Expectation-Aware Planning: A Unifying Framework for Synthesizing and Executing Self-Explaining Plans for Human-Aware Planning *

Sarath Sreedharan[1], Tathagata Chakraborti[2], Christian Muise[2] and **Subbarao Kambhampati[1]**

[1]CIDSE, Arizona State University, Tempe, AZ 85281 USA
[2]IBM Research AI, Cambridge, MA, USA

ssreedh3@asu.edu, tchakra2@ibm.com, christian.muise@ibm.com, rao@asu.edu

Abstract

In this work, we present a new planning formalism called *Expectation-Aware* planning for decision making with humans in the loop where the human's expectations about an agent may differ from the agent's own model. We show how this formulation allows agents to not only leverage existing strategies for handling model differences like explanations (Chakraborti *et al.* 2017) and explicability (Kulkarni *et al.* 2019), but can also exhibit novel behaviors that are generated through the combination of these different strategies. Our formulation also reveals a deep connection to existing approaches in epistemic planning. Specifically, we show how we can leverage classical planning compilations for epistemic planning to solve *Expectation-Aware* planning problems. To the best of our knowledge, the proposed formulation is the first complete solution to planning with diverging user expectations that is amenable to a classical planning compilation while successfully combining previous works on explanation and explicability. We empirically show how our approach provides a computational advantage over our earlier approaches that rely on search in the space of models.

Introduction

One of the greatest challenges in designing agents that can work with humans is in making sure that the agents are capable of acting in a manner that is interpretable to the humans. A major barrier towards achieving fluent collaboration occurs when the human's expectations regarding the agent's capabilities and preferences differ from reality. Such knowledge asymmetry implies that even in cases where the agent is coming up with the best decisions it can, the human would not be able to agree to the quality of that plan. Previous works have proposed two strategies to handle this: (1) provide information that reconciles the model differences, either through explicit communication (Chakraborti *et al.* 2017) or by performing actions that convey robots capabilities (Kwon *et al.* 2018) (2) or by acting in a manner that aligns with human expectations (Zhang *et al.* 2017). While each of these are reasonable strategies on their own, for the agent to be truly effective we would want it to be capable of combining the strengths of each. While there exists some initial works in this direction (for example (Chakraborti *et al.* 2019b)), we are unaware of any existing works that are able to capitalize on the agent's ability to effect and leverage human expectations through behavior, or explicit communication, for these two purposes.

Our formulation, on the other hand, leads to what may be best described as *self-explaining plans* with the plan now containing actions that are responsible for explaining the rest of the plan. Such explanations may be delivered by purely communicative actions (thereby allowing for explanations as studied in (Chakraborti *et al.* 2017)) that are meant to update the human's mental model or task level actions that could also have epistemic side effects (thereby allowing for actions of the type studied in (Kwon *et al.* 2018)). Additionally, the framework allows for selecting plans that aligns with human expectations whenever possible. Our contributions are thus two-fold:

- We present the first unification of various threads of planning with differing human expectation: including acting in-accordance with the human expectation (explicability), bridging model asymmetry through implicit (epistemic effects of plan execution on the mental model) and explicit communication (explanations).
- We show how our formulation is complete (as compared to our previous attempt at a balanced approach in (Chakraborti *et al.* 2019b)) and also lends itself to compilation to classical planning. The latter provides significant computational advantage with respect to existing algorithms that search directly in the space of models.

Background

We will assume that the planning models used by both the human and the robot are represented as classical planning problems described by the tuple $\mathcal{M} = \langle F, A, I, G, C \rangle$ (Geffner and Bonet 2013), where F is the set of propositional fluents used to describe the planning task states, A the set of actions, I the initial state, G the goal. Each action $a \in A$ is further defined as a tuple $a = \langle \text{prec}^a, \text{adds}^a, \text{dels}^a \rangle$, where prec^a is its preconditions, and adds^a and dels^a are its add and delete effects. The precondition is a propositional formula defined over state fluents such that an action a can only be executed in a state S if $S \models \text{prec}^a$. The effects are generally of the form $c \rightarrow e$, where the antecedent repre-

*Updated Version.

sents the condition under which the effect e should be applied (where the fluent corresponding to e is set to true in the state if $c \rightarrow e$ is part of the add effects, and if it is part of the delete it is set to false).

Each action is associated with a cost $C(a)$. A plan or a sequence of actions $\pi = \langle a_1, ..., a_n \rangle$ is a valid solution of a planning problem $\mathcal{M}$ if $\pi(I) \models_{\mathcal{M}} G$ and $G \subseteq \pi(I)$. The cost of a plan is the sum of individual action costs, i.e. $C(\pi) = \sum_{i=1}^{n} C(a_i)$. A plan π is said to be optimal if there exist no valid plan π' such that $C(\pi') < C(\pi)$. We will use $\Pi^*_{\mathcal{M}}$ to represent the set of all optimal plans for $\mathcal{M}$.

Our setting involves an agent that makes decisions using its own model $\mathcal{M}_R = \langle F, A_R, I_R, G_R, C \rangle$ while a human evaluates the plan using their mental model $\mathcal{M}_H = \langle F, A_H, I_H, G_H, C \rangle$. For ease of discussion, we concentrate on the specific case where conditions for actions only consist of conjunction of positive literals and the actions have the same cost in both models. While the human is under the assumption that M_H is an accurate representation of the task at hand, the model could be different from $\mathcal{M}_R$ in terms of action definitions, the initial state, and the goal. This difference means that plans generated for the model $\mathcal{M}_R$ may have different properties in the mental model $\mathcal{M}_H$. For example, a plan π^* that is optimal in $\mathcal{M}_R$ may be considered suboptimal or even un-executable by the human.

When model asymmetry becomes a source of confusion for the observer, explaining the plan must involve bridging this gap. One of the ideas proposed by earlier works in *explanations as model reconciliation* (c.f (Chakraborti *et al.* 2017)) is that given a specific plan, the agent does not need to achieve complete reconciliation. Rather they can focus on providing enough information that the current plan has required properties (such as executability, optimality, etc.). When the agent is aware of M_H, it can use this knowledge to figure out the minimal (where minimality of explanations is defined with respect to an explanation cost C_E) information it needs to provide to achieve the required properties. For example, the problem of identifying explanations for establishing optimality of a given plan π thus becomes:

$$\text{argmin}_{\mathcal{E}}(C_E(\mathcal{E}))$$
$$\text{such that } \pi \in \Pi^*_{\mathcal{M}_H + \mathcal{E}}$$

where $\mathcal{E}$ is a set of model information about the agent to be provided to the user as explanation (this could include truth value of fluents in initial state, presence or absence of literals in preconditions/effects, etc.) and $\mathcal{M}_H + \mathcal{E}$ is the updated user model after the explanation. Note that our use of '+' operator does not imply that all model reconciliation explanations are additive as $\mathcal{E}$ could include information aimed at correcting user's misconceptions about additional effects or even additional actions that the robot is capable of. We will follow the conventions set in (Chakraborti *et al.* 2017) and focus on three main types of model updates:

(1) Turn a fluent p true or false in initial state (represented by the operator {add/remove}-p-from-I)

(2) Add or remove a fluent p from the precondition (also add or delete effect) list of an action a (represented by the operator {add/remove}-p-from-{prec/adds/dels}-of-a)

(3) Add or remove a fluent p from the goal list (represented by the operator {add/remove}-p-from-G)

This focuses on cases where the agent is explaining its plan to the human after generating it. The flip side would be to try generating plans that are tailored for the human model. This is referred to as explicable planning and the most basic version of this problem can be formulated as:

$$\text{argmin}_{\pi}(C(\pi))$$
$$\text{such that } \pi(I_R) \models G_R \text{ and } \pi(I_H) \models G_H$$

This computes a plan that is executable in the agent model and the human mental model with the lowest cost. Our approach is capable of both explaining its plans as well as choosing plans that align with the user expectations. Before delving into details, we briefly introduce the search and rescue domain from (Chakraborti *et al.* 2019b) which we will use as an illustrative example for the rest of the paper.

Our Running Example: Search & Rescue

A typical Urban Search and Rescue (USAR) scenario consists of an autonomous robot deployed to a disaster scene with an external commander who is monitoring its activities. Both agents start with the same model of the world (i.e the map of the building before the disaster) but the models diverge over time since the robot, being internal to the scene, has access to updated information about the building. This model divergence could lead to the commander incorrectly evaluating valid plans from the robot as sub-optimal or even unsafe. One way to satisfy the commander would be to communicate or explain changes to the model that led the robot to come up with those plans in the first place.

Figure 1 illustrates a scenario where the robot needs to travel from P1 to its goal at P17. The optimal plan expected by the commander is highlighted in grey in their map and involves the robot moving through waypoint P7 and follow that corridor to go to P15 and then finally to P16. The robot knows that it should in fact be moving to P2 – its optimal plan is highlighted in blue. This disagreement rises from the fact that the human incorrectly believes that the path from P16 to P17 is clear while that from P2 to P3 is blocked.

If the robot were to follow the explanation scheme established in (Chakraborti *et al.* 2017), it would stick to its own plan and provide the following explanation:

```
> remove-(clear p16 p17)-from-I
    (i.e. Path from P16 to P17 is blocked)
> add-(clear p2 p3)-to-I
    (i.e. Path from P2 to P3 is clear)
```

If the robot were to stick to a purely explicable plan (Zhang *et al.* 2017) then it can choose to use the passage through P5 and P6 after performing a costly clear_passage action (this plan is not optimal in either of the models).

Expectation-Aware Planning

We call the task of computing plans with the expectations of an external agent: "Expectation-Aware planning".

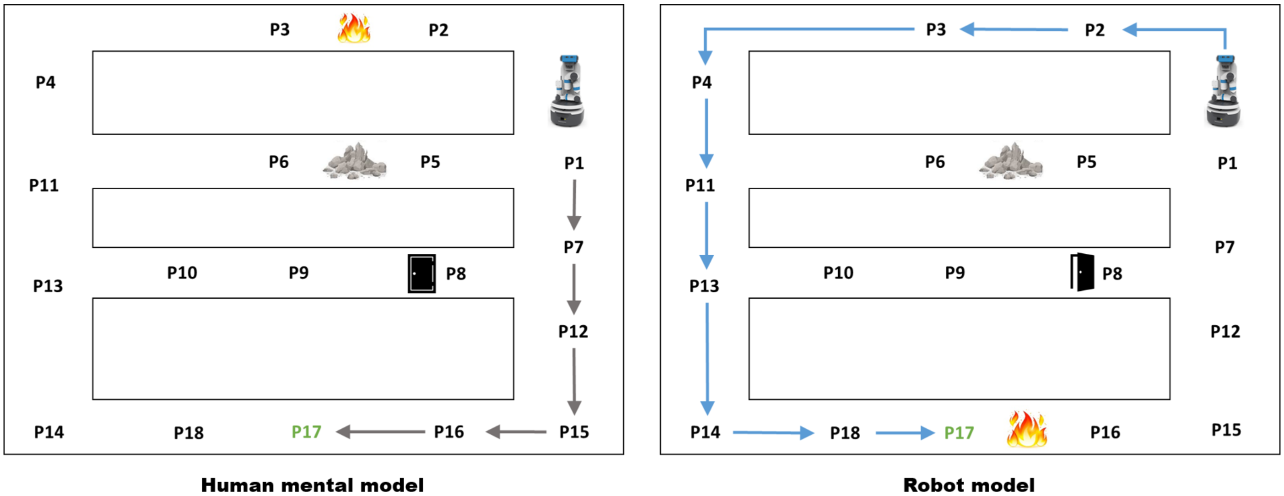

Figure 1: Illustration of the robot model and the corresponding mental model of the human. The robot starts at P1 and needs to go to P17. The human incorrectly believes that the path from P16 to P17 is clear and the one from P2 to P3 is blocked due to fire. Both agents know that there is movable rubble between P5 and P6 which can be cleared with a costly clear_passage action. Finally, in the mental model, the door at P8 is locked while it is unlocked in the model for the robot which cannot open unlocked doors.

Definition 1. *An* Expectation-Aware planning problem *(EA) is defined by the tuple* $\Psi = \langle \mathcal{M}_R, \mathcal{M}_H \rangle$*, where* $\mathcal{M}_R$ *is the robot model and* $\mathcal{M}_H$ *is the model ascribed to the robot by an observer. A solution to the problem* Ψ *is then given by the tuple* $\langle \mathcal{E}^\Psi, \pi_\Psi \rangle$*, where* $\mathcal{E}^\Psi$ *is a set of model updates for* $\mathcal{M}_H$ *consistent with* $\mathcal{M}_R$ *and* π_Ψ *a plan. The given solution is considered valid iff* $\pi_\Psi(I_{\mathcal{M}_R}) \models_{\mathcal{M}_R} G_{\mathcal{M}_R}$ *(it is valid in the agent model) as well as* $\pi_\Psi(I_{\mathcal{M}_H+\mathcal{E}^\Psi}) \models_{\mathcal{M}_H+\mathcal{E}^\Psi} G_{\mathcal{M}_H+\mathcal{E}^\Psi}$ *(valid in the updated mental model).*

This means that a solution to an expectation aware problem may consist of model information to be provided to the observer along with the plan that will be followed by the agent. In the USAR example, the optimal robot plan along with the two initial state updates, and the explicable plan with no model updates, would both be valid solutions.

At first glance, the need to keep track of both models and identifying the model changes may make the problem of solving EA planning problems considerably harder than the original decision making problem. However, we show that, in fact, finding a valid solution in this setting is no harder than identifying valid plans for classical planning problems:

Theorem 1. *For a given EA problem* $\Psi = \langle \mathcal{M}_R, \mathcal{M}_H \rangle$*, where both* $\mathcal{M}_R$ *and* $\mathcal{M}_H$ *are represented as classical planning problems, the problem of identifying a valid solution for* Ψ *is PSPACE-complete.*

Proof Sketch. The PSPACE-hardness of an EA is easy to establish since the problem of planning with just agent model can be mapped to a specific EA planning scenario where both agent and user have the same model. We can establish membership in PSPACE class by showing that there exist a sound and complete compilation from EA to a planning problem with conditional effects and disjunctive/negative preconditions that is linear in size of the original planning problems. We can then follow the same proof specified in (Bylander 1994) to show that the problem of plan existence is still in PSPACE for this class of planning problems. The exact details of the compilation along with the soundness and completeness proofs be discussed in Theorem 2. □

Self-Explaining Plans as Solutions to EA

One of the main challenges of compiling an EA problem to a traditional planning problems is to allow for a way to handle the identification of model updates and to account for the effect of these model updates on the user's expectation. A good way to go about this would be by acknowledging that that if the observer is actually watching the agent executing a plan, these explanations can delivered through and hence modeled as communicative or *explanatory actions*. These actions can, in fact, be seen as actions with epistemic effects in as much as they are aimed towards modifying the human mental model (knowledge state). This means that a solution to an EA planning problem can be seen as *self-explaining plans*, in the sense that some of the actions in the plan are aimed at helping people better understand the rest of it.

This puts EA planning squarely in the purview of epistemic planning, but the additional constraints enforced by the setting allow us to leverage relatively efficient methods to solve the problem at hand. These constraints include facts such as: all epistemic actions are public, modal depth is restricted to one, modal operators only applied to literals, for any literal the observer believes it to be true or false and the robot is fully aware of all of the observer beliefs.

Model updates in the form of epistemic effects of communication actions also open up the possibility of other actions having epistemic *side effects*. The definition of EA makes no claims as to how the model update information is delivered. It is quite possible that actions that the agent is performing to achieve the goal (henceforth referred to as task-level actions to differentiate it from primary epistemic communication actions) itself could have epistemic side-effects. This is something people leverage to simplify communication – e.g. one might avoid providing prior description of some skill they are about to use when they can simply demonstrate it. So one of our goals with the compilation is to allow for such epistemic side effects; a factor that has previously been not considered in any of the earlier works. This consideration also enables us to also capture task level constraints that may be imposed on the communication actions.

Compilation to classical planning. To support such self-explaining plans, we adopt a formulation that is similar to the one introduced in (Muise *et al.* 2015) to compile reasoning about epistemic states into a classical planning problem. In our setting, each explanatory action can be viewed as an action with epistemic effects. One interesting distinction to make here is that the mental model now not only includes the human's belief about the task state but also their belief about the robot's model. This means that the planning model will need to separately keep track of (1) the current robot state, (2) the human's belief regarding the current state, (3) how actions would effect each of these (as humans may have differing expectations about the effects of each action) and (4) how those expectations change with explanations.

Given the model reconciliation planning problem $\Psi = \langle \mathcal{M}_R, \mathcal{M}_H \rangle$, we will generate a new planning model $\mathcal{M}_\Psi = \langle F_\Psi, A_\Psi, I_\Psi, G_\Psi, C_\Psi \rangle$ as follows $F_\Psi = F \cup F_\mathcal{B} \cup F_\mu \cup \{\mathcal{G}, \mathcal{I}\}$, where $F_\mathcal{B}$ is a set of new fluents that will be used to capture the human's belief about the task state and F_μ is a set of meta fluents that we will use to capture the effects of explanatory actions and $\mathcal{G}$ and $\mathcal{I}$ are special goal and initial state propositions. We will use the notation $\mathcal{B}(p)$ to capture the human's belief about the fluent p. We are able to use a single fluent to capture the human belief for each (as opposed to introducing two new fluents $\mathcal{B}(p)$ and $\mathcal{B}(\neg p)$) as we are specifically dealing with a scenario where the human's belief about the robot model is fully known and human either believes each of the fluent to be true or false. In this case, we also do not require any of the additional rules that were employed in (Muise *et al.* 2015) to ensure that the state captures the deductive closure of the agent beliefs.

F_μ will contain an element for every part of the human model that can be changed by the robot through explanations. A meta fluent corresponding to a literal ϕ from the precondition of an action a takes the form of $\mu^+(\phi^{\text{prec}^a})$, where the superscript $+$ refers to the fact that the clause ϕ is part the precondition of the action a in the robot model (for cases where the fluent represents an incorrect human belief we will be using the superscript $-$).

For every action $a = \langle \text{prec}^a, \text{adds}^a, \text{dels}^a \rangle \in A_R$ and its human counterpart $a_h = \langle \text{prec}^{a_h}, \text{adds}^{a_h}, \text{dels}^{a_h} \rangle \in A_H$, we define a new action $a_\Psi = \langle \text{prec}^{a_\Psi}, \text{adds}^{a_\Psi}, \text{dels}^{a_\Psi} \rangle \in \mathcal{M}_\Psi$ whose precondition is given as:

$$\begin{aligned} \text{prec}^{a_\Psi} = \text{prec}^{a_R} &\cup \{\mu^+(\phi^{\text{prec}^a}) \to \mathcal{B}(\phi) \\ &\quad |\phi \in \text{prec}^{a_R} \setminus \text{prec}^{a_H}\} \\ &\cup \{\mu^-(\phi^{\text{prec}^a}) \to \mathcal{B}(\phi) | \phi \in \text{prec}^{a_H} \setminus \text{prec}^{a_R}\} \\ &\cup \{\mathcal{B}(\phi) | \phi \in \text{prec}^{a_H} \cap \text{prec}^{a_R}\} \end{aligned}$$

The important point to note here is that at any given state, an action in the augmented model is only applicable if the action is executable in robot model and the human believes the action to be executable. Unlike the executability of the action in the robot model (captured through unconditional preconditions) the human's beliefs about the action executability can be manipulated by turning the meta fluents on and off. The effects of these actions can also be defined similarly by conditioning them on the relevant meta fluent. In addition to these task level actions (represented by the set A_τ), we can also define explanatory actions (A_μ) that either add $\mu^+(*)$ fluents or delete $\mu^-(*)$.

Special actions a_0 and a_∞ that are responsible for setting all the initial state conditions true and checking the goal conditions are also added into the domain model. a_0 has a single precondition that checks for $\mathcal{I}$ and has the following add and delete effects:

$$\begin{aligned} \text{adds}^{a_0} = \{\top \to p \mid p \in I_R\} &\cup \{\top \to \mathcal{B}(p) \mid p \in I_H\} \\ &\cup \{\top \to p \mid p \in F_{\mu^-}\} \end{aligned}$$

$$\text{dels}^{a_0} = \{\mathcal{I}\}$$

where F_{μ^-} is the subset of F_μ that consists of all the fluents of the form $\mu^-(*)$. Similarly, the precondition of action a_∞ is set using the original goal and adds the proposition $\mathcal{G}$.

$$\text{prec}^{a_\infty} = G_R \cup \{\mu^+(p^G) \to \mathcal{B}(p) \mid p \in G_R \setminus G_H\} \cup$$
$$\{\mu^-(p^G) \to \mathcal{B}(p) \mid p \in G_H \setminus G_R\} \cup \{\mathcal{B}(p) \mid G_H \cap G_R\}$$

Finally the new initial state and the goal specification becomes $I_\mathcal{E} = \{\mathcal{I}\}$ and $G_\mathcal{E} = \{\mathcal{G}\}$ respectively. To see how such a compilation would look in practice, consider an action (move_from p1 p2) that allows the robot to move from point p1 to p2 only if the path is clear. The action is defined as follows in the robot model:

```
(:action move_from_p1_p2
    :precondition (and (at_p1)
                       (clear_p1_p2))
    :effect (and (not (at_p1)) (at_p2) ))
```

Let us assume the human is aware of this action but does not care about the status of the path (as they assume the robot can move through any debris filled path). In this case, the corresponding action in the augmented model and the relevant explanatory action will be:

```
(:action move_from_p1_p2
  :precondition
  (and
    (at_p1)
    (B((at_p1))) (clear_p1_p2)
    (implies
     (μ+_prec(move_from_p1_p2, (clear_p1_p2)))
     (B((clear_p1_p2))) ))
    :effect (and  (not (at_p1)) (at_p2)
            (not B(at_p1)) B(at_p2)) ))

(:action explain_μ+_prec_move_from_clear
  :precondition (and )
  :effect (and μ+_prec(move_from_p1_p2,
              (clear_p1_p2)) ))
```

Finally C_Ψ captures the cost of all explanatory and task level actions. For now we will assume that the cost of task-level actions are set to the original action cost in either robot or human model and the explanatory action costs are set according to C_E. Later, we will discuss how we can adjust the explanatory action costs to generate desired behavior.

We will refer to an augmented model that contains an explanatory action for each possible model updates and has no actions with effects on both the human's mental model and the task level states as the *canonical augmented model*.

Given an augmented model, let $\pi_\mathcal{E}$ be a plan that is valid for this model ($\pi_\mathcal{E}(I_\Psi) \subseteq G_\Psi$). From $\pi_\mathcal{E}$, we extract two types of information – the model updates induced by the actions in the plan (represented as $\mathcal{E}(\pi_\mathcal{E})$) and the sequence of actions that have some effect of the task state represented as $\mathcal{D}(\pi_\mathcal{E})$ (we refer to the output of $\mathcal{D}$ as the task level fragment of the original plan $\pi_\mathcal{E}$)). $\mathcal{E}(\pi_\mathcal{E})$ may contain effects from action in $\mathcal{D}(\pi_\mathcal{E})$. This brings us to our next theorem.

Theorem 2. *For a given EA problem $\Psi = \langle \mathcal{M}_R, \mathcal{M}_H \rangle$ the corresponding augmented model $\mathcal{M}_\Psi$ is a sound and complete formulation: (1) for every valid π for $\mathcal{M}_\Psi$ the tuple $\langle \mathcal{E}(\pi), \mathcal{D}(\pi) \rangle$ is a valid solution for Ψ and (2) for every valid solution $\langle \mathcal{E}^\Psi, \pi \rangle$, there exists a corresponding valid plan for π' for $\mathcal{M}_\Psi$ such that $\mathcal{D}(\pi') = \pi$ and $\mathcal{E}(\pi') = \mathcal{E}^\Psi$.*

Proof Sketch. The soundness of plans generated from $\mathcal{M}_\Psi$ are guaranteed by the construction of the model as all the preconditions of the actions in the updated user model have to be met in the current plan. To see why the formulation is complete, consider a solution $< \mathcal{E}^\Psi, \pi >$ for Ψ. From the procedure for constructing $\mathcal{M}_\Psi$ we know that there must exist an explanatory action for each possible model difference. This means that there should exist a sequence of explanatory actions $\langle a_1, .., a_k \rangle$ that results in the same model updates captured by $\mathcal{E}^\Psi$. It is easy to see that $\langle a_1, .., a_k \rangle + \pi$ is a valid plan for $\mathcal{M}_\Psi$ hence proving the assertion. □

The planner can automatically find positions of the explanatory actions, but to avoid any confusion that may arise from belief revisions on the users' end, we can enforce some common sense ordering like making any explanation related to an action to appear before the first instance of that action. This ordering will make sure that users are not confused about earlier action effects and also helps reduce branching, making planning more efficient.

Stage of Interaction and Epistemic Side Effects: One of the important parameters of the problem setting that we have yet to discuss is whether the explanation is meant for a plan that is proposed by the system (i.e the system presents a sequence of actions to the user) or are we explaining some plan that is being executed either in the real world or some simulation the user (observer) has access to. Even though the above formulation can be directly used for both scenarios, we can use the fact that the human is observing the execution of the plans to simplify the explanatory behavior by leveraging the fact that many of these actions may have epistemic side effects. This allows us to not explain any of the effects of the actions that the human can observe (for those effects we can directly update the believed value of the corresponding state fluent and the meta-fluent).[1] This is beyond the capability of any of the existing algorithms in this space of the explicability-explanation dichotomy.

This consideration also allows for the incorporation of more complicated epistemic side-effects wherein the user may infer facts about the task that may not be directly tied to the effects of actions. Such effects may be specified by domain experts or generated using heuristics. Once identified, adding them to the model is relatively straightforward as we can directly add the corresponding meta fluent into the effects of the relevant action. An example for a simple heuristic would be to assume that the firing of a conditional effect results in the human believing the condition to be true. For example, if we assume that the robot had an action (open_door_d1_p3) that had a conditional effect:

```
(when (and (unlocked_d1)) (open_d1))
```

Then in the compiled model, we can add a new effect:

```
(when (and (unlocked_d1))
    (and B(open_d1) B(unlocked_d1)))
```

Even in this simple case, it may be useful to restrict the rule to cases where the effect is conditioned on previously unused fluents so the robot does not expect the observer to be capable of regressing over the entire plan.

Optimality of the Agent

The compilation explored so far only takes into consideration the expectations the agent has about the safety of the plans (i.e the user would expect any plans generated to be valid and executable) and does not account for the user's expectation on whether the agent should act optimally. In the earlier example, if the agent just followed the plan that takes the robot through P5 and P6 with a clear_passage_P5_P6 action with no additional explanatory actions then the user may still be confused why the agent does not just follow the plan that involves going through P16 to P17 that it believes to be cheaper (marked in grey in the human's map).

Even in cases where the action costs are the same for the agent and the human, we cannot account for such expectations by merely generating optimal plans in the augmented model. For example, the optimal plan in the augmented model would be the one through P2 and P3 (the full plan is marked in blue in the robot map) with one extra explanatory action explain_μ_I^+_clear_P2_P3. While the above plan provides an explanation to ensure validity, ensuring the optimality of the resultant plan would require the agent to also explain that the passage from P16 to P17 is blocked, which would clearly be more expensive than choosing the valid plan for any non-zero cost for explanatory actions.

One approach to address this would be to prune all solutions where the task level fragment of the plan ($\mathcal{D}(\pi)$) is

[1]This means that when the plan is being executed, the problem definition should include the observation model of the human (which we assume to be deterministic). To keep the formulation simple, we ignore this for now. Including this additional consideration is straightforward for deterministic sensor models.

suboptimal in the updated human model. A simple way to enforce this would be to extend the planner to perform an optimality test for the current plan during the goal test. It may be possible to use more intelligent pruning to reduce the number of goal tests (e.g. one could leverage the fact that the optimality test never needs to be repeated for the same set of model updates) and we could design heuristics that take into account optimality aspects. In this paper, we adopt this simple approach as a first step towards modeling these novel behaviors.

Balanced Plans *vs.* Agent Optimal Plans

Even when generating plans that preserve the user's expectations about agent optimality, the agent could generate two types of plans: agent optimal plans (Chakraborti *et al.* 2017) or balanced plans (Chakraborti *et al.* 2019b). In the first scheme, the agent chooses to select self-explanatory plans whose task level fragment is going to be optimal in the original agent model and then choose the minimal explanations that justifies the optimality plan (i.e the plan is optimal in the user's updated model). Such explanations are referred to as Minimally Complete Explanation or MCE (the agent could also choose among the optimal plans the one that requires the cheapest MCE). An example would be choosing the plan highlighted in blue in robot model and then explaining that the path from P2 to P3 is clear and P16 to P17 is blocked. In the latter scheme, the agent could choose plans that are easiest to explain (here again we need to ensure that after the explanation the plan is optimal in the updated model). For example, in the USAR scenario if communication is expensive, it may be easier to choose the plan to move through P5 and P6 with a clear passage action since we only need to explain that the passage P16 to P17 is blocked.

In the first case, the agent is effectively prioritizing any loss of optimality over any overhead accrued by communicating the explanation, while in the second case the agent accounts for the cost of both the plan it is performing and the explanation cost (the cost of communication and possibly the computational overhead experienced by the user on receiving the explanation). By assigning explanatory costs to explanatory actions we are essentially generating balanced plans but there may be scenarios where the agent needs to stick to its optimal plan. We can generate such agent optimal plans by setting lower explanatory action costs. Before we formally state the bounds for explanatory costs, let us define the concept of *optimality delta* (denoted as $\Delta\pi_{\mathcal{M}}$) for a planning model, which captures the cost difference between the optimal plan and the second most optimal plan. More formally $\Delta\pi_{\mathcal{M}}$ can be specified as:

$$\begin{aligned}\Delta\pi_{\mathcal{M}} = \min\{v \mid\ & v \in \mathbb{R} \wedge \\ & \nexists \pi_1, \pi_2((0 < (C(\pi_1) - C(\pi_2)) < v) \\ & \wedge \pi_1(I_{\mathcal{M}}) \models_{\mathcal{M}} G_{\mathcal{M}} \wedge \pi_2(I_{\mathcal{M}}) \in \Pi^*_{\mathcal{M}}\}\end{aligned}$$

Theorem 3. *In a canonical augmented model $\mathcal{M}_\Psi$ for an EA planning problem Ψ, if the sum of costs of all explanatory actions is $\leq \Delta\pi_{\mathcal{M}_R}$ and if π is the cheapest valid plan for $\mathcal{M}_\Psi$ such that $\mathcal{D}(\pi) \in \Pi^*_{\mathcal{M}_\Psi + \mathcal{E}(\pi)}$, then:*

(1) $\mathcal{D}(\pi)$ is optimal for $\mathcal{M}_R$

(2) $\mathcal{E}(\pi)$ is the MCE for $\mathcal{D}(\pi)$

(3) There exists no plan $\hat{\pi} \in \Pi^*_R$ such that MCE for $\mathcal{D}(\hat{\pi})$ is cheaper than $\mathcal{E}(\pi)$, i.e. the search will find an the plan with the smallest MCE.

Proof Sketch. We observe that there exists no valid plan π' for the augmented model ($\mathcal{M}_\Psi$) with a cost lower than that of π and where the task level fragment ($\mathcal{D}(\pi')$) is optimal for the human model. Let's assume $\mathcal{D}(\pi) \notin \Pi^*_{\mathcal{R}}$ (i.e current plan's task-level fragment is not optimal in robot model) and let $\hat{\pi} \in \Pi^*_{\mathcal{R}}$. Now let's consider a plan $\hat{\pi}_{\mathcal{E}}$ for augmented model that corresponds to the plan $\hat{\pi}$, i.e, $\mathcal{E}(\hat{\pi}_{\mathcal{E}})$ is the MCE for the plan $\hat{\pi}$ and $\mathcal{D}(\hat{\pi}_{\mathcal{E}}) = \hat{\pi}$. Then the given augmented plan $\hat{\pi}_{\mathcal{E}}$ is a valid solution for our augmented planning problem $\mathcal{M}_\Psi$ (since the $\hat{\pi}_{\mathcal{E}}$ consists of the MCE for $\hat{\pi}$, the plan must be valid and optimal in the human model), moreover the cost of $\hat{\pi}_{\mathcal{E}}$ must be lower than π. This contradicts our earlier assumption hence we can show that $\mathcal{D}(\pi)$ is in fact optimal for the robot model.

Using a similar approach we can also show that no cheaper explanation exists for $\pi_{\mathcal{E}}$ and there exists no other plan with a cheaper explanation. □

Note that while it is hard to find the exact value of the optimality $\Delta\pi_{\mathcal{M}}$, it is guaranteed to be ≥ 1 for domains with only unit cost actions or $\geq (C_2 - C_1)$, where C_1 is the cost of the cheapest action and C_2 is the cost of the second cheapest action, i.e. $\forall a(C_{\mathcal{M}}(a) < C_2 \rightarrow C_{\mathcal{M}}(a) = C_1)$. Thus allowing us to easily scale the cost of the explanatory actions to meet this criteria.

Disallowing explicable plans that are too costly

There could be scenarios where forcing the agent to always choose plans that are optimal in the human model may not be the best strategy. For example, there may be cases where we would prefer the agent to deviate from the optimal expected plan if it results in significant gains. The penalty for deviating from the expected optimal plan should thus be another optimization criteria. To avoid complexities arising due to multi-criteria optimization, we will assume the agent is optimizing a single objective function of the form

$$C(\pi) + \alpha * (C_{\mathcal{M}_H + \mathcal{E}(\pi)}(\pi) - C^*_{\mathcal{M}_H + \mathcal{E}(\pi)})$$

where the second term basically captures the difference between the cost of the current plan in the resultant model and the cost of the expected plan and α is some scaling factor to allow linear combination of the two terms. Now with this new objective function we can define an *Optimally Balanced Plan* as a plan that is executable in both robot and the resultant human model and minimizes the above objective.

Definition 2. *For a problem Ψ, the tuple $\langle \mathcal{E}^\Psi, \pi^\Psi \rangle$ is said to be the optimally balanced solution if,*

1. $\pi^\Psi(I_R) \models G_R$.
2. $\pi^\Psi(I_H) \models_{\mathcal{M}_H} G_H$.
3. $\nexists \langle \hat{\mathcal{E}}, \hat{\pi} \rangle$, *such that the tuple satisfies (1.) and (2.), and* $C(\hat{\mathcal{E}}) + C_{\mathcal{M}_R}(\hat{\pi}) + \alpha * (C_{\mathcal{M}_h + \hat{\mathcal{E}}}(\hat{\pi}) - C^*_{\mathcal{M}_H + \hat{\mathcal{E}}}) < C(\mathcal{E}^\Psi) + C_{\mathcal{M}_R}(\pi^\Psi) + \alpha * (C_{\mathcal{M}_h + \mathcal{E}^\Psi}(\pi) - C^*_{\mathcal{M}_H + \mathcal{E}^\Psi})$.

To generate such optimal balanced plans, we need to relax the goal requirement that the final plan is optimal in the human model. Instead we can incorporate the inexplicability penalty into the reasoning about the plan, by assigning the cost of a_∞ (the goal action) to be α times the cost difference between the optimal plan in the human model and the current plan. When α is set to zero the problem would just identify the cheapest plan in the original robot model that is executable in the human model. We can also use this formulation to generate plans that are still guaranteed to be optimal in the human model by setting α higher than a threshold κ, where κ is some upper bound on plan length for the robot (that includes explanatory actions).

Evaluation

Since the nature of our solution has already been validated in literature through human factors evaluation – model reconciliation explanation has been studied in (Chakraborti *et al.* 2019a), balanced plans in (Chakraborti *et al.* 2019b), explicable plans in (Zhang *et al.* 2017; Kulkarni *et al.* 2019), and the use of physical actions to communicate robot model information in (Kwon *et al.* 2018) – we will focus on demonstrating the generality of our framework and studying empirically the performance of the compilation. The code can be found at `http://bit.ly/2Xb7OCp`.

Illustrative Example of Cost-Tradeoff

We start by demonstrating how our approach can lead to different solution by altering various costs associated with agent actions. Consider again the USAR domain described earlier: the models for the robot and the user is provided in the supplementary (the action for opening a door has an epistemic side effect that the observer would know that the door is unlocked). We start by assigning a cost of 10 to every robot action other than clear-rubble action (which is 50) and the move-through-door action (set to 20). We set the cost of communication action to 1 to start with. The solution produced corresponds to the blue plan in Figure 1.

explains_μ^{+}_{init}_clear_p2_p3 –>
explains_μ^{-}_{init}_clear_p16_p17 –>
move_p1_p2 –> move_p2_p3 –> move_p3_p4 –>
move_p4_p11 –>move_p11_p13 –> move_p13_p14
–> move_p14_p18 –> move_p18_p17

This plan includes the **optimal robot plan and corresponding MCE**. Now if we were to set the cost of communication actions to 100, we see the agent deviating to plans which on their own may not be optimal – e.g. a plan that involves opening the door at P8:

explains_μ^{-}_{init}_clear_p16_p17 –>
move_p1_p7 –> move_p7_p8 –> opendoor_p8_d1 –>
movethroughdoor_p8_p9_d1 –> move_p9_p10 –>
move_p10_p13 –> move_p13_p14 –> move_p14_p18 –>
move_p18_p17

Here the robot does not have to explicitly provide a separate explanation for the status of the door, but still needs to explain that the path from P15 to P16 is blocked. Note that this plan is an example of **a balanced plan** that leverages **epistemic side effects**. Now we go one step further and relax the need to assure optimality of the plan in the human model by changing it from a hard constraint to just a penalty. This gets us the exact same plan as above but without the explanation about the blocked corridor from P15 to P16, thus allowing a notion of soft explicability.

	New Compilation		Model Space Search	
	coverage	**runtime**	**coverage**	**runtime**
Blocksworld	13/15	**569.38**	13/15	2318.73
Elevator	**15**/15	**59.20**	1/15	3382.462
Gripper	5/15	2301.90	**6**/15	**2093.54**
Driverlog	**4**/15	**2740.38**	2/15	3158.59
Satellite	**2**/15	**3186.93**	0/15	3600

Table 1: Coverage and average runtime (sec) for explanations generated for a few standard IPC domains.

Runtime Complexity

Next we establish how our approach compares in terms of runtime to previous work. In particular, we will use as reference the optimistic and approximate version of the balancing approach in (Chakraborti *et al.* 2019b) that identifies only one optimal plan per search node and the search ends as soon as it finds a node where the optimal plan produced has the same cost as the robot plan and is executable in the robot model. This means all the solutions we generate are guaranteed to be better (in terms of cost) than that generated by the other. For comparison, we selected five IPC domains and for each domain, we created three unique models by introducing 10 random updates in the model, except in the case of Gripper and Driverlog where only 5 were removed. Each of these five domains were paired with five problem instances and then tested on each of the possible configurations. Each instance was run with a limit of one hour, all explanatory actions were restricted to the beginning of the plan and the cost of explanatory actions were set to be twice the cost of original action. Table 1 lists the time taken to solve each of these problems. For calculating the average runtime, we used 3600 secs as the stand in for the runtime of all the instances that timed out. We used h_max (admissible) as the heuristic for all the configurations.

The table shows that the new approach does better than the original method for generating balanced plans for most of the domains. Gripper seems to be the only domain, where model search seems to perform slightly better but this is also a domain that had the smallest number of model differences. This indicates that the ability to leverage planning heuristics can make a marked difference in domains with a large number of explanatory actions.

Related Work

We end with a review of existing literature and emphasize the key differentiators for our framework.

Epistemic Planning It is well understood in social sciences that explanations must be generated while keeping in mind the beliefs of the agent receiving the explanation (Miller 2018). As such, epistemic planning makes for an excellent framework for studying the problem of generating these explanations. While the most general formulation of epistemic planning has been shown to be undecidable, many simpler fragments have been identified (Bolander *et al.* 2015). In human-aware planning settings too, there is wide consensus that epistemic planning could be an extremely useful tool. Readers can refer to (Miller 2017) for an overview of works done in employing epistemic planning for "social planning". Recently, there have been a lot of interest in developing efficient methods for planning in such settings (Muise *et al.* 2015; Kominis and Geffner 2015; 2017; Le *et al.* 2018; Huang *et al.* 2018). Cohen and Perrault (1979) have also investigated the use of speech acts in planning problems. Following the conventions of (Cohen and Perrault 1979), the explanatory actions studied within this paper can be viewed as INFORM acts.

Model Reconciliation Among the works related to model reconciliation, the work that is most closely connected to ours is (Chakraborti *et al.* 2019b). The idea of balanced plans were first proposed in that work. Unfortunately, the actual algorithm study there is incomplete and is not guaranteed to produce the least expensive balanced plan. Even the complete version they hypothesize in their paper relies enumerating all the possible optimal plans for a given updated model, which can be extremely inefficient, particularly since it is expected to be performed for every possible model in the model space. As we see in the empirical evaluations, our method (which is also complete) is often faster against the optimistic approximate version. Moreover the methods discussed in that paper are unable to utilize task-level actions with epistemic side effect or take into account task level constraints for purely communicative actions and the effects of execution on an observer, as we illustrate through examples.

Communicative Actions Our work also looks at the use of explanatory actions as a means of communicating information to the human observer. The most obvious types of such explanatory action includes purely communicative actions such as speech (Tellex *et al.* 2014) or the use of mixed reality projections (Chakraborti *et al.* 2018; Ganesan 2017). Recent works have shown that physical agents could also use movements to relay information such as intention (MacNally *et al.* 2018; Dragan *et al.* 2013) and incapability (Kwon *et al.* 2018). Our framework allows for a natural trade-off between these different types of communication.

Contrastive Explanations and Inferential Capabilities Many recent works dealing with explanation generation for planning have looked at characterizing explanations in terms of the types of questions they answer (Fox *et al.* 2017; Smith 2012). This characterization is orthogonal to the question of what type of information constitutes valid explanations. Putting aside questions regarding observability, the reason why a user may request an explanation is either due to knowledge mismatch (incomplete or incorrect knowledge of the task) or due to limitations of their inferential capabilities. The answer to any of these questions would require correcting the human's model of the task and/or providing inferential assistance. Works that have looked at model reconciliation explanations have mostly focused on the former. Explanations discussed in this paper can be viewed as an answer to the question *"Why this plan?"* (which can also be viewed as a contrastive question of the form *"Why this plan and not any other plan?"*). This is not to say that in complex scenarios just the model reconciliation information would suffice, but it would need to be supplemented with information internal to the model that can address the differences in inferential capabilities. Use of abstractions (Sreedharan *et al.* 2018), providing refutation of specific foils (Sreedharan *et al.* 2018) and providing causal explanations (Seegebarth *et al.* 2012) could be used to augment model reconciliation.

Explicable Planning Explicable planning looks at cases where the agent is incapable of updating the users' expectations and can choose to following the plan that best matches the user expectations and is valid in the robot model. Two representative works in this direction are (Zhang *et al.* 2017) and (Kulkarni *et al.* 2019). (Zhang *et al.* 2017) investigates scenarios where the human model may be unknown while (Kulkarni *et al.* 2019) proposes an iterative planning formalism that tries to find the most explicable plan by generating all possible valid solutions of a given cost threshold and then tries to find the most explicable plan from that set. Unlike the work presented here they look at more general distance measures for explicability, some of which are based on global plan properties. We can extend our current formulation to take into account such scores by turning these distances into the cost of an extra GOAL action (similar to the balancing formalism that allows for sub-optimality).

Conclusion

The paper presents a unifying formulation for the task of planning in the presence of users with incorrect mental models of the planning agent's capabilities. The formulation allows us to unify, for the first time, explanatory and explicable paradigms into a single framework plus is also compilable to classical planning. We discuss how this formulation can be extended to capture novel explanatory behaviors hitherto unexplored in literature while being computationally more efficient than methods that rely on direct model space search. One of the exciting features of our work is that we are able to place Expectation-Aware planning within the realm of epistemic planning, thereby laying the ground work to study more complex interaction scenarios including cases with more levels of nesting, uncertainty about mental models, more expressive models, incorporating non-deterministic effects, and so on. It would also be worth investigating specific considerations for choosing heuristics or formulating new ones for such problems.

Acknowledgments

We thank Dan Weld for helpful comments on a previous draft. This research is supported in part by ONR grants N00014-16-1-2892, N00014-18-1-2442, N00014-18-1-2840, N00014-9-1-2119, AFOSR grant FA9550-18-1-0067, DARPA SAIL-ON grant W911NF-19-2-0006, NSF grants 1936997 (C-ACCEL), 1844325, NASA grant

NNX17AD06G, and a JP Morgan AI Faculty Research grant..

References

Thomas Bolander, Martin Holm Jensen, and Francois Schwarzentruber. Complexity Results in Epistemic Planning. In *IJCAI*, 2015.

Tom Bylander. The Computational Complexity of Propositional STRIPS Planning. *Artificial Intelligence*, 1994.

Tathagata Chakraborti, Sarath Sreedharan, Yu Zhang, and Subbarao Kambhampati. Plan Explanations as Model Reconciliation: Moving Beyond Explanation as Soliloquy. In *IJCAI*, 2017.

Tathagata Chakraborti, Sarath Sreedharan, and Subbarao Kambhampati. Projection-Aware Task Planning and Execution for Human-in-the-Loop Operation of Robots in a Mixed-Reality Workspace . In *IROS*, 2018.

Tathagata Chakraborti, Sarath Sreedharan, Sachin Grover, and Subbarao Kambhampati. Plan Explanations as Model Reconciliation – An Empirical Study. In *HRI*, 2019.

Tathagata Chakraborti, Sarath Sreedharan, and Subbarao Kambhampati. Balancing Explicability and Explanation in Human-Aware Planning. *IJCAI*, 2019.

Philip R Cohen and C Raymond Perrault. Elements of a Plan-Based Theory of Speech Acts. *Cognitive science*, 1979.

Anca Dragan, Kenton Lee, and Siddhartha Srinivasa. Legibility and Predictability of Robot Motion. In *HRI*, 2013.

Maria Fox, Derek Long, and Daniele Magazzeni. Explainable Planning. In *IJCAI XAI Workshop*, 2017.

Ramsundar Kalpagam Ganesan. *Mediating Human-Robot Collaboration through Mixed Reality Cues*. PhD thesis, Arizona State University, 2017.

Hector Geffner and Blai Bonet. A concise introduction to models and methods for automated planning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2013.

Xiao Huang, Biqing Fang, Hai Wan, and Yongmei Liu. A General Multi-Agent Epistemic Planner Based on Higher-Order Belief Change. In *IJCAI*, 2018.

Filippos Kominis and Hector Geffner. Beliefs In Multiagent Planning: From One Agent to Many. In *ICAPS*, 2015.

Filippos Kominis and Hector Geffner. Multiagent Online Planning with Nested Beliefs and Dialogue. In *ICAPS*, 2017.

Anagha Kulkarni, Tathagata Chakraborti, Yantian Zha, Satya Gautam Vadlamudi, Yu Zhang, and Subbarao Kambhampati. Explicable Robot Planning as Minimizing Distance from Expected Behavior. In *AAMAS*, 2019.

Minae Kwon, Sandy Huang, and Anca Dragan. Expressing Robot Incapability. In *HRI*, 2018.

Tiep Le, Francesco Fabiano, Tran Cao Son, and Enrico Pontelli. EFP and PG-EFP: Epistemic Forward Search Planners in Multi-Agent Domains. In *ICAPS*, 2018.

Aleck M MacNally, Nir Lipovetzky, Miquel Ramirez, and Adrian R Pearce. Action Selection for Transparent Planning. In *AAMAS*, 2018.

Tim Miller. Social Planning – Reasoning with and about others, 2017. Invited Talk at IJCAI Workshop on Impedance Matching in Cognitive Partnerships.

Tim Miller. Explanation in Artificial Intelligence: Insights from the Social Sciences. *Artificial Intelligence*, 2018.

Christian J Muise, Vaishak Belle, Paolo Felli, Sheila A McIlraith, Tim Miller, Adrian R Pearce, and Liz Sonenberg. Planning Over Multi-Agent Epistemic States: A Classical Planning Approach. In *AAAI*, 2015.

Bastian Seegebarth, Felix Müller, Bernd Schattenberg, and Susanne Biundo. Making Hybrid Plans More Clear to Human Users – A Formal Approach for Generating Sound Explanations. In *ICAPS*, 2012.

David E Smith. Planning as an Iterative Process. In *AAAI*, 2012.

Sarath Sreedharan, Siddharth Srivastava, and Subbarao Kambhampati. Hierarchical Expertise-Level Modeling for User Specific Contrastive Explanations. In *IJCAI*, 2018.

Stefanie Tellex, Ross Knepper, Adrian Li, Daniela Rus, and Nicholas Roy. Asking for Help Using Inverse Semantics. In *RSS*, 2014.

Yu Zhang, Sarath Sreedharan, Anagha Kulkarni, Tathagata Chakraborti, Hankz Hankui Zhuo, and Subbarao Kambhampati. Plan Explicability and Predictability for Robot Task Planning. In *ICRA*, 2017.