Step-by-Step Task Plan Explanations Beyond Causal Links

Felix Lindner[1] and Conny Olz[2]

Abstract— **Explainable robotics refers to the challenge of designing robots that can make their decisions transparent to humans. Recently, a number of approaches to task plan explanation have been proposed, which enable robots to explain each step in their plan to humans. These approaches have in common that they are based on the causal links in the plan. We discuss problems with using causal links for plan explanation. Particularly, their inability to distinguish enabling actions from requiring actions can lead to counter-intuitive explanations. We propose an extension that allows for making this relevant distinction and demonstrate how it can be applied to create a robot that explains its actions.**

I. INTRODUCTION

When robots are tasked with making decisions with potentially critical outcomes, it is important that the decision-making process can be made transparent to humans. Guidelines for the ethical design of autonomous systems, such as The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems [1], require robot designers to ensure that "a particular robot's decision should always be discoverable" (i.e., *transparency*), and that robots "[...] shall be created and operated to provide an unambiguous rationale for all decisions made" (i.e., *accountability*). The capacity of robots to explain their behavior to humans has also been argued to be crucial for fostering trust during human-robot interaction [2], [3]. Moreover, explanations can be useful for a robotics engineer's debugging task, or just for making the robot's behavior understandable to a curious person.

One line of research in explainability for robotics develops methods for automatically generating explanations based on actual action sequences a robot has planned to execute [4], [5], [6], [7], [8]. Such approaches are applicable to robot systems that make use of task planning systems to generate high-level representations of their planned actions [9], [10], [11], [12], [13]. A suitable explanation method can then be used to generate a description of each action in the plan along with a justification for why this action is in the plan. A significant number of recent plan explanation methods make use of causal links [14], [15], [5], [8], [16]. Technically, causal links in action plans make transparent the connections between two actions via their effects and preconditions.

We present an analysis of the causal-link approach and argue that it is less suited for explaining robot action plans than suggested by previous work. Particularly, the inability to distinguish enabling actions from requiring actions can lead to counter-intuitive explanations. The remainder of the paper is structured as follows: We first review recent work on explainable robotics with a focus on causal-link-based plan explanation. Section III introduces the technical framework of task planning and causal links as far as needed to understand our analysis. We then analyze the standard approach and identify fundamental issues. In Sect. V, we present our approach to plan explanation, present a demonstrator, and discuss the approach's advantages and disadvantages.

[1]Felix Lindner is with the Institute of Artificial Intelligence, Ulm University, Ulm, Germany felix.lindner@uni-ulm.de

[2]Conny Olz is with the Institute of Artificial Intelligence, Ulm University, Ulm, Germany conny.olz@uni-ulm.de

II. RELATED WORK

How people explain their own behavior to other people has been studied in cognitive and social science for a long time. Results from these research areas have been taken up for proposals of high-level explanation frameworks for robotics that emphasize the social-interactive and human behavior aspects of explanation. Matarese and colleagues [17] stress the importance of causal chains and contrastivity for explaining a robot's actions. The framework by de Graaf and Malle [18] encompasses several types of explaining a robot's behavior by referring to causal histories, reasons, enabling factors, and by making the distinction between explaining intentional and unintentional actions. Both the conceptual frameworks for behavior explanation have a focus on causality, viz., what is it that caused the robot to execute a to-be-explained action, or what is it that was intentionally caused by that action.

One prominent way to computationally account for causality in plan explanation is to make use of causal links between actions in the plan. Stulp and colleagues [5] explain the actions of a robot's task plan by following the causal link chain through the plan. This way, the robot can explain what it is doing, viz., by citing the current action, and why it is executing an action by citing what other actions it enables. PlanVerb [8] is a tool for verbalizing task plan as natural language descriptions. The explanation generation module relies on causal links as a representation of the causal and temporal order of actions. Collins and colleagues [15] suggest generating plan explanations by translating a plan's causal links to an argumentation structure, which can then be queried for explanations of why some action is in the plan. Seegebarth and colleagues [14] generate explanations based on causal links but in a hierarchical planning context, and Sreedharan and colleagues [16] use causal-link explanations in conversational assistants.

The literature review reveals that employing causal links for generating plan explanations is an idea that pops up steadily in recent work. The work by Farrell and Ware [19] analyzes an issue with causal-link explanations in the context of narrative planning, viz., the selection of causal links for

explanation in case of overdetermination. We also briefly discuss this problem in Sect. IV-B.2. We then point out further problems and thereby strive to make a contribution towards better understanding the relation between causal links and plan explanations, but also to make a constructive proposal for an alternative approach.

III. TECHNICAL PRELIMINARIES

A. Task Planning

The goal of task planning is to find a sequence of actions that transforms an initial state into a goal state. The application environment is represented by states, which are described by propositional state variables $v \in \mathcal{V}$, i.e., a state s is a set of facts expressing which state variables are true and which are false in s. A *fact* is a state variable or its negation. The set of all facts is denoted by F. Actions can change one state into another. An action A has preconditions $pre(A)$ and effects $\mathit{eff}(A)$ that are also sets of the aforementioned facts. An action is applicable in a state s if its preconditions are satisfied, i.e., if $pre(A)$ is a subset of s. The application of A in s results in a new state s' where s' is obtained by updating s according to $\mathit{eff}(A)$, i.e., if there is $v \in \mathit{eff}(A)$ but $\neg v \in s$ then $\neg v$ is replaced by v and vice versa, if $\neg v \in \mathit{eff}(A)$ but $v \in s$ then v is replaced by $\neg v$. A sequence of actions $A_0 \circ \cdots \circ A_{n-1}$ is applicable in a state s_0 and results in s_n if the successive application changes the state step-by-step so that all actions are applicable. Formally, this is the case if A_0 is applicable in s_0 and every A_i is applicable in s_i, where s_i results from applying A_{i-1} in s_{i-1}. A planning problem consists of an initial state s_0, a set of actions, and a goal description $s_\star$. The latter is also a set of facts. A plan $\pi = A_0 \circ \ldots \circ A_{n-1}$ is a solution to a planning problem if π is applicable in s_0 and results in a state s_n that satisfies the goal condition, i.e., $g \subseteq s_n$. We further assume that plans do not contain superfluous actions, i.e., removing some action from the plan results in the goal not being reached.

B. Causal Links

In recent times, causal links were mainly exploited for plan explanations. Originally, they descended from least commitment planning where they were used as a tool for finding and validating partially-ordered plans [20], [21].

Let A_1, A_2 be two actions such that A_1 happens before A_2 in plan π. A causal link (A_1, e, A_2) represents that A_1 has e among its effects, that e is part of the precondition of A_2, and no other action between A_1 and A_2 invalidates e. That is, A_1 produces e for A_2, hence, A_1 is also called the *producer* of this causal link and A_2 is called the *consumer*. At the beginning of a chain of causal links, a link with no producer is allowed: (Init, e, A_1) means that e is true in the initial state, maintains its truth until A_1 is executed, and is in one of A_1's preconditions. Likewise, the causal link without consumer (A_1, e, Goal) is allowed at the end of a causal chain representing that e is an actual effect of A_1, remains true until the final state, and is a goal fact (i.e., $e \in s_\star$).

If causal links are not already computed by the planning procedure, one can infer them as follows: Consider a given plan $\pi = A_0 \circ \ldots \circ A_{n-1}$ and goal description $s_\star$. For $s_\star$ and every action A_i and every open precondition e of $s_\star$ or A_i (a precondition not yet supported by a causal link) traverse the actions $A_0, \ldots, A_{i-1}$ in inverse order and stop as soon as you found an action A_j with $e \in \mathit{eff}(A_j)$, then add the link (A_j, e, A_i). Note that generally the choice of a producer of a causal link is not always unique. In the described procedure we pick the "nearest" action, which is in compliance with the algorithm described by Stulp and colleagues [5].

IV. CAUSAL LINK EXPLANATIONS

A. The Standard Approach

Several authors [8], [14], [5] have proposed step-by-step plan explanation methods based on the causal links of a given plan. The main idea is to interpret causal links as models of causal relationships between actions. For the sake of explanation, causal links are given an intentional reading. That is, a causal link of the form (A, e, B) is verbalized as an explanation such as (1).

(1) *The robot executes A in order to achieve e which enables the execution of B.*

We refer to this approach to plan explanations as *the standard approach*. In the following, we scrutinize the standard approach and identify problem areas. Some but not all problems originate from verbalizing causal links using intentional speak. It will also turn out that causal links are no general models of causality, and that they do not support inferences schemes which are often assumed to work well.

B. The Problem With Intentional Language

1) Demanders: Consider a robot asked to serve some coke from the fridge. Moreover, the robot is asked to make sure that there is always coke in the fridge. The robot first serves the coke ($\mathit{ServeCoke}$). As an effect, the coke is served ($\mathit{cokeServed}$) and there is no more coke in the fridge ($\neg \mathit{cokeInFridge}$). Consequently, the robot puts some new coke into the fridge ($\mathit{RefillFridge}$), which results in there being coke in the fridge again ($\mathit{cokeInFridge}$). For the sake of the example, we assume that the fridge provides space for only one coke at a time. Therefore, $\mathit{RefillFridge}$ has the precondition that there is no coke in the fridge. We obtain, among others, the causal links $(\mathit{ServeCoke}, \neg \mathit{cokeInFridge}, \mathit{RefillFridge})$ and $(\mathit{RefillFridge}, \mathit{cokeInFridge}, \mathit{Goal})$. The standard approach thus generates explanations such as (2) or (3). (Here, explanation (3) is an abstraction of (2) in line with the proposal by Canal and colleagues [8].)

(2) *The robot serves coke **to achieve** there is no coke in the fridge, which enables refilling the fridge. Refilling the fridge is executed to achieve there being coke in the fridge.*

(3) *The robot serves coke **to later** refill the fridge to achieve the goal of there being coke in the fridge.*

Explanations (2) and (3) do not sound quite correct. The problem is that the standard approach assumes that every effect which is a precondition of a later action is an intended

effect and thus can be verbalized using intentional language. As the example shows, this is not always the case. Some effects are unintended facts that call for repair actions later on. We call this kind of facts *demanders*. In fact, demanders are omnipresent, e.g., they occur when a robot is navigating an indoor environment while making sure the doors are kept shut, or when robots utter excuses for social norm violations they may commit.

That said, explanation (4) appears way more appropriate.

(4) *Serving coke results in there being no coke in the fridge. This* ***requires*** *refilling the fridge. Refilling the fridge achieves there being coke in the fridge (again).*

It is worth stressing that generating explanation (4) is not merely a verbalization problem but a problem of making fundamental distinctions. Computationally generating it presupposes a way to formally tell apart effects which bring the plan closer to the goal from effects which call for additional repair actions (demanders). It is impossible to make this distinction based on causal links alone.[1] If one still wants to stick to causal-link explanations, one way out is to refrain from intentional language and generate something like explanation (5) instead.

(5) *Serving coke* ***results in*** *there being no coke in the fridge. This* ***enables*** *refilling the fridge. Refilling the fridge achieves there being coke in the fridge.*

This explanation only refers to preconditions and effects and refrains from using words that signal achievements and purpose. A downside of this kind of explanation in the context of social robotics might be that it does not foster the perception of a competently intentional agent.

2) Overdetermination: Overdetermination occurs when a goal or precondition has multiple producers. Consider a robot that has the goal to make sure the human has something to drink ($servedDrink$). The human orders coffee with water. Hence, the robot computes a plan for the goal that consists of the three facts $servedDrink$, $servedCoffee$, and $servedWater$. Two actions $ServeCoffee$ and $ServeWater$ are executed. Serving coffee has the two effects that some drink has been served ($servedDrink$) and that coffee has been served ($servedCoffee$). Serving water has the two effects that some drink has been served ($servedDrink$) and that water has been served ($servedWater$). Among others, the robot's plan gives rise to two causal links $(ServeCoffee, servedDrink, Goal)$ and $(ServeWater, servedDrink, Goal)$. Depending on which of these causal links get verbalized, explanations (6) or (7) will be generated (or both).

(6) *Serving coffee is executed to achieve the goal of having served a drink.*

[1]There is also the causal link $(ServeCoke, cokeServed, Goal)$. One may have the idea that this causal link could be preferred for the explanation over $(ServeCoke, \neg cokeInFridge, RefillFridge)$ by a simple heuristic that favors goals rather than non-goals. However, this will not work in general. For instance, the domain could be extended by a $GettingPaid$ action which has as precondition $cokeServed$ and as effect $getPaid$; And $getPaid$ is set as goal fact instead of $cokeServed$. Then, said problem cannot be resolved by the heuristic.

(7) *Serving water is executed to achieve the goal of having served a drink.*

There is no way to infer from the causal links alone which of the two actions were executed under the intention of serving a drink. To make a choice, the robot would need to consider additional world knowledge, e.g., that coffee is the main drink and the human prefers to have coffee with a glass of water. Note that this problem cannot, in general, be solved by imposing simple heuristics like, for instance, that the first or the last production of an effect is intentional. In fact, the two plans $ServeCoffee \circ ServeWater$ and $ServeWater \circ ServeCoffee$ both achieve the three goal facts. There is nothing that makes a planning system prioritize one plan over the other.

Another way of dealing with overdetermination is, again, to refrain from using intentional language altogether. Explanation (8) just states the relations between actions and their effects in a purely factual manner.

(8) *Both serving coffee and serving water result in having served a drink.*

C. Causality Without Preconditions

We now turn to a problem of the standard approach to plan explanation which does not originate from the lack of intentionality of causal links. What might be even more surprising, causal links are no good models of causality. Consider a robot that uses a planner to navigate a social environment. It therefore has navigational actions which are sensitive to proxemic norms. The robot plans to pass by a human in close vicinity ($ClosePassby$). The action has two effects, viz., the robot reached the goal position ($atGoalPose$), and a proxemic violation has occurred ($violation$). As the robot has the goal that there should be no uncompensated violations, the robot planner adds the action $SaySorry$, which deletes all violations (let us allow for this simplification for the sake of giving a concise example). As $SaySorry$ can always get executed it possibly does not have any preconditions. (It might have preconditions, such as that the text-to-speech module is up running, or so, but, importantly, it does not need to have $violation$ among its preconditions because, technically, nothing prevents the robot from saying sorry even if it is not necessary to do so.) Explaining why the robot said sorry by (9) seems both intuitively correct and desirable. Unfortunately, though, there is no causal link whatsoever between the passing-by action and the say-sorry action.

(9) *The robot says sorry because it passed by closely.*

In order to be able to computationally generate explanation (9), a more sophisticated reasoning procedure is needed. In fact, counterfactual reasoning can be applied to help out: If $ClosePassby$ had not produced a violation, then the $SaySorry$ action would not have been necessary. That is, we have another case of demander facts like in Sect. IV-B.1. This time, however, the question is not whether or not a causal link can be interpreted as intentional. The causal link is missing altogether, and therefore, under the standard approach, no explanation can ever catch the causal relation between the two actions.

D. Transitivity

Causal-link approaches to explanation often assume that the causal-link relation is transitive [8], [14], [5]. Oftentimes, making the transitive inference is appropriate: I turn on the coffee machine to achieve that it is up working, which enables pressing the "brew coffee" button, and pressing that button achieves my having coffee. Therefore, the inference goes, I turn on the coffee machine to achieve my having coffee. More formally, the alleged inference scheme allows inferring the causal link (A, y, C) from (A, x, B) and (B, y, C). Some inference like this would be very useful for plan explanation because it allows summarizing the plan by skipping some intermediate actions. In general, however, the transitive inference scheme is invalid for causal links. To see that, reconsider the example from IV-B.1. There were two causal links $(\mathit{ServeCoke}, \neg \mathit{cokeInFridge}, \mathit{RefillFridge})$ and $(\mathit{RefillFridge}, \mathit{cokeInFridge}, \mathit{Goal})$. Applying the inference scheme, the causal link $(\mathit{ServeCoke}, \mathit{cokeInFridge}, \mathit{Goal})$ is obtained. Verbalizing this causal link would result in something like explanation (10).

(10) *Serving coke achieves there being coke in the fridge.*

This is clearly counter-intuitive under all imaginable readings, intentional or not. This indicates that summarizing plans by skipping intermediate causal links may result in undesired explanations, and doing it right requires much more sophisticated methods.

V. SEMANTIC ROLES FOR PLAN EXPLANATIONS

A. Semantic Roles Connecting Facts to Actions

The outlined analysis in the previous section leaves us with the following problem we are going to propose a solution for: We want to distinguish effects which render subsequent actions necessary from effects that render subsequent actions possible. In the first case, we say that the effect is playing the semantic role DEMANDER, and in the second case, the effect is playing the semantic role ENABLER. An effect can also play both roles at the same time. For instance, in explanation (5), the empty fridge makes the refilling both possible and necessary. We thus write $\textsc{Demander}(\neg \mathit{cokeInFridge}, \mathit{RefillFridge})$ and $\textsc{Enabler}(\neg \mathit{cokeInFridge}, \mathit{RefillFridge})$.

To define the demander role more formally, let $\pi = A_0 \circ \ldots \circ A_{n-1}$ be any plan of n actions. A fact e (positive or negative) that holds at time t is a DEMANDER of action A_t if and only if a subsequence[2] of $A_{t+1} \circ \ldots \circ A_{n-1}$ could reach the goal if, counterfactually, $\neg e$ held at t instead of e. In other words, a shorter plan could reach the goal when the negation of e held at t. In particular, action A_t could then be omitted. In this sense, fact e made action A_t necessary, and we write $\textsc{Demander}(e, A_t)$. Reconsider the coke-serving plan $\mathit{ServeCoke} \circ \mathit{RefillFridge}$. After $\mathit{ServeCoke}$, the fact $\neg \mathit{cokeInFridge}$ holds. If, counterfactually, $\mathit{cokeInFridge}$ had held instead, then the goal could have been reached by $\mathit{ServeCoke}$ alone. That is, $\neg \mathit{cokeInFridge}$ makes $\mathit{RefillFridge}$ necessary, and therefore $\neg \mathit{cokeInFridge}$ is the *demander* of $\mathit{RefillFridge}$.

Note that the demander definition does not require the demander to be a precondition of the demanded action. This can be the case (as in the coke example), but it could be otherwise. Reconsider the socially navigating robot that utters the excuse when passing close by. Here the fact $\mathit{violation}$ is the demander of action $\mathit{SaySorry}$, although $\mathit{violation}$ is no precondition of $\mathit{SaySorry}$. It is true that $\mathit{SaySorry}$ had not been necessary if, counterfactually, $\neg \mathit{violation}$ had held after $\mathit{ClosePassby}$. The definition of demander based on counterfactual analysis thus allows for the distinction between demanders and non-demanders even for actions that have the demander not among their preconditions.

We call preconditions that are no demanders *proper enablers*. That is, if e is a fact true at t, it is not a demander, and action A_t has e among its preconditions, then e is a proper enabler of A_t, and we write $\textsc{Enabler}^*(e, A_t)$.

Finally, we employ the *producer* relation introduced in Sect. III-B. We said an action A_{t_1} is the *producer* of fact e at t_2 if and only if $t_1 < t_2$, e is an effect of A_{t_1}, and none of the actions between $t_1 + 1$ and $t_2 - 1$ have $\neg e$ among its effects. If A_{t_1} is the producer of e at t_2, we write $\textsc{Producer}(A_{t_1}, e_{t_2})$. For example, $\mathit{ServeCoke}$ is the producer of $\mathit{cokeServed}$ at goal state (t_2) because none of the actions in the plan between the $\mathit{ServeCoke}$ action and the goal have $\neg \mathit{cokeServed}$ among their effects. Therefore, we have that $\textsc{Producer}(\mathit{ServeCoke}, \mathit{cokeServed}_{t_2})$ holds.

B. Semantic Roles Connecting Actions to Actions

Causal links connect actions to actions via effects. To accomplish the same, we define two types of links based on the semantic roles defined above. The first link type we call *enabler link*, and we write $\textsc{E-Link}(A, e, B)$. Action A is e-linked to action B via effect e if and only if A is the producer of e and e is a proper enabler of B. The second link type is called *demander link*, and we write $\textsc{D-Link}(A, e, B)$. Action A is d-linked to action B via effect e if and only if A is the producer of e and e is a demander of B.

$$\textsc{E-Link}(A, e, B) \equiv \textsc{Producer}(A, e) \wedge \textsc{Enabler}^*(e, B)$$
$$\textsc{D-Link}(A, e, B) \equiv \textsc{Producer}(A, e) \wedge \textsc{Demander}(e, B)$$

C. Using E-Links and D-Links for Explanation

Employing E-Links and D-Links, instead of the common causal links, can solve the problems analyzed in Sections IV-B.1 and IV-C. To give an idea about the validity of this claim, we reconsider the examples from the previous sections. Beforehand, we briefly introduce a verbalization scheme that employs the distinction between E-Links and D-Links. The following list provides a sample mapping of E-Links and D-Links to natural language. This scheme additionally is sensitive to the cases when the first argument is the initial state or when the last argument is the goal.

- $\textsc{E-Link}(A, e, B)$
  A results in e, which enables B.

[2]By *subsequence* of a plan we mean a plan that is obtained by deleting a possibly empty set of actions from the plan.

- E-LINK$(Init, e, B)$
  e holds initially and enables B.
- E-LINK$(A, e, Goal)$
  A results in e, which fulfills the goal.
- E-LINK$(Init, e, Goal)$
  e holds initially and fulfills the goal.
- D-LINK(A, e, B)
  A results in e, which requires B.
- D-LINK$(Init, e, B)$
  e holds initially and requires B.
- D-LINK$(A, e, Goal)$
  Cannot occur per definition.
- D-LINK$(Init, e, Goal)$
  Cannot occur per definition.

We want to stress that natural-language generation is not our main focus. Hence, the scheme should be understood as one possible way of verbalizing E-Links and D-Links. We will employ it to demonstrate how links systematically map to explanations. Coming back to the analysis of the examples from the previous sections, we obtain the following links for the coke serving example:

E-LINK$(Init, cokeInFridge, ServeCoke)$

D-LINK$(Init, \neg cokeServed, ServeCoke)$

E-LINK$(ServeCoke, cokeServed, Goal)$

D-LINK$(ServeCoke, \neg cokeInFridge, RefillFridge)$

E-LINK$(RefillFridge, cokeInFridge, Goal)$

These links represent the relevant distinction of demanding and enabling actions, and thus warrant the explanation (11) by employing the verbalization scheme.

(11) *There being coke holds initially and enables serving coke. That coke is not served holds initially and requires serving the coke. Serving the coke results in coke being served, which fulfills the goal. Serving coke results in there not being coke in the fridge, which requires refilling the fridge. Refilling the fridge results in there being coke in the fridge, which fulfills the goal.*

For the action plan of the proxemics-aware robot, the following links are generated:

D-LINK$(Init, \neg atGoalPose, ClosePassby)$

E-LINK$(ClosePassby, atGoalPose, Goal)$

D-LINK$(ClosePassby, violation, SaySorry)$

E-LINK$(SaySorry, \neg violation, Goal)$

The verbalization scheme leads to explanation (12).

(12) *Not being at the goal pose holds initially, which requires passing close by. Passing close by results in being at the goal pose, which fulfills the goal. Passing close by results in there being a violation, which requires saying sorry. Saying sorry results in there not being a violation, which fulfills the goal.*

Interestingly, the modeling of the goal specification in the planning problem can influence the explanation: A robot picks up a chair in the office and puts it into the hallway. Let $BringChairToHallway$ be the action with precondition $chairInOffice$ and effects $\neg chairInOffice, chairInHallway$. The goal could either be to bring the chair to the hallway, or, alternatively, to bring the chair out of the office. Action $BringChairToHallway$ achieves both goals. Employing the distinction between E-Links and D-Links, different explanations are obtained depending on whether the plan is executed for the purpose of the first or the second goal. When the goal is to bring the chair to the hallway, then the chair in the office is a proper enabler and explanation (13) is generated. In the other case, the chair is a demander and explanation (14) is generated.

(13) *The chair being in the office holds initially and enables bringing the chair from the office to the hallway.*

(14) *The chair being in the office holds initially and requires bringing the chair from the office to the hallway.*

This way, the distinction between demanding and merely enabling is, in an intuitively appealing way, sensitive to how the goal specification is formulated.

VI. DEMONSTRATOR

We have set up a demonstrator using the TIAGo robot platform from PAL robotics, see Fig. 1. TIAGo has three actions implemented: Asking a human to open the door to the lab, asking a human to close the door to the lab, and moving through the door. The actions are described in terms of their preconditions and effects in a planning domain model:

ASKHUMANTOOPENLABDOOR
pre: $\emptyset$ **eff:** $\{labOpen, \neg soundProtected\}$

ASKHUMANTOCLOSELABDOOR
pre: $\emptyset$ **eff:** $\{\neg labOpen, soundProtected\}$

MOVEOFFICELAB
pre: $\{inOffice\}$ **eff:** $\{\neg inOffice, inLab\}$

In addition to the domain model, the planning problem description contains a model of the current situation. The planning problem that particularly describes the situation depicted in Fig. 1 is given by:

$$s_0 = \{\neg labOpen, inOffice, \neg inLab, soundProtected\}$$
$$s_\star = \{inLab, soundProtected\}$$

The robot is initially located in the office and the door to the lab is closed. Whenever the door to the lab is closed, the office is sound-protected. There is this unwritten rule saying that the door should be kept shut so that noise from the lab does not disturb those people working in the office. Hence, the goal of the planning problem is for the robot to be in the lab and to make sure that the office is sound-protected. Taking both the planning domain and problem description as input, a task planner outputs the plan ASKHUMANTOOPENLABDOOR $\circ$ MOVEOFFICELAB $\circ$ ASKHUMANTOCLOSELABDOOR.[3]

[3]We have used the FastDownward [22] task planner to automatically compute the plan, but any other classical AI planner that takes PDDL as input will work.

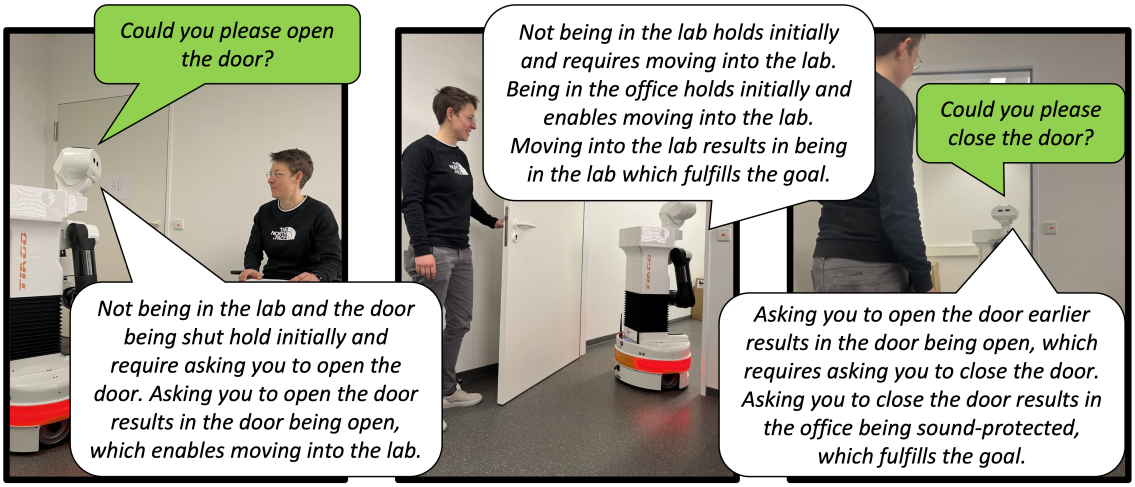

Fig. 1: The robot is executing a three-step plan while explaining each step to the human.

Each action in the plan is associated with a skill implemented on the robot. The plan gets executed by invoking the associated skills one after the other. Each skill accepts an explanation as an additional parameter. Skills that output speech send the explanation to the text-to-speech module after the main utterance was spoken, The move skill starts the explanation utterance in parallel to the movement. The textual explanations are generated according to the verbalization scheme for E-Links and D-Links described in V-C with the following additional considerations: Each explanation starts with the D-Links first and then proceeds with the E-Links. The robot first explains why an action is required and then what it enables. Multiple links get merged to one explanation if they all refer to a fact that initially holds (e.g., the first sentence in the first explanation in Fig. 1). Moreover, when a D-Link refers back to an action that was executed more than one step before, we add the temporal marker *earlier* to the explanation, as exemplified by the first sentence in the last explanation in Fig. 1).

Regarding runtime behavior, computing the E-Links and D-Links along with their verbalizations took 120ms in total. Given that the explanation procedure only needs to run once, viz., after the planning phase, the additional computational footprint is acceptable. All code implementing the generation of E-Links and D-Links, our verbalization procedure, as well as the planning domains and problem specifications (in PDDL), have been made available in a github repository: `https://github.com/existenzquantor/plan-explainer`.

VII. DISCUSSION

The primary aim of our work is to make aware of possible problems that come with employing causal links for plan explanations, and to propose an extension that allows for making the crucial distinction between enabled and demanded actions. Despite our criticism, we grant that the standard approach based on causal links can work fine in many domains. Causal links are computationally easy to compute (viz., in polynomial time) and thus the first choice for application domains where the problem areas we have pointed out do not apply. Our approach to task plan explanation comes with higher computational costs, as deciding whether a fact is a demander of some action is NP-complete (we provide a proof in the appendix). This is unproblematic for shorter plans, but the computational effort for computing D-Links and E-Links will increase exponentially with the size of the plan. We leave a deeper investigation of the computational aspects (runtime behavior, optimizations etc.) for future work.

Our approach also does not provide a solution to summarizing intermediate steps in a plan by transitive links, as problematized in Sect. IV-D. It would be interesting for future research to investigate conditions under which transitive links are acceptable. Also, we leave open the problem of overdetermination (Sect. IV-B.2). We believe that a solution to this problem requires broader world knowledge represented in the system (e.g., knowledge about the practice that coffee gets served with a glass of water on the side, but not the other way round).

While building the demonstrator (Sect. VI), we have identified several empirical research questions: One concerns the timing of explanations, viz., whether an explanation should be uttered before, during, or after the to-be-explained action, or only when explicitly asked for. A second set of questions concerns the modes of verbalizations: should D-Links appear before E-Links in an explanation, or vice versa, should active or passive voice be used, and could non-verbal explanations be generated (e.g., a robot can point at the empty space in the fridge to explain why it is putting a coke into the fridge). We leave these questions for future work.

VIII. CONCLUSIONS

Task plan explanations are considered as one useful tool for implementing explainable robots. Several recent computational approaches to task plan explanations are based on causal links in plans. We have pointed out conceptual problems with this approach. A major issue is that causal links do not allow for a distinction between enabling and

demanding an action, which can lead to counter-intuitive explanations. We have presented a method, based on counterfactual analysis, that allows for computing links that explicitly refer to enabling and demanding actions in a plan. Future research will focus on open problems such as transitivity and overdetermination in link-based explanations, as well as on empirical work on the timing, wording, content, and modality of task plan explanations in human-robot interaction scenarios.

APPENDIX

Let $\pi = A_0 \circ \ldots \circ A_{n-1}$ be a plan of n actions and $s_\star$ the goal. Deciding whether a fact e at step t is a demander of action A_t is NP-complete.

Proof: Membership: Generate (non-deterministically) a subsequence π' of $A_{t+1} \circ \ldots \circ A_{n-1}$ so that π' executed in s'_t results in $s'_n \supseteq s_\star$, where s'_t is obtained from s_t by negating e. If such a π' exists, e is a demander of action A_t at step t. Validating that π' reaches the goal can be done in polynomial time, so the problem is in NP.

We show hardness by a reduction from 3SAT. Consider a 3SAT problem with clauses $c_1, \ldots, c_m$ over the variables $x_1, \ldots, x_k$, where each clause consists of three literals l_{j1}, l_{j2}, l_{j3}. We construct a plan together with an initial state s_0 and goal $s_\star$. Therefore we consider variables $\mathcal{V} = \{p, z, x_1^\top, x_1^\bot, ex_1, \ldots, x_k^\top, x_k^\bot, ex_k, c_1, \ldots, c_m\} \cup \mathcal{Y}$, where $\mathcal{Y} = \{y_{i\mu}^x, y_{j\mu}^c \mid 1 \leq i \leq k, 1 \leq j \leq m, 1 \leq \mu \leq 3\}$. For every variable x_i there are three actions $X_i^\top$ with $pre(X_i^\top) = \{ex_i\}$ and $\mathit{eff}(X_i^\top) = \{x_i^\top, y_{i1}^x, \neg ex_i\}$, $X_i^\bot$ with $pre(X_i^\bot) = \{ex_i\}$ and $\mathit{eff}(X_i^\bot) = \{x_i^\bot, y_{i2}^x, \neg ex_i\}$ and eX_i with $pre(eX_i) = \{z\}$, $\mathit{eff}(eX_i) = \{ex_i, y_{i3}^x\}$. For every literal $l_{j\mu}$ in every clause c_j there is an action $C_{j\mu}$ such that $\mathit{eff}(C_{j\mu}) = \{c_j, y_{j\mu}^c\}$ and if $l_{j\mu} = x_i$ then $pre(C_{j\mu}) = \{x_i^\top\}$ and otherwise if $l_{j\mu} = \neg x_i$ then $pre(C_{j\mu}) = \{x_i^\bot\}$. Finally there is an action P with $pre(P) = \{p\}$ and $\mathit{eff}(P) = \{z, \neg p\} \cup \{\neg y \mid y \in \mathcal{Y}\}$. Note that every action except P adds a unique $y \in \mathcal{Y}$ and P negates all of them. Let $\pi = P \circ X_1^\top \circ eX_1 \circ X_1^\bot \circ \cdots \circ X_k^\top \circ eX_k \circ X_k^\bot \circ C_1 \circ \cdots \circ C_{3m}$, $s_0 = \{p, ex_1 \ldots ex_k\} \cup \mathcal{Y} \cup \{\neg z, \neg x_1^\top, \neg x_1^\bot, \ldots, \neg x_k^\top, \neg x_k^\bot\}$ and $s_\star = \{\neg p, c_1, \ldots, c_m\} \cup \mathcal{Y}$. Now, π is applicable in s_0 and leads to state $s_n \supseteq s_\star$. The plan does not have superfluous actions as only P can negate p thereby also negating all y_i, which are needed in the goal. We check whether p at step 0 is a demander of action P. Upon negating p in s_0 P is not applicable anymore, and consequently all eX_i are inapplicable. Now, either $X_i^\top$ or $X_i^\bot$ must be removed because the preconditions ex_i can consumed only once. Depending on which of $X_i^\top$ or $X_i^\bot$ remain, only a subset of the $C_{j\mu}$ remain applicable. The actions $X_i^\top$ or $X_i^\bot$ simulate whether x_i is set to true or false in the 3SAT problem. The 3SAT problem has a solution iff there is an applicable subsequence of π, i.e., iff p at step 0 is a demander of action P. Hence, the decision problem is NP-hard. ■

REFERENCES

[1] IEEE, "Ethically aligned design, first edition overview – a vision for prioritizing human well-being with autonomous and intelligent systems," Tech. Rep., 2019.

[2] M. Edmonds, F. Gao, H. Liu, X. Xie, S. Qi, B. Rothrock, Y. Zhu, Y. Wu, H. Lu, and S. Zhu, "A tale of two explanations: Enhancing human trust by explaining robot behavior," *Science Robotics*, vol. 4, p. eaay4663, 12 2019.

[3] S. Tolmeijer, A. Weiss, M. Hanheide, F. Lindner, T. Powers, C. Dixon, and M. Tielman, "Taxonomy of trust-relevant failures and mitigation strategies," in *HRI '20: Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, 2020.

[4] S. Rosenthal, S. P. Selvaraj, and M. Veloso, "Verbalization: Narration of autonomous robot experience," in *Proceedings of the Twenty-Fifth Int. Joint Conf. on Artificial Intelligence (IJCAI'16)*, 2016, p. 862–868.

[5] F. Stulp, A. S. Bauer, S. Bustamante Gomez, F. S. Lay, P. Schmaus, and D. Leidner, "Explainability and knowledge representation in robotics: The green button challenge," in *Explainable Logic-Based Knowledge Representation (XLoKR 2020) at the 17th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2020)*, 2020.

[6] B. Krarup, S. Krivic, F. Lindner, and D. Long, "Towards contrastive explanations for comparing the ethics of plans," in *ICRA 2020 Workshop Against Robot Dystopias: Thinking through the ethical, legal and societal issues of robotics and automation (AGAINST-20)*, 2020.

[7] Q. Zhu, V. Perera, M. Wächter, T. Asfour, and M. Veloso, "Autonomous narration of humanoid robot kitchen task experience," in *IEEE-RAS 17th Int. Conf. on Humanoid Robotics (Humanoids'17)*, 2017, pp. 390–397.

[8] G. Canal, S. Krivic, P. Luff, and A. Coles, "Task plan verbalizations with causal justifications," in *ICAPS 21 Workshop on Explainable Planning (XAIP)*, 2021.

[9] L. De Silva, R. Lallement, and R. Alami, "The HATP hierarchical planner: Formalisation and an initial study of its usability and practicality," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'15)*, 2015, pp. 6465–6472.

[10] K. Rajan and A. Saffiotti, "Towards a science of integrated AI and robotics," *Artificial Intelligence*, vol. 247, pp. 1–9, 2017.

[11] J. C. González, J. C. Pulido, and F. Fernández, "A three-layer planning architecture for the autonomous control of rehabilitation therapies based on social robots," *Cognitive Systems Research*, vol. 43, pp. 232–249, 2017.

[12] Y. S. Liang, D. Pellier, H. Fiorino, and S. Pesty, "End-user programming of low-and high-level actions for robotic task planning," in *28th IEEE Int. Conf. on Robot and Human Interactive Communication (RO-MAN'19)*, 2019, pp. 1–8.

[13] M. Faroni, M. Beschi, S. Ghidini, N. Pedrocchi, A. Umbrico, A. Orlandini, and A. Cesta, "A layered control approach to human-aware task and motion planning for human-robot collaboration," in *29th IEEE Int. Conf. on Robot and Human Interactive Communication (RO-MAN'20)*, 2020, pp. 1204–1210.

[14] B. Seegebarth, F. Müller, B. Schattenberg, and S. Biundo, "Making hybrid plans more clear to human users—a formal approach for generating sound explanations," in *Twenty-Second Int. Conf. on Automated Planning and Scheduling (ICAPS'12)*, 2012, pp. 225–233.

[15] A. Collins, D. Magazzeni, and S. Parsons, "Towards an argumentation-based approach to explainable planning," in *ICAPS 19 Workshop on Explainable Planning (XAIP)*, 2019.

[16] S. Sreedharan, T. Chakraborti, Y. Rizk, and Y. Khazaeni, "Explainable composition of aggregated assistants," in *ICAPS 2020 Workshop on Explainable AI Planning*, 2020.

[17] M. Matarese, F. Rea, and A. Sciutti, "A user-centred framework for explainable artificial intelligence in human-robot interaction," in *AAAI Fall Symposium Series – Artificial Intelligence for Human-Robot Interaction (AI-HRI)*, 2021.

[18] M. de Graaf and B. Malle, "How people explain action (and autonomous intelligent systems should too)," in *AAAI Fall Symposium Series*, 2017.

[19] R. Farrell and S. G. Ware, "Causal link semantics for narrative planning using numeric fluents," in *13th AAAI Int. Conf. on Artificial Intelligence and Interactive Digital Entertainment (AIIDE-17)*, 2017, pp. 193–199.

[20] D. McAllester and D. Rosenblitt, "Systematic nonlinear planning," in *9th Nat. Conf. on Artificial Intelligence (AAAI'91)*, 1991, pp. 634–639.

[21] D. S. Weld, "An introduction to least commitment planning," *AI Magazine*, vol. 15, no. 4, pp. 27–61, 1994.

[22] M. Helmert, "The fast downward planning system," *J. Artif. Int. Res.*, vol. 26, no. 1, p. 191–246, jul 2006.