Handling Model Uncertainty and Multiplicity in Explanations via Model Reconciliation

Sarath Sreedharan and **Tathagata Chakraborti** and **Subbarao Kambhampati**
Arizona State University, Tempe, AZ 85281 USA
{ ssreedh3, tchakra2, rao } @ asu.edu

Abstract

Model reconciliation has been proposed as a way for an agent to explain its decisions to a human who may have a different understanding of the same planning problem by explaining its decisions in terms of these model differences. However, often the human's mental model (and hence the difference) is not known precisely and such explanations cannot be readily computed. In this paper, we show how the explanation generation process evolves in the presence of such model uncertainty or incompleteness by generating *conformant explanations* that are applicable to a *set of possible models*. We also show how such explanations can contain superfluous information and how such redundancies can be reduced using *conditional explanations* to iterate with the human to attain common ground. Finally, we will introduce an anytime version of this approach and empirically demonstrate the trade-offs involved in the different forms of explanations in terms of the computational overhead for the agent and the communication overhead for the human. We illustrate these concepts in three well-known planning domains as well as in a demonstration on a robot involved in a typical search and reconnaissance scenario with an external human supervisor.

In (Chakraborti et al. 2017) we looked at how a robot can explain its decisions to a human in the loop who might have a different understanding of the same problem (either in terms of the agent's knowledge or intentions, or in terms of its capabilities). These explanations are intended to bring the human's mental model closer to the robot's estimation of the ground truth – this is referred to as the *model reconciliation process*, by the end of which a plan that is optimal in the robot's model is also estimated to be optimal in the human's updated mental model. It was also shown how this process can be achieved successfully while transferring the minimum number of model updates possible via what are called *minimally complete explanations* or MCEs.

Explanations of this form have been inspired by works like (Lombrozo 2006; 2012; Miller 2017) which identify properties of explanations in terms of *selectivity*, *contrastiveness* and *mental modeling* of the explainee, and our recent work (Chakraborti et al. 2018) also demonstrated the usefulness of such explanations. Such techniques can thus be essential contributors to the dynamics of trust and teamwork in human-agent collaborations by significantly lowering the communication overhead between agents while at the same time providing the right amount of information to keep the agents on the same page with respect to their understanding of each others' beliefs, intentions and capabilities – thereby reducing the cognitive burden on the human teammates and increasing their situational awareness.

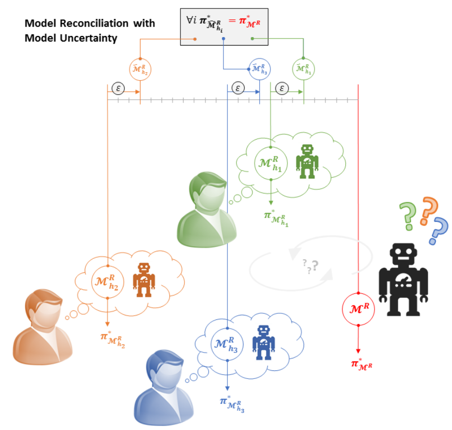

Figure 1: An illustration of the model reconciliation process in case of model uncertainty or multiple explainees.

This process of model reconciliation is illustrated in Figure 1. The robot's model, which is its estimate of the ground truth, is represented by $\mathcal{M}^R$ (note: "model" of a planning problem includes the state and goals information as well as the domain or action model) and $\pi^*_{\mathcal{M}^R}$ is the optimal plan in it. A human H who is interacting with it may have a different model $\mathcal{M}^R_h$ of the same planning problem, and the optimal plan $\pi^*_{\mathcal{M}^R_h}$ in the human's model can diverge from that of the robot's leading to the robot needing to explain its decision to the human. As explained above, an explanation is an update or correction to the human's mental model to a new intermediate model $\widehat{\mathcal{M}}^R_h$ where (according to cost or some other suitable measure of similarity) the optimal plan $\pi^*_{\widehat{\mathcal{M}}^R_h}$ is *equivalent* to the original plan $\pi^*_{\mathcal{M}^R}$.

However, this process is only feasible if inconsistencies of the robot's model with the human's mental model are known precisely. Although we made this assumption before (Chakraborti et al. 2017) as a first step towards formalizing the model reconciliation process, this can be hard to achieve in practice. Instead, the agent may end up having to explain its decisions with respect to a *set of possible models* which is its best estimation of the human's knowledge state learned in the process of interactions. Sets of possible models can be concisely represented as planning models with *annotations* for possible preconditions and effects (Nguyen, Sreedharan, and Kambhampati 2017; Bryce, Benton, and Boldt 2016). In this situation, the robot can compute MCEs for each possible configuration. However, this can result in situations where the explanations computed for individual models independently are not consistent across all the possible target domains. Thus, in the case of model uncertainty, such an approach cannot guarantee that the resulting explanation will be acceptable.

Instead, we want to find an explanation such that $\forall i \; \pi^*_{\widehat{\mathcal{M}}^R_{h_i}} \equiv \pi^*_{\mathcal{M}^R}$. This is a single model update that makes the given plan optimal (and hence explained) in all the updated domains (or in all possible domains). At first glance, it appears that such an approach, even though desirable, might turn out to be prohibitively expensive especially since solving for a *single* MCE involves search in the model space where each search node is an optimal planning problem. However, it turns out that the same search strategy can be employed here as well by modifying the way in which the models are represented and the equivalence criterion is computed during the search process. Thus, in this paper, we –

(1) show how uncertainty over the human mental model can be represented in the form of *annotated* models;

(2) outline how the concept of an MCE becomes one of *conformant explanations* in the revised setting and the search for these can be compiled to the original MCE search;

(3) show how superfluous information in conformant explanations can be reduced interactively via *conditional explanations* which can be computed in an anytime manner;

(4) demonstrate how the model reconciliation process in the presence of *multiple humans* in the loop can be viewed as a special case of uncertain models

Background

In this section, we provide a brief introduction to classical planning and incompleteness of planning models.

A Classical Planning Problem is a tuple $\mathcal{M} = \langle \mathcal{D}, \mathcal{I}, \mathcal{G} \rangle$ with domain $\mathcal{D} = \langle F, A \rangle$ – where F is a finite set of fluents that define a state $s \subseteq F$, and A is a finite set of actions – and initial and goal states $\mathcal{I}, \mathcal{G} \subseteq F$. Action $a \in A$ is a tuple $\langle c_a, \mathit{pre}(a), \mathit{eff}^{\pm}(a) \rangle$ where c_a is the cost, and $\mathit{pre}(a), \mathit{eff}^{\pm}(a) \subseteq F$ are the preconditions and add/delete effects, i.e. $\delta_{\mathcal{M}}(s, a) \models \bot$ *if* $s \not\models \mathit{pre}(a)$; *else* $\delta_{\mathcal{M}}(s, a) \models s \cup \mathit{eff}^{+}(a) \setminus \mathit{eff}^{-}(a)$ where $\delta_{\mathcal{M}}(\cdot)$ is the transition function. The cumulative transition function is given by $\delta_{\mathcal{M}}(s, \langle a_1, a_2, \ldots, a_n \rangle) = \delta_{\mathcal{M}}(\delta_{\mathcal{M}}(s, a_1), \langle a_2, \ldots, a_n \rangle)$.

This forms the classical definition of a planning problem (Russell and Norvig 2003) whose models are represented in the syntax of `PDDL` (McDermott et al. 1998). The solution to the planning problem is a sequence of actions or a (satisficing) *plan* $\pi = \langle a_1, a_2, \ldots, a_n \rangle$ such that $\delta_{\mathcal{M}}(\mathcal{I}, \pi) \models \mathcal{G}$. The cost of a plan π is given by $C(\pi, \mathcal{M}) = \sum_{a \in \pi} c_a$ if $\delta_{\mathcal{M}}(\mathcal{I}, \pi) \models \mathcal{G}$; ∞ otherwise. The cheapest plan $\pi^* = \arg\min_{\pi} C(\pi, \mathcal{M})$ is the (cost) optimal plan. We refer to the cost of the optimal plan in the model $\mathcal{M}$ as $C^*_{\mathcal{M}}$.

In previous work (Nguyen, Sreedharan, and Kambhampati 2017) we introduced an updated representation of planning problems in the form of *annotated* models or `APDDL` to account for uncertainty or incompleteness over the definition of a classical planning model. In addition to the standard preconditions and effects associated with actions, it introduces the notion of *possible* preconditions and effects which may or may not be realized in practice.

An Incomplete (Annotated) Model is the tuple $\mathbb{M} = \langle \mathbb{D}, \mathbb{I}, \mathbb{G} \rangle$ with a domain $\mathbb{D} = \langle F, \mathbb{A} \rangle$ – where F is a finite set of fluents that define a state $s \subseteq F$, and $\mathbb{A}$ is a finite set of annotated actions – and annotated initial and goal states $\mathbb{I} = \langle \mathcal{I}^0, \mathcal{I}^+ \rangle$, $\mathbb{G} = \langle \mathcal{G}^0, \mathcal{G}^+ \rangle$; $\mathcal{I}^0, \mathcal{G}^0, \mathcal{I}^+, \mathcal{G}^+ \subseteq F$. Action $a \in \mathbb{A}$ is a tuple $\langle c_a, \mathit{pre}(a), \widetilde{\mathit{pre}}(a), \mathit{eff}^{\pm}(a), \widetilde{\mathit{eff}}^{\pm}(a) \rangle$ where c_a is the cost and, in addition to its *known* preconditions and add/delete effects $\mathit{pre}(a), \mathit{eff}^{\pm}(a), \subseteq F$ each action also contains *possible preconditions* $\widetilde{\mathit{pre}}(a) \subseteq F$ containing propositions that action a *might* need as preconditions, and *possible add (delete) effects* $\widetilde{\mathit{eff}}^{\pm}(a) \subseteq F$) containing propositions that the action a *might* add (delete, respectively) after execution. Similarly, $\mathcal{I}^0, \mathcal{G}^0$ (and $\mathcal{I}^+, \mathcal{G}^+$) are the known (and possible) parts of the initial and goal states.

Each possible condition $f \in \widetilde{\mathit{pre}}(a) \cup \widetilde{\mathit{eff}}^{\pm}(a)$ also has a probability $p(f)$ associated with it denoting how likely it is to appear as a known condition in the ground truth model – i.e. $p(f)$ measures the confidence with which that condition has been learned. The sets of known and possible conditions of a model $\mathcal{M}$ is called $\mathbb{S}_k(\mathcal{M})$ and $\mathbb{S}_p(\mathcal{M})$ respectively.

An *instantiation* of an annotated model $\mathbb{M}$ is a classical planning model where a subset of the possible conditions have been realized, and is thus given by the tuple $inst(\mathbb{M}) = \langle \mathcal{D}, \mathcal{I}, \mathcal{G} \rangle$ with domain $\mathcal{D} = \langle F, A \rangle$, initial and goal states $\mathcal{I} = \mathcal{I}^0 \cup \chi;\ \chi \subseteq \mathcal{I}^+$ and $\mathcal{G} = \mathcal{G}^0 \cup \chi;\ \chi \subseteq \mathcal{G}^+$ respectively, and action $A \ni a = \langle c_a, \mathit{pre}(a) \leftarrow \mathit{pre}(a) \cup \chi;\ \chi \subseteq \widetilde{\mathit{pre}}(a), \mathit{eff}^{\pm}(a) \leftarrow \mathit{eff}^{\pm}(a) \cup \chi;\ \chi \subseteq \widetilde{\mathit{eff}}^{\pm}(a) \rangle$. Given an annotated model with k possible conditions, there may be 2^k such instantiations, which forms its *completion set* (Nguyen, Sreedharan, and Kambhampati 2017).

The Likelihood $\mathcal{L}$ of an instantiation $inst(\mathbb{M})$ of the annotated model $\mathbb{M}$ is given by –

$$\mathcal{L}(inst(\mathbb{M})) = \prod_{f \in \mathbb{S}_p(\mathbb{M}) \wedge \mathbb{S}_k(inst(\mathbb{M}))} p(f) \times \prod_{f \in \mathbb{S}_p(\mathbb{M}) \setminus \mathbb{S}_k(inst(\mathbb{M}))} (1 - p(f))$$

Such models turn out to be especially useful for the representation and learning of human (mental) models from observations, where uncertainty after the learning process can be represented in terms of model annotations as in (Nguyen, Sreedharan, and Kambhampati 2017; Bryce, Benton, and Boldt 2016). Let $\mathbb{M}^R_H$ be the culmination of a model learning process and $\{\mathcal{M}_{h_i}^R\}_i$ be the completion set of $\mathbb{M}_H^R$. Note that one of these models is the actual ground truth (i.e. the human's real mental model). We refer to this as $g(\mathbb{M}_H^R)$.

The representation itself is general enough to handle all model differences including initial and goal states in addition to precondition/effects. Cases with unknown actions, as long as their existence is known (but possibly uncertain), can just appear with empty conditions if the action is in the robot's model but not in the human's (or with a special indicator condition if the action is in the human's model but not in the robot's) and are thus subsumed by the current representation. Thus the representation does not preclude situations where the robot is completely unaware of the human mental model (as long as the robot is aware of the list of action names that the human may expect). An approach to capturing this would be to consider all possible predicates as preconditions and effects as in (Bryce, Benton, and Boldt 2016) where authors used this to model the mental model of expert users. A more efficient method to handle empty human models would be to learn or refine an annotated model from training data collected from the human teammate as in (Nguyen, Sreedharan, and Kambhampati 2017). The current work specifically focuses on the explanation generation problem once such a model has already been learned.

The Human-Aware Planning Setting

The human-aware planning paradigm (Chakraborti, Sreedharan, and Kambhampati 2017) introduces the *mental model* of the human in the loop into a planner's deliberative process, in addition to the planner's own model in the classical sense. In such settings, when a planner's optimal plans diverge from human expectations due to differences in these models, the planner can attempt corrections to the human's mental model to resolve the *perceived* inoptimality by participating in what we call the *model reconciliation* process.

A Human-Aware Planning (HAP) Setting is the tuple $\Phi = \langle \mathcal{M}^R, \mathcal{M}_H^R \rangle$, where $\mathcal{M}^R = \langle D^R, \mathcal{I}^R, \mathcal{G}^R \rangle$ is the planner's model of a planning problem, while $\mathcal{M}_H^R = \langle D_H^R, \mathcal{I}_H^R, \mathcal{G}_H^R \rangle$ is the robot's (annotated) estimate of the human's knowledge of the same[1].

The Model Reconciliation Problem (MRP) is the tuple $\Psi = \langle \pi, \Phi \rangle$, given an MMP Φ, where $C(\pi, \mathcal{M}^R) = C^*_{\mathcal{M}^R}$, i.e. π is the optimal plan in $\mathcal{M}^R$.

A solution to an MRP (Chakraborti et al. 2017) is the set of model changes $\mathcal{E}$ or an *explanation*, such that the given plan π is optimal in both the robot model $\mathcal{M}^R$ and the updated human model $\widehat{\mathcal{M}}_H^R$. Thus –

(1) $\widehat{\mathcal{M}}_H^R \longleftarrow \mathcal{M}_H^R + \mathcal{E}$; and

(2) $C(\pi, g(\mathbb{M}_H^R)) = C^*_{g(\mathbb{M}_H^R)}$.

[1]Note that the robot model need not be the ground truth. However, the robot can only explain with respect to what it believes to be true. This can, of course, be wrong and be refined iteratively through interaction with the human (Sengupta et al. 2017).

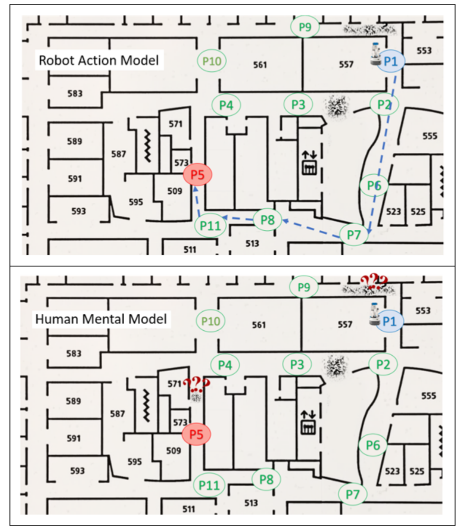

Figure 2: USAR scenario with an internal robot and an external human. The robot plan is marked in blue, uncertain parts of the human model is marked with red question marks.

A Minimally Complete Explanation (MCE) is the shortest explanation that satisfies conditions (1) and (2).

Condition (2) is hard to achieve since it is not known with certainty which is the actual mental model. So we want to preserve (2) for all (or as many) instantiations of the incomplete estimation of the explainee's mental model. In the following discussion, we are going to show how this can be achieved by modified versions of the model-space MCE-search in (Chakraborti et al. 2017) using annotated models.

Use Case: The USAR Domain

We will now introduce a Urban Search And Reconnaissance (USAR) domain (Murphy 2000) which we will use as an illustrative purposes throughout the rest of the paper. A video demonstrating the different scenarios play out is provided at `https://youtu.be/bLqrtffW6Ng`. Here a robot is involved in a typical disaster response operation, controlled partly or fully by an external human commander (Bartlett 2015). The robot's job is to infiltrate areas that may be otherwise inaccessible to humans, and report on its surroundings as and when instructed by the external, or required by its team. The external has a map of the environment, but this map may no longer be accurate in a disaster scenario - e.g. new paths may have opened up, or older paths may no longer be available, due to rubble from collapsed structures like walls and doors. The robot (internal), however, does not need to inform the external of all these changes so as not to cause information overload of the commander who is usually otherwise engaged in orchestrating the entire operation, and it must do this keeping in mind its estimate of the latter's mental model which may be uncertain.

In this particular scenario, we have a robot located at P1 (blue), that needs to collect data from point P5. While the human commander understands the goal, she is confused about the current status of the scenario. The commander is under the false impression that the paths from P1 to P9 and P4 to P5 are unusable (red question marks). The human is also unaware of the robot's inability to use its hands.

While the robot does not have a complete picture of the human's mental model, it understands that any differences between the models would be related to (1) Path from P1 to P9; (2) Path from P4 to P5; (3) Robot's ability to use its hands; and (4) Whether the Robot needs its arm to clear rubble. As far as the robot is concerned, the human model can be one of sixteen possible models (one of which is the human's actual mental model). The robot can now possibly adopt one of the two approaches; namely, it can try to reduce the uncertainty over the human mental models or try to come up with an explanation that would work in any one of these possible models. We call the latter *Conformant Explanations* which can explain the plan to the human irrespective of their actual mental model (provided it lies in the set of possible models). In the above scenario, a conformant explanation for the optimal robot plan (blue) is as follows –

```
remove-known-INIT-has-add-effect-hand_capable
add-annot-clear_passage-has-precondition-hand_capable
remove-annot-INIT-has-add-effect-clear_path P1 P9
```

Notice that the second explanation (regarding the need of the hand to clear rubble) was already known to the human and was thus superfluous information. We will now formally define conformant explanations and introduce an algorithm to generate the same. We will also look at methods to reduce possibly superfluous information in such explanations.

Conformant Explanations

Given the above discussion, we define *robustness* of an explanation for an incomplete mental models as the probability mass of models where it is a valid explanation. Formally,

Robustness of an explanation $\mathcal{E}$ for an MRP $\Psi = \langle \pi, \langle \mathcal{M}^R, \mathbb{M}_H^R \rangle \rangle$ is given by –

$$R(\mathcal{E}) = \sum_{inst(\widehat{\mathcal{M}}_H^R) \text{ s.t. } C(\pi, inst(\widehat{\mathcal{M}}_H^R)) = C^*_{inst(\widehat{\mathcal{M}}_H^R)}} \mathcal{L}(inst(\widehat{\mathcal{M}}_H^R))$$

A Conformant Explanation is such that $R(\mathcal{E}) = 1$.

This means a conformant explanation ensures that the given plan is explained in all the models in the completion set of the human model. The above example in the USAR domain is in fact such an explanation.

MRP with Model Uncertainty – $\mathcal{M}_{max}$ & $\mathcal{M}_{min}$

We begin by defining two models – the most relaxed model possible $\mathcal{M}_{max}$ and the least relaxed one $\mathcal{M}_{min}$. The former is the model where all the possible add effects and none of the possible preconditions and deletes hold, the state has all the possible conditions set to true, and the goal is the smallest one possible; while in the latter all the possible preconditions and deletes and none of the possible adds are realized and with the minimal start state and the maximal goal. This means that, if a plan is executable in $\mathcal{M}_{min}$ it will be executable in all the possible models. Also, if this plan is optimal in $\mathcal{M}_{max}$, then it must be optimal throughout the set. Of course, such a plan may not exist, but we are not trying to find one either. Instead, we are trying to find a set of model updates which when applied to the annotated model, produce a new set of models where a *given* plan is optimal. In providing these model updates, we are in effect reducing the set of possible models to a smaller set. The new set need not be a subset of the original set of models but will be equal or smaller in size to the original set. For any given annotated model, such an explanation always exists (entire model difference in the worst case), and we intend to find the smallest one. $\mathbb{M}_H^R$ thus affords the following two models –

$\mathcal{M}_{max} = \langle \mathcal{D}, \mathcal{I}, \mathcal{G} \rangle$ with domain $\mathcal{D} = \langle F, A \rangle$ and

- initial state $\mathcal{I} \leftarrow \mathcal{I}^0 \cup \mathcal{I}^+$; given $\mathbb{I}$
- goal state $\mathcal{G} \leftarrow \mathcal{G}^0$; given $\mathbb{G}$
- $\forall a \in A$
  - $pre(a) \leftarrow pre(a)$; $a \in \mathbb{A}$
  - $eff^+(a) \leftarrow eff^+(a) \cup \widetilde{eff}^+(a)$; $a \in \mathbb{A}$
  - $eff^-(a) \leftarrow eff^-(a)$; $a \in \mathbb{A}$

$\mathcal{M}_{min} = \langle \mathcal{D}, \mathcal{I}, \mathcal{G} \rangle$ with domain $\mathcal{D} = \langle F, A \rangle$ and

- initial state $\mathcal{I} \leftarrow \mathcal{I}^0$; given $\mathbb{I}$
- goal state $\mathcal{G} \leftarrow \mathcal{G}^0 \cup \mathcal{G}^+$; given $\mathbb{G}$
- $\forall a \in A$
  - $pre(a) \leftarrow pre(a) \cup \widetilde{pre}(a)$; $a \in \mathbb{A}$
  - $eff^+(a) \leftarrow eff^+(a)$; $a \in \mathbb{A}$
  - $eff^-(a) \leftarrow eff^-(a) \cup \widetilde{eff}^-(a)$; $a \in \mathbb{A}$

As explained before, $\mathcal{M}_{max}$ is a model where all the add effects hold and it is easiest to achieve the goal, and similarly $\mathcal{M}_{min}$ is the model where it is the hardest to achieve the goal. Note that these definitions might end up creating inconsistencies in the models (e.g. in an annotated model for the `BlocksWorld` domain, the definition of `unstack` action may have add effects to make the block both `holding` and `ontable` at the same time), but the model reconciliation process will take care of these.

Proposition 1 For a given MRP $\Psi = \langle \pi, \langle \mathcal{M}^R, \mathbb{M}_H^R \rangle \rangle$, if the plan π is optimal in $\mathcal{M}_{max}$ and executable in $\mathcal{M}_{min}$, then conditions (1) and (2) hold for all i.

This now becomes the new criterion to satisfy in the course of search for an MCE for a set of models.

MEGA*-Conformant

Similar to (Chakraborti et al. 2017) we define a state representation over planning problems with a mapping function $\Gamma : {}^a\mathcal{M} \mapsto \mathcal{F}$ which represents any planning problem in the new state space by transforming every condition (including the possible conditions) in the model of a planning problem into a predicate. The set Λ of actions contains unit model change actions $\lambda : \mathcal{F} \to \mathcal{F}$ which make a single change to a domain at a time, as defined in (Chakraborti et al. 2017).

We start the search (Algorithm 1) by first creating the corresponding $\mathcal{M}_{max}$ and $\mathcal{M}_{min}$ model for the given annotated model $\mathbb{M}_H^R$. While the goal test for the original MCE only included an optimality test, here we need to both check the optimality of the plan in $\mathcal{M}_{max}$ and verify the correctness of the plan in $\mathcal{M}_{min}$. As stated in Proposition 1, the

Algorithm 1 `MEGA*-Conformant`

procedure MCE-SEARCH
Input: MRP $\langle \pi^*, \langle \mathcal{M}^R, \mathbb{M}_h^R \rangle \rangle$
Output: Explanation $\mathcal{E}^{MCE}$
Procedure:
  fringe $\leftarrow$ `Priority_Queue()`
  c_list $\leftarrow \{\}$ ▷ Closed list
  $\pi^*_R \leftarrow \pi^*$ ▷ Optimal plan being explained
  $\mathcal{M}_{max}$, $\mathcal{M}_{min} \leftarrow (\mathbb{M}_h^R)$ ▷ Proposition 2
  fringe.push($\langle \mathcal{M}_{min}, \mathcal{M}_{max}, \{\} \rangle$, priority $= 0$)
  while True **do**
    $\langle \widehat{\mathcal{M}}_{min}, \widehat{\mathcal{M}}_{max}, \mathcal{E} \rangle, c \leftarrow$ fringe.pop()
    if $C(\pi^*_R, \widehat{\mathcal{M}}_{max}) = C^*_{\widehat{\mathcal{M}}_{max}} \wedge \delta(\mathcal{I}_{\widehat{\mathcal{M}}_{min}}, \pi^*_R) \models \mathcal{G}_{\widehat{\mathcal{M}}_{min}}$ **then**
      return $\mathcal{E}$ ▷ Proposition 1
    else
      c_list $\leftarrow$ c_list $\cup \langle \widehat{\mathcal{M}}_{max}, \widehat{\mathcal{M}}_{min} \rangle$
      for $f \in \{\Gamma(\widehat{\mathcal{M}}_{min}) \cup \Gamma(\widehat{\mathcal{M}}_{max})\} \setminus \Gamma(\mathcal{M}^R)$ **do**
        $\lambda \leftarrow \langle 1, \langle \widehat{\mathcal{M}}_{min}, \widehat{\mathcal{M}}_{max} \rangle, \{\}, \{f\} \rangle$ ▷ Removes f from $\widehat{\mathcal{M}}$
        if $\delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda) \notin$ c_list **then**
          fringe.push($\langle \delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda)$, $\mathcal{E} \cup \lambda \rangle$, $c + 1$)
      for $f \in \Gamma(\mathcal{M}^R) \setminus \{\Gamma(\widehat{\mathcal{M}}_{min}) \cup \Gamma(\widehat{\mathcal{M}}_{max})\}$ **do**
        $\lambda \leftarrow \langle 1, \{\langle \widehat{\mathcal{M}}_{min}, \widehat{\mathcal{M}}_{max} \rangle, \{f\}, \{\} \rangle$ ▷ Adds f to $\widehat{\mathcal{M}}$
        if $\delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda) \notin$ c_list **then**
          fringe.push($\langle \delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda)$, $\mathcal{E} \cup \lambda \rangle$, $c + C_\lambda$)

plan is only optimal in the entire set of possible models if it satisfies both tests. Since the correctness of a given plan can be verified in polynomial time with respect to the plan size, this is a relatively easy test to perform.

The other important point of difference between the algorithm mentioned above and the original MCE is how we calculate the applicable model updates. Here we consider the superset of model differences between the robot model and $\mathcal{M}_{min}$ and the differences between the robot model and $\mathcal{M}_{max}$. This could potentially mean that the search might end up applying a model update that is already satisfied in one of the models but not in the other. Since all the model update actions are formulated as set operations, the original MRP formulation can handle this without any further changes. The models obtained by applying the model update to $\mathcal{M}_{min}$ and $\mathcal{M}_{max}$ are then pushed to the open queue.

Proposition 2 $\mathcal{M}_{max}$ and $\mathcal{M}_{min}$ only need to be computed once before the search – i.e. with a model update $\mathcal{E}$ to $\mathbb{M}$: $\mathcal{M}_{max} \leftarrow \mathcal{M}_{max} + \mathcal{E}$ and $\mathcal{M}_{min} \leftarrow \mathcal{M}_{min} + \mathcal{E}$.

Following Proposition 2, these models form the new $\mathcal{M}_{min}$ and $\mathcal{M}_{max}$ models for the set of models obtained by applying the current set of model updates to the original annotated model. This proposition ensures that we no longer have to keep track of the current list of models or recalculate $\mathcal{M}_{min}$ and $\mathcal{M}_{max}$ for the new set.

We saw earlier that conformant explanations can contain superfluous information – i.e. asking the human to remove non-existent conditions or add existing ones. Such redundant information can be annoying and may end up reducing the human's trust in the robot. This can be avoided by –

- Increasing the cost of model updates involving uncertain conditions relative to those involving known preconditions or effects. This ensures that the search prefers explanations that contain known conditions. By definition, such explanations will not have superfluous information.
- However, sometimes such explanations may not exist. Instead, we can convert conformant explanations into *conditional* ones. This can be achieved by turning each model update for an annotated condition into a question and only provide an explanation if the human's response warrants it – e.g. instead of asking the human to update the precondition of `clear_passage`, the robot can first ask if the human thinks that action has a precondition `hand_usable`. This is the topic of the next section.

Conditional Explanations

One way of removing superfluous explanations is to engage the human in conversation and ask questions that can reduce the size of the completion set. To this end, we define –

A Conditional Explanation is represented by a policy that maps the annotated model (represented by a $\mathcal{M}_{min}$ and $\mathcal{M}_{max}$ model pair) to either a question regarding the existence of a condition in the human ground model or a model update request. The resultant annotated model is produced, by either applying the model update directly into the current model or by updating the model to conform to human's answer regarding the existence of the condition.

We can generate these conditional explanations by either performing post-processing on conformant explanations or by performing an AND-OR graph search with AO^*(Nilsson 1980). Here each model update related to a known condition forms an OR successor node while each *possible* condition can be applied on the current state to produce a pair of AND successors, where the first node reflects a node where the annotated condition holds while the second one represents the state where it does not. So the number of possible conditions reduces by one in each one of these AND successor nodes. This AND successor relates to the answers the human could potentially provide when asked about the existence of that particular possible condition. Note that this AND-OR graph will not contain any cycles as we only provide model updates that are consistent with the robot model and hence we can directly use the AO^* search here.

`MEGA*-Conditional`

The possibility of asking humans for clarification on uncertain predicates opens the door to generating potentially cheaper explanations. Consider the following exchange –

```
R : Are you aware that the path from P1 to P4 has collapsed?
H : Yes.
> R realizes the plan is optimal in all possible human models.
> It does not need to explain further.
```

Unfortunately, if we used the standard AO^* search, it will not produce a conditional explanation that contains this "less robust" explanation as one of the potential branches in the conditional explanation. This is because, if the human had said that the path was free, the robot would need to revert to the original conformant explanation. Thus the cost of the subtree containing this solution will be higher than the one that only includes the original conformant explanation.

To overcome this shortcoming, we introduce a discounted version of the AO^* search. Where the cost contributed by a pair of AND successors is calculated as –

$$min(\textit{node1.h\_val, node2.h\_val}) + \gamma * max(\textit{node1.h\_val, node2.h\_val})$$

where node1 and node2 are the successor nodes and node1.h_val, node2.h_val are their respective h-values. Here γ represents the discount fact and controls how much the search values short paths in its solution subtree. When $\gamma = 1$, the search becomes standard AO^* search and when $\gamma = 0$, the search myopically optimizes for short branches (at the cost of the depth of the solution subtree). The rest of the algorithm stays the same as the standard AO^* search. The pseudocode is provided in Algorithm 3.

Remark. Note that in asking questions such as these, the robot is trying to exploit the human's (lack of) knowledge of the problem in order to provide more concise explanations. This can be construed as a case of lying by omission and can raise interesting ethical considerations (Chakraborti and Kambhampati 2018) Humans, during an explanation process, tend to undergo this same "selection" process (Miller 2017) as well in determining which of the many reasons that could explain an event is worth highlighting. It is worthwhile investigating similar behavior for autonomous agents.

`MEGA`*`-Anytime`

Both the algorithms discussed above can be computationally expensive. However, we can relax the minimality requirement of explanation for shorter explanation generation time. For this we introduce an anytime depth first explanation generation algorithm. Here, for each state, the successor states include all the nodes that can be generated by applying the model edit actions on all the known predicates and two possible successors for each possible condition – one where the condition holds and one where it does not. Once the search reaches a goal state (a new model where the target plan is optimal throughout its completion set), it queries the human to see if the assumptions it has made regarding possible conditions hold in the human mental model (the list of model updates made related to possible conditions). If all the assumptions hold in the human model, then we return the current solution as the final explanation (or use the answers to look for smaller explanations), else continue the search after pruning the search space using the answers provided by the human. The pruning can be performed efficiently by keeping track of all the human answers and enforcing these specifications only at the time of expansion of new nodes. Algorithm 2 presents a depth-first search approach for an anytime solution. Here we add an additional variable $\mathcal{A}$ to the search node to keep track of the possible assumptions that we have made for any given search path. The `TEST_ASSUMPTION` denotes the function responsible for testing the set of assumptions during the goal test. `TEST_ASSUMPTION` returns the set of assumptions that were invalidated by the human $\mathcal{A}_{invalid}$ and we can return the current search path as a solution if the invalid set is empty. We will use the validated and invalidated assumption to update our current search stack (via the `UPDATE_STACK` function).

Algorithm 2 `MEGA`*`-Anytime`

procedure ANYTIME-EXPLANATION
Input: MRP $\langle \pi^*, \langle \mathcal{M}^R, \mathbb{M}_h^R \rangle \rangle$
Output: Explanation $\mathcal{E}$
Procedure:
  fringe $\leftarrow$ `Stack()`
  $\pi_R^* \leftarrow \pi^*$ ▷ Optimal plan being explained
  $\mathcal{M}_{max}, \mathcal{M}_{min} \leftarrow (\mathbb{M}_h^R)$ ▷ Proposition 2
  $\mathcal{A} \leftarrow \{\}$ ▷ Current assumptions
  fringe.push($\langle \mathcal{M}_{min}, \mathcal{M}_{max}, \mathcal{A}, \{\} \rangle$)
  while True **do**
    $\langle \widehat{\mathcal{M}}_{min}, \widehat{\mathcal{M}}_{max}, \mathcal{A}, \mathcal{E} \rangle \leftarrow$ fringe.pop()
    if $C(\pi_R^*, \widehat{\mathcal{M}}_{max}) = C^*_{\widehat{\mathcal{M}}_{max}} \wedge \delta(\mathcal{I}_{\widehat{\mathcal{M}}_{min}}, \pi_R^*) \models \mathcal{G}_{\widehat{\mathcal{M}}_{min}}$ **then**
      $\mathcal{A}_{valid}, \mathcal{A}_{invalid} \leftarrow$ TEST_ASSUMPTION($\mathcal{A}$)
      $\mathcal{A}_{valid} \leftarrow \mathcal{A} \setminus \mathcal{A}_{invalid}$
      if $|\mathcal{A}_{invalid}| = 0$ **then**
        return $\mathcal{E}$ ▷ Proposition 1
      else
        UPDATE_STACK(fringe, $\mathcal{A}_{valid}, \mathcal{A}_{invalid}$)
    else
      c_list $\leftarrow$ c_list $\cup \langle \widehat{\mathcal{M}}_{max}, \widehat{\mathcal{M}}_{min} \rangle$
      for $f \in \{\Gamma(\widehat{\mathcal{M}}_{min}) \cup \Gamma(\widehat{\mathcal{M}}_{max})\} \setminus \Gamma(\mathcal{M}^R)$ **do**
        $\lambda \leftarrow \langle 1, \langle \widehat{\mathcal{M}}_{min}, \widehat{\mathcal{M}}_{max} \rangle, \{\}, \{f\} \rangle$ ▷ Removes f from $\widehat{\mathcal{M}}$
        if $\delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda) \notin$ c_list **then**
          if $f \notin \{\Gamma(\widehat{\mathcal{M}}_{min}) \cap \Gamma(\widehat{\mathcal{M}}_{max})\} \setminus \Gamma(\mathcal{M}^R)$ **then**
            $\mathcal{A} \leftarrow \mathcal{A} \cup f$ ▷ Add to assumptions if possible condition
          fringe.push($\langle \delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda), \mathcal{E} \cup \lambda \rangle, \mathcal{A}$)
      for $f \in \Gamma(\mathcal{M}^R) \setminus \{\Gamma(\widehat{\mathcal{M}}_{min}) \cup \Gamma(\widehat{\mathcal{M}}_{max})\}$ **do**
        $\lambda \leftarrow \langle 1, \{\langle \widehat{\mathcal{M}}_{min}, \widehat{\mathcal{M}}_{max} \rangle, \{f\}, \{\} \rangle$ ▷ Adds f to $\widehat{\mathcal{M}}$
        if $\delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda) \notin$ c_list **then**
          fringe.push($\langle \delta_{\mathcal{M}^H, \mathcal{M}^R}(\langle \Gamma(\widehat{\mathcal{M}}_{min}), \Gamma(\widehat{\mathcal{M}}_{max}) \rangle, \lambda), \mathcal{E} \cup \lambda \rangle, \mathcal{A}$)

Remark. The purpose of the paper is to *demonstrate* how existing notions of conditional and conformant solutions in planning can be adopted for the explanation process equally well in the presence of uncertainty over the human mental model. While there are significant differences between how conditional or conformant explanations work with respect to their planning counterparts, it may be worth exploring the state-of-the-art (Albore, Palacios, and Geffner 2009; Bonet and Geffner 2005) in those fields to further develop on the concepts introduced in the paper.

Evaluation

We have already seen a demonstration of the algorithms in action in the USAR use case. In this section, we will evaluate the algorithms on three well-known IPC (International Planning Competition 2011) domains. For each domain, we chose five problems (generated through standard problem generators), and for each domain and problem pair, we create a new domain and problem by removing five random predicates. This new domain and problem represent the ground truth human model. Next, we generate the uncertain

Table 1: Runtime and solution size for the algorithms introduced in the paper.

Domain	Problem	Conformant explanations			Conditional Explanations			Anytime Explanations		
		Question Size	Explanation Size	Time (secs)	Question Size	Explanation Size	Time (secs)	Question Size	Explanation Size	Time (secs)
Blocksworld	p1	–	3 (6)	134.84	3	5	140.75	3	3	19.97
	p2	–	1 (1)	1.64	0	1	9.19	0	2	2.37
	p3	–	2 (3)	20.56	1	3	55.90	3	2	17.74
	p4	–	1 (2)	11.23	1	2	128.50	3	3	21.24
	p5	–	3 (6)	130.63	3	5	150.60	3	3	24.66
Logistics	p1	–	2 (4)	62.30	2	4	99.78	4	2	26.29
	p2	–	2 (5)	61.45	3	5	80.73	3	2	23.09
	p3	–	3 (5)	246.23	2	4	297.71	4	4	17.57
	p4	–	2 (5)	54.79	3	5	72.69	3	2	22.07
	p5	–	2 (5)	59.87	3	5	86.72	3	2	24.49
Rover	p1	–	2 (2)	3.83	0	1	8.63	0	3	3.24
	p2	–	2 (3)	26.93	1	2	141.20	4	3	9.11
	p3	–	2 (3)	99.01	2	3	165.82	3	3	20.42
	p4	–	3 (4)	102.56	1	3	253.41	1	4	3.97
	p5	–	1 (2)	14.87	0	1	10.58	3	3	18.75

estimate of this model by moving random predicates into the annotated list. By doing this, we ensure that the ground truth model remains in the completion list of this incomplete model. For these tests, we assume all the possible conditions are equally likely. We will now evaluate each of the proposed algorithms using the problems produced above.

Table 1 shows the runtime[2] and the size[3] of the explanations generated by each of the algorithms evaluated on these domain problem pairs. Note that the MEGA*-Conditional was run with γ set to 0.4 and the results for the anytime algorithm only presents the time and size of the first solution found. Also note that both the MEGA*-Conditional search and the MEGA*-Anytime algorithm expect that it can query the human about its ground truth. So each question that the algorithm comes up with is tested against the ground model. The "Question Size" column represents the number of questions that were produced by the search, where each question is related to a single annotated condition. While the "Explanation Size" represents the actual explanation presented to the human. Unlike MEGA*-Conditional and MEGA*-Anytime, MEGA*-Conformant generates no questions but may produce superfluous explanations. Thus, in the "Explanation Size" column for MEGA*-Conformant, we present both the size of the non-superfluous component of the explanation (model updates involving only the known conditions) and the total size of the explanation generated (within parenthesis). The results closely follow intuition. MEGA*-Anytime takes considerably shorter time in most cases, but ends up producing explanations that are longer. While MEGA*-Conformant terminates slightly faster than MEGA*-Conditional, the latter produces shorter explanations whenever possible.

[2]The experiments were run on a Linux workstation with 12 core Intel(R) Xeon(R) CPU and 64G RAM.

[3]There can be more sophisticated measures of the complexity of explanations, other than size, to model cognitive load. However, the techniques introduced here remain largely unchanged since such metrics mainly determine the stopping condition of the search.

Table 2: Runtime for MEGA*-Conformant and the time needed to run MCE for every member of the completion set.

# of models →	2	4	8	16
Baseline	10.95	41.71	195.81	936.30
MEGA*-Conformant	11.11	37.01	117.26	291.88

Finally, as we mentioned in the introduction, one of the major advantages of compiling the set of possible models into $\mathcal{M}_{max}$ and $\mathcal{M}_{min}$ is that we no longer need to compute explanations over each individual model in the set of possible models separately (baseline). Table 2 illustrates the significant scale-ups we can achieve as a result of this.

Model Uncertainty versus Multiplicity: The Case of Multiple Humans in the Loop

While generating explanations for a *set of models*, the robot is essentially trying to cater to multiple human models at the same time. We posit then that the same approaches can be adopted to situations when there are multiple humans in the loop instead of a single human whose model is not known with certainty. As before, computing separate explanations (Chakraborti et al. 2017) for each agent can result in situations where the explanations computed for individual models independently are not consistent across all the possible target domains. In the case of multiple teammates being explained to, this may cause confusion and loss of trust, especially in teaming with humans who are known (Cooke et al. 2013) to rely on shared mental models. Thus *conformant explanations* can find useful applications in dealing with not only model uncertainty but also model multiplicity.

In order to do this, from the set of target human mental models we construct an annotated model so that *the preconditions and effects that appear in all target models become necessary ones, and those that appear in just a subset are possible ones*. As before, we find a single explanation that is a satisfactory explanation for the entire set of models, without having to repeat the standard MRP process over all possible models while coming up with an explanation that can satisfy all of them and thus establish common ground.

While the explanation generation technique may be equivalent, the *explanation process* may be different depending on the setup. For example, while in the case of model uncertainty, the safest approach might be to generate explanations that work for the largest set of possible models, in scenarios with multiple explainees, the robot may have to decide whether it needs to save computational and communication time by generating one explanation to fit all models, or if it needs to tailor the explanation to each human. This choice may depend on the particular domain and the nature of the teaming relationship with the human.

Demonstration on the USAR domain

We go back to our use case, now with *two* human teammates, one external and one internal. *A video of the demonstration is available at `https://youtu.be/hlPTmggRTQA`.* The robot is now positioned at `P1` and is expected to collect data from location `P5`. Before the robot can perform its `surveil` action, it needs to obtain a set of tools from the internal human agent. The human agent is initially located at `P10` and is capable of traveling to reachable locations to meet the robot for the handover. Here the external commander incorrectly believes that the path from `P1` to `P9` is clear and while the one from `P2` to `P3` is closed. The internal human agent, on the other hand, not only believes in the errors mentioned above but is also under the assumption that the path from `P4` to `P5` is not traversable. Due to these different initial states, each of these agents ends up generating a different optimal plan. The plan expected by the external commander requires the robot to move to location `P10` (via `P9`) to meet the human. After collecting the package from the internal agent, the commander expects it to set off to `P5` via `P4`. The internal agent, on the other hand, believes that he needs to travel to `P9` to hand over the package. As he believes that the corridor from `P4` to `P5` is blocked, he expects the robot to take the longer route to `P5` through `P6`, `P7`, and `P8` (orange). Finally, the optimal plan for the robot (blue) involves the robot meeting the human at `P4` on its way to `P5`. Using `MEGA*-Conformant`, we find the smallest explanation, which can explain this plan to both humans.

In this particular case, since the models differ from each other with respect to their initial states, the initial state of the corresponding annotated model, will be defined as –

$\mathcal{I}^0 = \{$`(at_P1), (at_human P10), ..., (clear_path P10 P9), (clear_path P9 P1)`$\}$

$\mathcal{I}^+ = \{$`(clear_path P4 P5), (collapsed_path P4 P5)`$\}$

where $\mathcal{I}^+$ represents the state fluents that may or may not hold in human's model. The corresponding initial states for M_{min} and M_{max} will be as follows –

$\mathcal{I}_{max} = \{$`(at_P1), (at_human P10), ..., (clear_path P10 P9), (clear_path P9 P1), (clear_path P4 P5), (collapsed_path P4 P5)`$\}$

$\mathcal{I}_{min} = \{$`(at_P1), (at_human P10), ..., (clear_path P10 P9), (clear_path P9 P1)`$\}$

`MEGA*-Conformant` generates the following explanation

```
add-INIT-has-clear_path P4 P5
remove-INIT-has-clear_path P1 P9
add-INIT-has-clear_path P2 P3
```

While the last two model changes are equally relevant for both the agents, the first change is specifically designed to help the internal. The first update helps convince the human that the robot can indeed reach the goal through `P4`, while the next two help convince both agents as to why the agents should meet at `P4` rather than other locations.

Conclusion & Future Work

We showed how recently developed techniques for explanation generation as a model reconciliation process can be extended to account for multiple possible models of the explainee – this is useful both in cases where the model of the explainee is uncertain as well as there are many explainees to explain to. We demonstrated this with a robot involved in a typical USAR scenario with an external supervisor whose model of the environment might have drifted in course of time, as well as provided empirical evaluations of the tradeoffs between different kinds (conformant versus conditional versus anytime) of such explanations.

Two immediate directions for future work are (1) developing efficient methods for learning annotated human mental models; and (2) extending the algorithms to work with scenarios where the human mental model is at a different level of abstraction. For (1) it is unrealistic to have access to a large set of plan traces. So it would be interesting to investigate whether we can learn annotated models through data collected from less intrusive and more practical sources than in (Bryce, Benton, and Boldt 2016; Nguyen, Sreedharan, and Kambhampati 2017). There has been some recent work (Nikolaidis et al. 2015; Hadfield-Menell et al. 2016) that aims to learn human mental models iteratively during the course of interactions especially when there is uncertainty about human preferences. With regards to (2), one of the assumptions made in this work is that both the robot and the human represent the world at the same level of fidelity. In recent work (Sreedharan, Srivastava, and Kambhampati 2018), we have started looking at how the robot can deal with human mental models at different level of abstractions (e.g. expert versus non-expert).

Acknowledgements. This research is supported in part by the AFOSR grant FA9550-18-1-0067, the ONR grants N00014161-2892, N00014-13-1-0176, N00014- 13-1-0519, N00014-15-1-2027, and the NASA grant NNX17AD06G. The second author is also supported by the IBM Ph.D. Fellowship from 2016 to 2018.

Algorithm 3 MEGA*-Conditional

procedure AO*-SEARCH
Input: MRP $\langle \pi^*, \langle \mathcal{M}^R, \mathcal{M}^H \rangle \rangle, \gamma$
Output: Explanation $\mathcal{E}^{MCE}$
Procedure:
c_list $\leftarrow$ {} ▷ Closed list
π_R^* $\leftarrow \pi^*$ ▷ Optimal plan being explained
M_{min}^H $\leftarrow$MIN_MODEL(M^H) ▷ Generates M_{min} as per definition
M_{max}^H $\leftarrow$MAX_MODEL(M^H) ▷ Generates M_{max} as per definition
G$\leftarrow$ Node($\langle \mathcal{M}_{min}^H, \mathcal{M}_{max}^H, \{\} \rangle$)
while G.label != Success **do**
current_node $\leftarrow$ Unexpanded_node(G) ▷ Return an unexpanded node in the current best path
S $\leftarrow$ {curent_node}
curent_node.successors $\leftarrow$ GetSuccessors(current_node)
while S not empty() **do**
node $\leftarrow$ S.pop() ▷ Refer to [Nilsson 1980] to see how to prioritize which nodes to remove
min_val $\leftarrow$ 0
label $\leftarrow$ None
for succ in node.successors **do**
if succ is a OR_Succ **then**
node1 $\leftarrow$ succ
if min_val $>$ node1.h_val **then**
min_val = node1.h_val
label = node1.label
if succ is a AND_Succ **then**
node1, node2 $\leftarrow$ succ
tmp_val = min(node1.h_val, node2.h_val) + γ * max(node1.h_val, node2.h_val)
if min_val $>$ tmp_val **then**
min_val = tmp_val
if node1.label == node2.label **then**
label = node1.label
node.label = label
nodel.h_val = min_val
Add all parents of node to S

procedure GETSUCCESSORS(node, $\mathcal{M}^R$)
min_state, max_state $\leftarrow$ node.state
Known_predicates $\leftarrow \Gamma(min\_state) \cap \Gamma(max\_state)$
Possible_predicates $\leftarrow \Gamma(min\_state) \Delta \Gamma(max\_state)$
OR_actions_deletes $\leftarrow$ {Known_predicates $\setminus \Gamma(\mathcal{M}^R)$}
OR_actions_adds{$\Gamma(\mathcal{M}^R) \setminus$Known_predicates}
AND_actions $\leftarrow$ Possible_predicates
Succ_nodes $\leftarrow$ Set()
for a $\in$ OR_actions_adds **do**
tmp_node = Node($\langle \Gamma^{-1}(min\_state \cup a), \Gamma^{-1}(max\_state \cup a) \rangle$)
tmp_node $\leftarrow$ Evaluate_Node(tmp_node)
Succ_nodes.push(OR_succ(tmp_node)))
for a $\in$ OR_actions_deletes **do**
tmp_node = Node($\langle \Gamma^{-1}(min\_state \setminus a), \Gamma^{-1}(max\_state \setminus a) \rangle$)
tmp_node $\leftarrow$ Evaluate_Node(node)
Succ_nodes.push(OR_succ(tmp_node))
for a $\in$ AND_actions **do**
tmp_node1 = Node($\langle \Gamma^{-1}(min\_state \cup a), \Gamma^{-1}(max\_state \cup a) \rangle$)
tmp_node2 = Node($\langle \Gamma^{-1}(min\_state \setminus a), \Gamma^{-1}(max\_state \setminus a) \rangle$)
tmp_node1 $\leftarrow$ Evaluate_Node(tmp_node1)
tmp_node2 $\leftarrow$ Evaluate_Node(tmp_node2)
Succ_nodes.push(AND_succ(tmp_node1, tmp_node2))
return Succ_nodes

procedure EVALUATE_NODE(node)
if Check_For_Goal(tmp_node) **then** ▷ Refer to MEGA*-Conformant for goal test
node.h_val $\leftarrow$ 0
node.label $\leftarrow$ SUCCESS
else
node.h_val $\leftarrow$ heuristic(tmp_node)
return node

References

Albore, A.; Palacios, H.; and Geffner, H. 2009. A Translation-Based Approach to Contingent Planning. In *IJCAI*, 1623–1628.

Bartlett, C. E. 2015. *Communication between Teammates in Urban Search and Rescue*. Master's Thesis, The Polytechnic School, Arizona State University.

Bonet, B., and Geffner, H. 2005. An algorithm better than AO*? In *AAAI*, 1343–1348.

Bryce, D.; Benton, J.; and Boldt, M. W. 2016. Maintaining Evolving Domain Models. In *IJCAI*.

Chakraborti, T., and Kambhampati, S. 2018. Algorithms for the Greater Good! On Mental Modeling and Acceptable Symbiosis in Human-AI Collaboration. *ArXiv e-prints*.

Chakraborti, T.; Sreedharan, S.; Zhang, Y.; and Kambhampati, S. 2017. Plan Explanations as Model Reconciliation: Moving Beyond Explanation as Soliloquy. In *IJCAI*.

Chakraborti, T.; Sreedharan, S.; Grover, S.; and Kambhampati, S. 2018. Plan Explanations as Model Reconciliation – An Empirical Study. *ArXiv e-prints*.

Chakraborti, T.; Sreedharan, S.; and Kambhampati, S. 2017. Balancing Explicability and Explanation in Human-Aware Planning. *ArXiv e-prints*.

Cooke, N. J.; Gorman, J. C.; Myers, C. W.; and Duran, J. L. 2013. Interactive team cognition. *Cognitive science* 37(2):255–285.

Hadfield-Menell, D.; Russell, S. J.; Abbeel, P.; and Dragan, A. 2016. Cooperative inverse reinforcement learning. In *NIPS*, 3909–3917.

International Planning Competition. 2011. IPC Competition Domains. `https://goo.gl/i35bxc`.

Lombrozo, T. 2006. The Structure and Function of Explanations . *Trends in Cognitive Sciences* 10(10):464 – 470.

Lombrozo, T. 2012. Explanation and abductive inference. *Oxford handbook of thinking and reasoning* 260–276.

McDermott, D.; Ghallab, M.; Howe, A.; Knoblock, C.; Ram, A.; Veloso, M.; Weld, D.; and Wilkins, D. 1998. PDDL-the planning domain definition language.

Miller, T. 2017. Explanation in Artificial Intelligence: Insights from the Social Sciences. *CoRR* abs/1706.07269.

Murphy, R. R. 2000. Marsupial and shape-shifting robots for urban search and rescue. *IEEE Intelligent Systems and their applications* 15(2):14–19.

Nguyen, T.; Sreedharan, S.; and Kambhampati, S. 2017. Robust planning with incomplete domain models. *Artificial Intelligence* 245:134 – 161.

Nikolaidis, S.; Lasota, P.; Ramakrishnan, R.; and Shah, J. 2015. Improved human–robot team performance through cross-training, an approach inspired by human team training practices. *International Journal of Robotics Research*.

Nilsson, N. J. 1980. *Principles of artificial intelligence*. Morgan Kaufmann.

Russell, S., and Norvig, P. 2003. *Artificial intelligence: a modern approach*. Prentice Hall.

Sengupta, S.; Chakraborti, T.; Sreedharan, S.; and Kambhampati, S. 2017. Radar-a proactive decision support system for human-in-the-loop planning. In *AAAI Fall Symposium on Human-Agent Groups*.

Sreedharan, S.; Srivastava, S.; and Kambhampati, S. 2018. Hierarchical Expertise-Level Modeling for User Specific Robot-Behavior Explanations. *ArXiv e-prints*.