π_{AliceBob} by aligning Alice's beliefs with Bob's. Assuming Alice is initially in the hallway,

$\pi'' = [\textit{move}(\text{Alice,Hall,RoomB}),$
$\textit{pckUp}(\text{Alice,Kit1,RoomB}),$
$\textit{move}(\text{Alice,RoomB,RoomA}),$
$\textit{dropOff}(\text{Alice,Kit1,RoomA})]$, such that
$\mathcal{I} \models \text{VAL}(\pi'',$
$B_{\text{Alice,Bob}} \text{VAL}(\pi_{\text{AliceBob}}, G) \wedge B_{\text{Alice}} \text{VAL}(\pi_{\text{AliceBob}}, G))$.

Intuitively, Alice aligns the environment with Bob's beliefs by bringing Kit1 from where it actually is (RoomB), to where Bob *believes* it to be (RoomA). Therefore, after Alice performs these actions, both Alice and Bob will believe that π_{AliceBob} is valid, thereby resolving the discrepancy. In many real world settings, it may either be undesirable or even not possible for agents to resolve discrepancies by means other than communication. For instance, if Alice were a virtual assistant, then it is likely she would only be able to communicate information to other agents (π'). However, if Alice were a robot, she could perhaps resolve Bob's discrepancy by executing π''. In Section 5 we show examples of these two discrepancy resolution 'modalities' in various domains.

There is also a *'higher-order' discrepancy* in our example. In particular, $\text{VAL}(\pi_{\text{AliceBob}}, G)$ is a discrepancy perceived by Alice between her beliefs and those of Mary about Bob's beliefs, where $\vec{v}$ is $\langle\text{Bob}\rangle$. That is, while Alice believes that Bob believes that π_{AliceBob} is valid (entailment (11)), she also believes that Mary believes that Bob believes that π_{AliceBob} is not valid (entailment (12)). This is because of Mary's false belief about Bob's beliefs about Kit1's location (entailments (8) and (9)). A possible discrepancy resolving plan is

$\pi''' = [\textit{inform}(\text{Alice,Mary,}B_{\text{Bob}}\textit{at}(\text{Kit1,RoomA}))]$,
such that $\mathcal{I} \models \text{VAL}(\pi''', B_{\text{Alice,Mary,Bob}} \text{VAL}(\pi_{\text{AliceBob}}, G)$
$\wedge B_{\text{Alice,Bob}} \text{VAL}(\pi_{\text{AliceBob}}, G))$.

Alice believes that after Mary learns that Bob believes that Kit1 is in RoomA, Mary will believe that Bob believes that π_{AliceBob} is valid (which resolves the discrepancy).

4 Computing Discrepancy Resolving Plans

As mentioned, epistemic planning combines automated planning and reasoning over the beliefs and knowledge of agents. In this section we present an algorithm that computes discrepancy resolving plans using epistemic planning tools and establish the soundness of our algorithm with a theorem. In its general form, the plan existence problem in multi-agent epistmeic planning (MEP) is undecidable (Bolander and Andersen 2011). However, the epistemic planner we use in this work, RP-MEP (Muise et al. 2015b, 2022), operates over a decidable and fairly expressive fragment of epistemic logic. In particular, (1) RP-MEP operates over PEKBs (and hence is not able to reason with disjunctive beliefs), (2) reasoning in RP-MEP is done from the perspective of a single agent, and (3) an upper bound is set on the depth of nested belief reasoning. To compute solutions for MEP problems, RP-MEP encodes a MEP problem as a classical^{+}[4] planning

[4]We refer to a classical^{+} planning problem as a classical planning problem augmented with ADL (Pednault 1989) features, most

problem and augments actions in the domain with conditional effects that enforce the KD45 axioms. The classically encoded MEP problem can then be given to an off-the-shelf classical planner. We appeal to this classical encoding of MEP problems in our computation. Exposition on classical planning is given in Appendix B.

Algorithm 1 accepts as input a tuple $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G\rangle$, where $Q = \langle \mathcal{P}, \mathcal{A}, Ag\rangle$ is a domain, $i, j \in Ag$ are agents, $\vec{v}$ is a tuple of agents in Ag, $\mathcal{I}$ is an initial state, π is a plan, and G is a goal, and returns a discrepancy resolving plan for it. Ideally, we would compute the validity formula for plan π and goal G, $\text{VAL}(\pi, G)$, and provide the epistemic goal specified in Definition 6 to an epistemic planner. However, RP-MEP cannot directly solve such goals due to the disjunction in the goal expression and the possible disjunction in $\text{VAL}(\pi, G)$.

Instead, we appeal to RP-MEP's encoding, which allows us to compute the validity formula and formulate the discrepancy resolution goal in a *classical*$^+$ *planning* setting. To this end, in Line 3, the CLASSENCODEMEP function returns a classical encoding of the initial state $\mathcal{I}$, MEP domain Q, the plan π and goal G, $\langle \mathcal{F}, \mathcal{I}', O, \pi_c, G_c\rangle$, where $\mathcal{F}$ is a set of propositional fluents representing each RML in the domain Q, $\mathcal{I}'$ is the classically encoded initial state $\mathcal{I}$, O is a set of classically encoded operators corresponding to the set of actions $\mathcal{A}$ in Q, π_c contains operators from O (corresponding to actions in $\mathcal{A}$ from π), and G_c corresponds to G and is expressed using propositional fluents from $\mathcal{F}$. We implement this function using RP-MEP's machinery which does not require any modification. Appendix C provides details of the encoding as well as the definitions of $\mathcal{C}()$ and $\mathcal{D}()$, which are mapping functions from RMLs in Q to propositional fluents in $\mathcal{F}$ (and vice versa).

Next, given the classical$^+$ planning domain $\langle \mathcal{F}, O\rangle$, the plan π_c, and the goal G_c, in Line 4 the COMPUTEPLANVALIDITYFORMULA function returns the formula $\phi = \text{VAL}_c(\pi_c, G_c)$, where VAL_c is the validity formula in a classical$^+$ planning setting, using an implementation of the regression operator REG for classical planning with conditional effects (Rintanen 2008). To force the classical planner to generate a discrepancy resolving plan, we formulate a goal that includes ϕ. Due to the possible disjunction in ϕ, and since RP-MEP does not support disjunctive belief, we formulate this goal using the *disjuncts* ϕ_d of DNF(ϕ) and DNF($\neg\phi$), where each ϕ_d is a conjunction. Specifically, in Line 6 CALLCLASSICALPLANNER tasks a classical planner that supports conditional effects and disjunctive goals with solving the classical$^+$ planning problem $\langle \mathcal{F}, \mathcal{I}', G', O\rangle$, where G' is

$$\bigvee_{\phi_d \in \text{DNF}(\phi)} \left(\bigwedge_{\phi_{dc} \in \phi_d} \mathcal{C}(B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc})) \wedge \mathcal{C}(B_{i,\vec{v}}\mathcal{D}(\phi_{dc})) \right) \vee$$

$$\bigvee_{\phi_d \in \text{DNF}(\neg\phi)} \left(\bigwedge_{\phi_{dc} \in \phi_d} \mathcal{C}(B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc})) \wedge \mathcal{C}(B_{i,\vec{v}}\mathcal{D}(\phi_{dc})) \right),$$

ϕ_{dc} are conjuncts of ϕ_d, $B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc})$ and $B_{i,\vec{v}}\mathcal{D}(\phi_{dc})$ are RMLs, and $\mathcal{C}(B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc}))$ and $\mathcal{C}(B_{i,\vec{v}}\mathcal{D}(\phi_{dc}))$ are the corresponding propositional fluents in $\mathcal{F}$ created in the encoding process. Each ϕ_{dc} is a propositional fluent in $\mathcal{F}$ and $\mathcal{D}(\phi_{dc})$ is the corresponding RML in the MEP domain Q.

notably here conditional effects and disjunctive goals. The popular Fast Downward (Helmert 2006) planner supports these.

Algorithm 1

1: **procedure** RESOLVEDISCREPANCY($\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G\rangle$)
Given a tuple $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G\rangle$, return a plan π'.
2: $\pi' \leftarrow [\ ]$
3: $\langle \mathcal{F}, \mathcal{I}', O, \pi_c, G_c\rangle \leftarrow$ CLASSENCODEMEP($Q, \mathcal{I}, \pi, G$)
4: $\phi \leftarrow$ COMPUTEPLANVALIDITYFORMULA($\mathcal{F}, O, \pi_c, G_c$)
5: $G' \leftarrow$
$$\bigvee_{\phi_d \in \text{DNF}(\phi)} \left(\bigwedge_{\phi_{dc} \in \phi_d} \mathcal{C}(B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc})) \wedge \mathcal{C}(B_{i,\vec{v}}\mathcal{D}(\phi_{dc})) \right) \vee$$
$$\bigvee_{\phi_d \in \text{DNF}(\neg\phi)} \left(\bigwedge_{\phi_{dc} \in \phi_d} \mathcal{C}(B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc})) \wedge \mathcal{C}(B_{i,\vec{v}}\mathcal{D}(\phi_{dc})) \right)$$
6: $\pi' \leftarrow$ CALLCLASSICALPLANNER($\langle \mathcal{F}, \mathcal{I}', G', O\rangle$)
7: **return** π'
8: **end procedure**

4.1 Example

Let us illustrate the workings of Algorithm 1 using our example, where agent i is Alice, j is Bob, and $\vec{v}$ is empty. In Line 3, we classically encode $\langle Q, \mathcal{I}, \pi_{\text{AliceBob}}, G\rangle$, where G is *holding*(Bob,Kit1), and obtain the tuple $\langle \mathcal{F}, \mathcal{I}', O, \pi_c, G_c\rangle$. Then, in Line 4 we obtain the validity formula $\phi = \text{VAL}_c(\pi_c, G_c)$ via regression, where

$$\phi = \textit{at\_Kit1\_RoomA} \wedge \textit{at\_Bob\_Hall}$$

and DNF($\neg\phi$) is therefore

$$\neg\textit{at\_Kit1\_RoomA} \vee \neg\textit{at\_Bob\_Hall},$$

where *at_Kit1_RoomA*, *at_Bob_Hall*, and their negations are propositional fluents in $\mathcal{F}$. In Line 6 we task a classical planner with solving $\langle \mathcal{F}, \mathcal{I}', G', O\rangle$, where G' is

$$\begin{aligned}
&(\mathcal{C}(B_{\text{Alice,Bob}}\textit{at}(\text{Kit1, RoomA})) \wedge \mathcal{C}(B_{\text{Alice}}\textit{at}(\text{Kit1, RoomA}))\\
&\quad \wedge \mathcal{C}(B_{\text{Alice,Bob}}\textit{at}(\text{Bob, Hall})) \wedge \mathcal{C}(B_{\text{Alice}}\textit{at}(\text{Bob, Hall})))\\
&\vee(\mathcal{C}(B_{\text{Alice,Bob}}\neg\textit{at}(\text{Kit1, RoomA}))\\
&\quad \wedge \mathcal{C}(B_{\text{Alice}}\neg\textit{at}(\text{Kit1, RoomA})))\\
&\vee(\mathcal{C}(B_{\text{Alice,Bob}}\neg\textit{at}(\text{Bob, Hall})) \wedge \mathcal{C}(B_{\text{Alice}}\neg\textit{at}(\text{Bob, Hall}))),
\end{aligned}$$

corresponding to one disjunct in DNF(ϕ) and two disjuncts in DNF($\neg\phi$). $\textit{at}(\text{Kit1, RoomA}) = \mathcal{D}(\textit{at\_Kit1\_RoomA})$. We discuss in the next section how the planner, depending on what actions Alice has at her avail, will either generate the communicative discrepancy resolving plan π' (where Alice informs Bob about Kit1's location) or the world-altering plan π'' (where Alice moves Kit1 to where Bob believes it to be), both discussed in the previous section. Both plans satisfy one of the disjuncts of G' and are therefore a solution for $\langle \mathcal{F}, \mathcal{I}', G', O\rangle$.

Finally, the input to Algorithm 1 includes a plan π and agents i and j. To resolve discrepancies between agent i's beliefs and the beliefs of multiple agents, Algorithm 1 is called multiple times, resulting in multiple discrepancy resolving plans, one for each agent. In our example, to resolve

discrepancies between (1) Alice and Bob and (2) Alice and Mary, we would call Algorithm 1 twice and generate two plans. Similarly, we would call Algorithm 1 twice to resolve discrepancies pertaining to two different plans.

4.2 Establishing the Soundness of Algorithm 1

Since we use RP-MEP in our implementation, we correspondingly appeal to a **R**estricted **P**erspectival **MEP** formulation to prove the soundness of Algorithm 1. In particular, an RP-MEP problem is a MEP problem from the perspective of a root agent $\star \in Ag$ and with a bounded depth of belief (Muise et al. 2022). Planning in RP-MEP is from the perspective of the root agent. See definition in Appendix C.

In our implementation, the discrepancy-resolving agent, agent i, is always the root agent. Since we wish to talk about the root agent's beliefs about other agents' beliefs about $\textsc{val}(\pi, G)$, we define $\textsc{val}(\pi, G)$ for the RP-MEP setting. To do so, we use the mapping function $\mathcal{D}()$, which can be applied compositionally to Boolean combinations of fluents (see Appendix C), and define $\textsc{val}(\pi, G) = \mathcal{D}(\textsc{val}_c(\pi_c, G_c))$. So $\textsc{val}(\pi, G)$ is a formula with the same structure as $\textsc{val}_c(\pi_c, G_c)$, but which replaces the propositional fluents in it with the corresponding RMLs in the MEP domain. Recall that $\textsc{val}_c(\pi_c, G_c)$ is the classical$^+$ planning validity formula, which ϕ is set to in Line 4 of Algorithm 1.

Theorem 1 (Soundness of Algorithm 1) *Suppose that a plan π' is returned by Algorithm 1 given $R = \langle\langle \mathcal{P}, \mathcal{A}, Ag\rangle, \mathcal{I}, i, j, \vec{v}, \pi, G\rangle$. Then π'' is a discrepancy resolving plan for R, where π'' is the plan comprising actions from $\mathcal{A}$ corresponding to the classically encoded operators in π'.*

Proof Sketch We want to show that if a plan π' is returned by RESOLVEDISCREPANCY given the tuple $\langle\langle \mathcal{P}, \mathcal{A}, Ag\rangle, \mathcal{I}, i, j, \vec{v}, \pi, G\rangle$, where $\langle\langle \mathcal{P}, \mathcal{A}, Ag\rangle, \mathcal{I}, G\rangle$ is an RP-MEP problem (where $\star \in Ag$ is the root agent), then the plan π'' corresponding to the plan π' is a discrepancy resolving plan for $\langle Q, \mathcal{I}, i, j, \vec{v}, \pi, G\rangle$ such that

$$\textsc{prog}(\pi'', \mathcal{I}) \models (B_{i,j,\vec{v}}\textsc{val}(\pi, G) \wedge B_{i,\vec{v}}\textsc{val}(\pi, G)) \vee (B_{i,j,\vec{v}}\neg\textsc{val}(\pi, G) \wedge B_{i,\vec{v}}\neg\textsc{val}(\pi, G)).$$

The plan π' is returned in Line 6 by the classical planner and therefore solves $\langle \mathcal{F}, \mathcal{I}', G', O\rangle$. By Lemma 2 (found in Appendix C), which builds on the correctness of RP-MEP's classical encoding, we have that

$$\textsc{prog}(\pi'', \mathcal{I}) \models \bigvee_{\phi_d \in \text{DNF}(\phi)} \left(\bigwedge_{\phi_{dc} \in \phi_d} B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc}) \wedge B_{i,\vec{v}}\mathcal{D}(\phi_{dc}) \right) \vee \bigvee_{\phi_d \in \text{DNF}(\neg\phi)} \left(\bigwedge_{\phi_{dc} \in \phi_d} B_{i,j,\vec{v}}\mathcal{D}(\phi_{dc}) \wedge B_{i,\vec{v}}\mathcal{D}(\phi_{dc}) \right).$$

It follows that

$$\textsc{prog}(\pi'', \mathcal{I}) \models [B_{i,j,\vec{v}}\mathcal{D}(\phi) \wedge B_{i,\vec{v}}\mathcal{D}(\phi)] \vee [B_{i,j,\vec{v}}\mathcal{D}(\neg\phi) \wedge B_{i,\vec{v}}\mathcal{D}(\neg\phi)].$$

Since $\phi = \textsc{val}_c(\pi_c, G_c)$, that can be rewritten (moving negation signs around using the definition of $\mathcal{D}()$) as

$$\textsc{prog}(\pi'', \mathcal{I}) \models [B_{i,j,\vec{v}}\mathcal{D}(\textsc{val}_c(\pi_c, G_c)) \wedge B_{i,\vec{v}}\mathcal{D}(\textsc{val}_c(\pi_c, G_c))] \vee [B_{i,j,\vec{v}}\neg\mathcal{D}(\textsc{val}_c(\pi_c, G_c)) \wedge B_{i,\vec{v}}\neg\mathcal{D}(\textsc{val}_c(\pi_c, G_c))].$$

Since we defined $\textsc{val}(\pi, G) = \mathcal{D}(\textsc{val}_c(\pi_c, G_c))$ in the RP-MEP setting, we are done. The full proof is in Appendix C.

5 Experimental Evaluation

In this section, we present the results of our evaluation, where we set out to (1) demonstrate that epistemic planning tools can be used to compute discrepancy resolving plans with different modalities (i.e., with communicative and/or world-altering actions) in various domains; and (2) evaluate the impact of increased depth of nested belief on Algorithm 1's runtime. Code can be found in bit.ly/3Kv1UDj.

5.1 Experimental Setup

To satisfy our objectives, we ran Algorithm 1 to generate discrepancy resolving plans in the following domains:

- **BlocksWorld for Teams** (BW4T) – an abstraction of a search & rescue domain, modelling our running example
- **Corridor** – an epistemic planning benchmark with epistemic goals such as selective communication of a secret. Interestingly, discrepancies may be resolved by closing doors to prevent agents from overhearing secrets.
- 7 **IPC** domains (e.g., Driverlog, Depots, Logistics)

To evaluate the impact of the required depth of nested belief, d, and number of agents, $|Ag|$, each domain (aside from *Corridor*) includes instances with $d = \{2, 3, 5\}$ (with 2, 3, and 5 agents, resp). When $d = 2$, a 'first order' discrepancy is resolved (e.g., where Alice resolves a discrepancy pertaining to Bob's beliefs about plan validity) and when $d = \{3, 5\}$, a 'higher order' discrepancy is resolved (e.g., where Alice resolves a discrepancy pertaining to Mary's beliefs about Bob's beliefs (about ...)).

We experimented with different discrepancy resolution modalities by conducting a simple ablation study where we created 3 versions of each problem instance where either: (1) no modifications were made to the problem instance; (2) a subset of ontic actions was manually removed (such that the discrepancy-resolving agent, i, cannot make certain changes to the environment to resolve discrepancies); or (3) a subset of communicative actions was removed (such that agent i cannot communicate with agent j to resolve discrepancies).

The various domains used in our experiments are described in detail in Appendix D. All domains were encoded using a file format used by RP-MEP called PDKB Domain Description Language (PDKBDDL) (a variant of the Planning Domain Definition Language (PDDL) (McDermott et al. 1998)) that can encode MEP problems, including nested agent beliefs. More details can be found in Appendix E and in (Muise et al. 2022). All problem instances across all domains were modelled as tuples comprising a domain, an initial state, a (possibly empty) tuple of agents, a plan, and a goal, and given to Algorithm 1. We encoded

Domain	d	Time (s)	$\lvert\pi'_{\mathbf{c}}\rvert/\lvert\pi'_{\mathbf{o}}\rvert$	Domain	d	Time (s)	$\lvert\pi'_{\mathbf{c}}\rvert/\lvert\pi'_{\mathbf{o}}\rvert$	Domain	d	Time (s)	$\lvert\pi'_{\mathbf{c}}\rvert/\lvert\pi'_{\mathbf{o}}\rvert$
BW4T	2	1.93	2 / 6	IPC - Driverlog	2	1.51	3 / 6	IPC - Logistics	2	47.19	2 / 8
BW4T	3	2.31	2 / –	IPC - Driverlog	3	3.77	3 / –	IPC - Logistics	3	59.33	2 / –
BW4T	5	1552.49	2 / –	IPC - Driverlog	5	472.97	3 / –	IPC - Logistics	5	MO	– / –
BW4T (EG)	3	3.16	2 / –	IPC - Gripper	2	1.33	2 / 5	IPC - Zeno	2	22.49	3 / 7
Corridor (EG)	3	2.94	2 / –	IPC - Gripper	3	4.47	2 / –	IPC - Zeno	3	51.43	3 / –
—	—	—	—	IPC - Gripper	5	572.78	2 / –	IPC - Zeno	5	MO	– / –
IPC - Depots	2	20.93	3 / 8	IPC - Rovers	2	1.39	3 / 6	IPC - Satellite	2	2.38	2 / 6
IPC - Depots	3	37.41	3 / –	IPC - Rovers	3	2.88	3 / –	IPC - Satellite	3	4.25	2 / –
IPC - Depots	5	4028.12	3 / –	IPC - Rovers	5	1428.50	3 / –	IPC - Satellite	5	807.63	2 / –

Table 1: The depth of nested belief significantly increases Algorithm 1's runtime. We report the average runtime in seconds for Algorithm 1. d is the required depth of nested belief and EG signifies that problems in the domain involve an epistemic goal. $|\pi'_{\mathbf{c}}|$ and $|\pi'_{\mathbf{o}}|$ are the average number of actions in plans returned by Algorithm 1 that resolve discrepancies via communication or world-altering actions, respectively. MO means that the planner or classical encoding ran out of memory.

the initial state for each problem instance such that agent i perceives a number of discrepancies between its beliefs and those of agent j, where one of the discrepancies is the validity formula of the plan π in the tuple given to Algorithm 1.

To implement Algorithm 1 we make use of the latest version of RP-MEP[5]. For every problem instance, RP-MEP is given in Line 3 PDKBDDL files and outputs classically encoded PDDL files. In Line 6, the Fast Downward planner (Helmert 2006) is given the encoded PDDL files and called with an admissible heuristic that supports conditional effects and disjunctive goals, to ensure optimal plans are computed. We also make use of the SymPy Python library to convert regression formulae to DNF and to compute their negation.

All plans given to Algorithm 1 were pre-computed using RP-MEP's machinery that allows for agents to *project to reason as other agents* and predict how they would achieve a certain goal (see discussion of *'Agent Projection'* in the RP-MEP repository and (Muise et al. 2015b, Sec. 5)).

5.2 Results

Table 1 summarizes the results for the various domains. The table shows the average runtime (in seconds) for Algorithm 1 (using RP-MEP) over **10** problem instances (and the 3 versions of each) of the respective domain. MO means that the planner or classical encoding ran out of memory. d is the depth of nested belief. The variances for the runtime for the set of problem instances for each domain (and value of d) ranged 0.07-0.45. The low variance is due to the planner's runtime being fairly similar for all problem instances in a certain domain and with a certain value of d. Finally, the $|\pi'_{\mathbf{c}}|$ and $|\pi'_{\mathbf{o}}|$ values are the average number of actions in plans returned by Algorithm 1 that resolve discrepancies via communication or world-altering actions, respectively, across all problem instances in the domain. All discrepancy resolving plans consisted of 1-8 world-altering or inform actions.

Different modalities of discrepancy resolution In our ablation study, as expected, when removing a subset of the ontic actions (such that the discrepancy-resolving agent cannot resolve discrepancies by changing the environment) the planner **only** found discrepancy resolving plans comprising communicative actions. Similarly, when removing a subset of communicative actions, the planner **only** found discrepancy resolving plans that involve world-altering actions (e.g., Alice moving Kit1 to RoomA in our example).

In the unmodified domains, the modality used depended on the length of the various achievable discrepancy resolving plans. For example, if a discrepancy resolving plan that involves communicating with agent j was shorter than a plan that involves moving an object between two rooms, then the planner chose the former. Overall, the planner found discrepancy resolving plans that involved communication (rather than environment alteration) in **74%** of problem instances. This is due to a bias in the way the domains were created in that we simplify inter-agent communication which typically only requires a single inform action, whereas altering the environment in order to resolve discrepancies typically requires additional actions. We note that our ablation-based approach to constraining the type of discrepancy resolving plan the planner can compute requires domain knowledge and manual effort. More generally, Φ can be specified appropriately. More details are in Appendix E.

Impact of d on runtime Table 1 shows that d, the depth of nested belief, and the number of agents $|Ag|$ that grows commensurate with d, significantly *increase* our algorithm's runtime. This is because the number of new fluents introduced during RP-MEP's encoding process is exponential in d and $|Ag|$ (Muise et al. 2022). When d is sufficiently high (as also observed by Muise et al.), some cells in Table 1 (where $d = 5$) read 'MO', i.e., either the compilation or the planner ran out of memory (with 32 GB RAM) because of the large number of fluents created during the compilation. Reflecting on their results, Muise et al. (2022, p.16) aptly say that: *"...the majority of interesting use cases we have found for planning with nested belief is restricted to depth [2-3]."* This is also true in our discrepancy resolution setting.

Finally, previous work empirically showed similar results concerning RP-MEP and also that the performance of some epistemic planners is not affected by the value of d (Le et al. 2018; Shvo et al. 2020). The application of these planners to discrepancy resolution could be investigated.

[5]https://github.com/QuMuLab/pdkb-planning

6 User Study

In the previous section we demonstrated that epistemic planning tools can be used to compute discrepancy resolving plans in a number of domains. However, these results are not necessarily a testament to the efficacy of our approach in the *presence of humans*. As such, we conducted a user study to evaluate the ability of our approach to resolve participants' misconceptions. This research was approved by the Institutional Review Board (IRB) at the authors' university. Specifically, we set out to test the following hypotheses:

H1: Participants will be more likely to generate a valid plan to achieve their goal when presented with information *derived from a discrepancy resolving plan*, compared to the likelihood of generating a valid plan prior to receiving the information.

H2: Participants will be more likely to correctly predict another agent's plan when presented with information *from a discrepancy resolving plan*, compared to their prediction prior to receiving the information.

To test these hypotheses, participants were told that they are part of an emergency response team whose members must communicate with one another and obtain various items. Participants were told that they are partnered with a virtual assistant meant to provide decision support, and were presented with two scenarios, mirroring two of the BW4T scenarios used in our evaluation and discussed in Appendix D.1. Initially, participants were given very limited and partially incorrect information, causing discrepancies and allowing us to control for the factors that impact participants' reasoning. For instance, in the first scenario participants were told that the supply tent is at the east end of the base, when in fact it was at the west end. Participants' feedback indicated that our controlled setting ensured that participants initially generated an invalid plan in the first scenario and incorrectly predicted their teammate's plan in the second scenario. This is aligned with the initial states in our evaluation (and running example) where some agents initially have false beliefs that cause discrepancies that need resolution.

In each scenario participants were given information by the virtual assistant. Using RP-MEP, we generated one discrepancy resolving plan (comprising only inform actions) for each scenario and the assistant's communication was simply a natural language representation of the inform actions in the discrepancy resolving plan (e.g., *"SupplyTent is at BaseWestEnd"*). Rephrased, H1 and H2 posited that the information given to participants would be sufficient to resolve the initial discrepancies we created and enable participants to perform better plan generation and prediction.

We had a total of 40 participants who were recruited via Amazon Mechanical Turk and were paid upon completing the questionnaire via SurveyMonkey. Participants had no prior knowledge about the study. Here we present a summary of the study's results; a detailed account of our method and results can be found in Appendix F.

Results In the first scenario, after receiving information from the assistant, all 40 participants generated a valid plan to achieve their goal, compared to 0 participants who did so prior to receiving the information. Next, a McNemar's test determined that participants' predictions about their teammate's plan (in the second scenario) after receiving information from the assistant were significantly more accurate than their predictions prior to receiving this information (95% compared to 5%, $p < .001$). These results support both **H1** and **H2** and show promise for human decision support.

7 Related Work

XAIP. Typically in Explainable AI Planning (XAIP) (Hoffmann and Magazzeni 2019; Chakraborti, Sreedharan, and Kambhampati 2020) – a special case of the general task of explanation generation (e.g., Miller 2019; Shvo, Klassen, and McIlraith 2020) – a planning agent is tasked with explaining some aspect of plan generation or execution (e.g., optimality or validity). Much work in XAIP has focused on allowing a planning agent to explain some aspect of its plan *without* considering potential model differences between the planning agent's model and that of the the recipient of the explanation (the *explainee*), typically the user of the planning system (e.g., Eifler et al. 2020). In contrast, our work emphasizes the need to consider the (possibly incomplete and incorrect) beliefs held by the explainee.

Indeed, a growing body of extant work has promoted this exact view. Chakraborti et al. (2017) have termed explanations that do not consider the explainee's perspective *soliloquies* and argued that planning agents offering explanations should eschew soliloquies and instead consider the possibly disparate model held by the explainee. To realize these desiderata, Chakraborti et al. formulated the *model reconciliation problem*, with a large body of work continuing this line of research (Sreedharan, Chakraborti, and Kambhampati 2021). In addition, Vasileiou, Previti, and Yeoh (2021) proposed a logic-based framework for model reconciliation that operates over two knowledge bases – of an explainer and an explainee. Relatedly, Miller (2021) – building on Halpern and Pearl (2005) – proposed to use structural causal models to generate contrastive explanations, with application potential in XAIP. Finally, Sreedharan et al. (2020) leveraged a simplified version of RP-MEP's compilation to classical planning to generate explicable plans as well as plan explanations delivered via implicit and/or explicit communication. Viewed through the lens of our work, this body of work can be seen to enable agents to use Theory of Mind-like reasoning and resolve discrepancies in XAIP settings.

Our work goes beyond extant literature by broadening the role of Theory of Mind in such settings. Namely, by appealing to multi-agent epistemic logic and epistemic planning, our work supports a unique variety of settings requiring complex Theory of Mind reasoning. Specifically, our works enables agents to (1) reason about the *nested beliefs* of other agents and resolve 'higher-order' discrepancies regarding plan validity; and (2) correct misconceptions pertaining to the validity of plans pursuant to *epistemic goals*.

BDI. Work on Belief-Desire-Intention (BDI) agents and architectures also explored the role of beliefs in explanation (e.g., Broekens et al. 2010). The explicit modelling of beliefs allows a BDI agent to explain its plans and goals in terms of its beliefs. However, these works did not appeal to epistemic planning and did not consider multi-agent settings

where agents may need to resolve discrepancies pertaining to other agents' beliefs about the validity of plans.

DEL. Finally, various fragments of Dynamic Epistemic Logic (DEL) have been used in epistemic planning (e.g., Bolander and Andersen 2011; Le et al. 2018). DEL often utilizes the modality $[\alpha]$ in formulae such as $[\alpha]\phi$ to express that ϕ is true after action α has occurred. Since the modality $[\alpha]$ is inherent to the language, agents can hold beliefs about such formulae as $[\alpha]\phi$ and thus about plan validity, and as part of future work we will explore the use of an epistemic logic that includes this modality.

8 Concluding Remarks

In this work, we examined how planning agents can use Theory of Mind to resolve discrepancies between their beliefs and the beliefs of other agents regarding plan validity. Our formulation appeals to epistemic logic and allows agents to reason about the nested beliefs of other agents and repair beliefs that give rise to plan validity discrepancies. We realized our approach using epistemic planning and showed how epistemic planning tools may be used to resolve discrepancies in various domains. A study showcased the ability of our approach to resolve misconceptions held by humans.

As part of future work, we wish to explore synergies with the body of work on implicit coordination (Engesser et al. 2017). Additionally, in some cases (e.g., two agents needing to hear the same piece of information) aggregating all pairwise communications (e.g., by running Algorithm 1 for each agent) might not be desirable. Algorithm 1 can be adapted to instead encourage a global optimization.

The objective of our study was modest – validate our approach by showing that discrepancy resolving plans generated by Algorithm 1 contain useful information for humans. To produce a user study with less modest objectives we are currently investigating related issues with a humanoid robot where we are able to test participants' perceptions of the robot's helpfulness in real-world settings.

Work on model reconciliation, discussed in Section 7, can handle misconceptions that stem from different agents holding different views about action definitions. While epistemic planning can in general address such settings, in our current regression-based approach we assume that the regression formula $\text{REG}(\pi, G)$ is shared by all agents. However, our approach could be extended by establishing satisfiability of different regression formulas.

Finally, Algorithm 1 accepts, as part of its input, a plan π and goal G. π and G may be obtained using *plan recognition*, where an observing agent attempts to predict an observed agent's plan and goal given a sequence of observations about the world and the behavior of the observed agent (e.g., Kautz 1987). However, in plan recognition observations often correspond to multiple plan or goal hypotheses. There are a number of ways to deal with uncertainty about other agents' plans, including collapsing (some of) the uncertainty via abstraction, exploiting probabilistic plan recognition (e.g. Ramírez and Geffner 2010), and appealing to ideas from conformant planning. To this last point, rather than commit to a single plan hypothesis (and risk being wrong), we might extend our current approach to resolve discrepancies wrt the entire set of possible plans the agent may be pursuing, without myopically treating each plan individually, one after the other. Another alternative is to use probabilistic plan recognition techniques, generate a probability distribution over possible plans and goals, and, at the risk of being wrong, resolve discrepancies pertaining to the most probable plan and goal. Integrating plan recognition and discrepancy resolution is left to future work.

Acknowledgements

We thank our anonymous reviewers for their constructive feedback and helpful questions, and for motivating us to eschew "Assumption 1". We gratefully acknowledge funding from the Natural Sciences and Engineering Research Council of Canada (NSERC), the Canada CIFAR AI Chairs Program, and Microsoft Research. Finally, we thank the Schwartz Reisman Institute for Technology and Society.

References

Baier, J. A.; and McIlraith, S. A. 2006. Planning with first-order temporally extended goals using heuristic search. In *Proc. of the 21st National Conference on Artificial Intelligence (AAAI)*, 788–795.

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

Broekens, J.; Harbers, M.; Hindriks, K.; Van Den Bosch, K.; Jonker, C.; and Meyer, J.-J. 2010. Do you get it? User-evaluated explainable BDI agents. In *Proc. of the 8th German conference on Multiagent System Technologies*.

Chakraborti, T.; Sreedharan, S.; and Kambhampati, S. 2020. The emerging landscape of explainable automated planning & decision making. In *Proc. of the 29th Int'l Joint Conference on Artificial Intelligence (IJCAI)*.

Chakraborti, T.; Sreedharan, S.; Zhang, Y.; and Kambhampati, S. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In *Proc. of the 26th Int'l Joint Conference on Artificial Intelligence (IJCAI)*, 156–163.

Eifler, R.; Cashmore, M.; Hoffmann, J.; Magazzeni, D.; and Steinmetz, M. 2020. A new approach to plan-space explanation: Analyzing plan-property dependencies in oversubscription planning. In *Proc. of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 9818–9826.

Engesser, T.; Bolander, T.; Mattmüller, R.; and Nebel, B. 2017. Cooperative epistemic multi-agent planning for implicit coordination. In *Proc. of the 9th Workshop on Methods for Modalities (EPTCS)*.

Fabiano, F.; Burigana, A.; Dovier, A.; and Pontelli, E. 2020. EFP 2.0: a multi-agent epistemic solver with multiple e-state representations. In *Proc. of the 30th Int'l Conference on Automated Planning and Scheduling (ICAPS)*, 101–109.

Fabiano, F.; Burigana, A.; Dovier, A.; Pontelli, E.; and Son, T. C. 2021. Multi-agent epistemic planning with inconsistent beliefs, trust and lies. In *Proc. of the 18th Pacific Rim Int'l Conference on Artificial Intelligence (PRICAI)*.

Fagin, R.; Halpern, J. Y.; Moses, Y.; and Vardi, M. 2004. *Reasoning about knowledge*. MIT press.

Fritz, C.; and McIlraith, S. A. 2007. Monitoring plan optimality during execution. In *Proc. of the 17th Int'l Conference on Automated Planning and Scheduling (ICAPS)*.

Halpern, J. Y.; and Pearl, J. 2005. Causes and explanations: A structural-model approach. Part II: Explanations. *The British journal for the philosophy of science*.

Haslum, P. 2014. Personal Communication.

Helmert, M. 2006. The Fast Downward planning system. *Journal of Artificial Intelligence Research*, 26: 191–246.

Hoffmann, J.; and Magazzeni, D. 2019. Explainable AI planning (XAIP): overview and the case of contrastive explanation. *Reasoning Web. Explainable AI*, 277–282.

Huang, X.; Fang, B.; Wan, H.; and Liu, Y. 2017. A general multi-agent epistemic planner based on higher-order belief change. In *Proc. of the 26th Int'l Joint Conference on Artificial Intelligence (IJCAI)*.

Kautz, H. A. 1987. *A formal theory of plan recognition*. Ph.D. thesis, University of Rochester.

Kominis, F.; and Geffner, H. 2015. Beliefs in multiagent planning: from one agent to many. In *Proc. of the 25th Int'l Conference on Automated Planning and Scheduling (ICAPS)*, 147–155.

Lakemeyer, G.; and Lespérance, Y. 2012. Efficient reasoning in multiagent epistemic logics. In *Proc. of the 20th European Conference on Artificial Intelligence (ECAI)*, 498–503.

Le, T.; Fabiano, F.; Son, T. C.; and Pontelli, E. 2018. EFP and PG-EFP: Epistemic forward search planners in multi-agent domains. In *Proc. of the 28th Int'l Conference on Automated Planning and Scheduling (ICAPS)*.

Levesque, H. J. 1998. A completeness result for reasoning with incomplete first-order knowledge bases. In *Proc. of the 6th Int'l Conference of Knowledge Representation and Reasoning (KR)*, 14–23.

Liu, Y.; Lakemeyer, G.; and Levesque, H. J. 2004. A logic of limited belief for reasoning with disjunctive information. In *Proc. of the 9th Int'l Conference on Knowledge Representation and Reasoning (KR)*, 587–597.

McDermott, D.; Ghallab, M.; Howe, A.; Knoblock, C.; Ram, A.; Veloso, M.; Weld, D.; and Wilkins, D. 1998. PDDL — The Planning Domain Definition Language. Technical Report TR-98-003/DCS TR-1165, Yale Center for Computational Vision and Control.

Miller, T. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.*, 267: 1–38.

Miller, T. 2021. Contrastive explanation: A structural-model approach. *The Knowledge Engineering Review*, 36.

Miller, T.; and Muise, C. J. 2016. Belief update for proper epistemic knowledge bases. In *Proc. of the 25th Int'l Joint Conference on Artificial Intelligence (IJCAI)*, 1209–1215.

Muise, C.; Belle, V.; Felli, P.; McIlraith, S. A.; Miller, T.; Pearce, A. R.; and Sonenberg, L. 2022. Efficient multi-agent epistemic planning: Teaching planners about nested belief. *Artif. Intell.*, 302.

Muise, C.; Miller, T.; Felli, P.; Pearce, A. R.; and Sonenberg, L. 2015a. Efficient reasoning with consistent proper epistemic knowledge bases. In *Proc. of the 14th Int'l Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS)*, 1461–1469.

Muise, C. J.; Belle, V.; Felli, P.; McIlraith, S. A.; Miller, T.; Pearce, A. R.; and Sonenberg, L. 2015b. Planning over multi-agent epistemic states: A classical planning approach. In *Proc. of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, 3327–3334.

Pednault, E. P. D. 1989. ADL: Exploring the Middle Ground Between STRIPS and the Situation Calculus. In *Proc. of the 1st Int'l Conference of Knowledge Representation and Reasoning (KR)*, 324–332.

Petrick, R. P.; and Bacchus, F. 2002. A knowledge-based approach to planning with incomplete information and sensing. In *Proc. of the 6th Int'l Conference on Artificial Intelligence Planning and Scheduling (AIPS)*, 212–222.

Premack, D.; and Woodruff, G. 1978. Does the chimpanzee have a Theory of Mind? *Behavioral and brain sciences*.

Ramírez, M.; and Geffner, H. 2010. Probabilistic plan recognition using off-the-shelf classical planners. In *Proc. of the 24th AAAI Conference on Artificial Intelligence (AAAI)*.

Reiter, R. 2001. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press.

Rintanen, J. 2008. Regression for classical and nondeterministic planning. In *Proc. of the 18th European Conference on Artificial Intelligence (ECAI)*, 568–572.

Shvo, M.; Klassen, T. Q.; and McIlraith, S. A. 2020. Towards the role of Theory of Mind in explanation. In *The 2nd International Workshop on EXplainable and TRAnsparent AI and Multi-Agent Systems (EXTRAAMAS)*, 75–93.

Shvo, M.; Klassen, T. Q.; and McIlraith, S. A. 2022. Resolving misconceptions about the plans of agents via Theory of Mind (Technical Appendix). Technical Report CSRG-642, Department of Computer Science, University of Toronto. URL: https://bit.ly/3pPR03k.

Shvo, M.; Klassen, T. Q.; Sohrabi, S.; and McIlraith, S. A. 2020. Epistemic plan recognition. In *Proc. of the 19th Int'l Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS)*, 1251–1259.

Sreedharan, S.; Chakraborti, T.; and Kambhampati, S. 2021. Foundations of explanations as model reconciliation. *Artif. Intell.*, 301.

Sreedharan, S.; Chakraborti, T.; Muise, C.; and Kambhampati, S. 2020. Expectation-aware planning: a general framework for synthesizing and executing self-explaining plans for human-AI interaction. In *Proc. of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2518–2526.

Vasileiou, S. L.; Previti, A.; and Yeoh, W. 2021. On Exploiting Hitting Sets for Model Reconciliation. In *Proc. of the 35th AAAI Conference on Artificial Intelligence (AAAI)*.

Waldinger, R. 1977. Achieving Several Goals Simultaneously. In *Machine Intelligence 8*, 94–136. Ellis Horwood.