HP-Causality in Multi-agent Planning Tasks

Axel Ind

March 12, 2019

Abstract

The causality proposed by Halpern and Pearl Halpern and Pearl [2005] allows us to determine which variable values are causes of certain other final variable values according to a variety of definitions. The normal HP-Causality approach is limited to a structural equation representations of the problem under consideration. This paper describes a multi-agent planning approach to defining causality. Extensions to the Original, Updated, and Modified definitions of HP-Causality to planning are formalised and discussed.

1 Introduction

Halpern and Pearl Halpern and Pearl [2005] introduced a framework for modelling and evaluating causality. This framework allows examination of modelled problems in terms of a number of standard and author-created definitions of causality.

Though powerful and extensively cited in literature, Halpern's work has yet to be translated into an Artificial Intelligence Planning Setting. The purpose of this paper is to capture the spirit and results of this framework in a multi-agent planning domain. This extension allows us to directly identify agents as causes, rather than just individual variable assignments.

Section @TODOsection describes the mathematical preliminaries of multi-agent planning (@TODOsection) and Halpern's causal models (@TODOsection). @TODOsection describes but-for causes, the original, updated, and modified formulations of HP-causality. @TODOsection provides direct interpretations of these definitions in the planning domain. @TODOsection examines several example which show the power of various causality definitions and contrasts the original HP-causality with the planning interpretation. Finally @TODOsection concludes this paper with a summary of its contributions and potential future work.

2 Planning Preliminaries

2.1 Single Agent Planning Tasks

Language

A single-agent planning task is a 4-tuple $\Pi = (V, A, s_0, s_\star)$. Where V denotes the set of variables and maps each to it's allowable values (or domain) such that, for some variable v, $v \in D_v$. A *fact* is a pair $(v, d \in D_v)$. A conjunction of facts $v_1 = d_1, ..., v_k = d_k$ is *consistent* if it contains no contradictory facts. That is, no pair of facts exists in this conjunction such that $v_j = d_j$ and $v_j \neq d_j$.

A denotes the set of *actions* available to the agent. An action $a =< pre, eff >$ consists of a precondition, and an effect. A precondition is a conjunction of facts $(v_1 = d_1, ..., v_k = d_k)$ and an effect is a conjunction of facts in *effect normal form* (@TODOref) of the form $\phi_i \triangleright v_i := d_i$. No contradictory effects are permitted for a single action. That is no pair of conditionals exists such that $\phi_i \triangleright v_j = d_j$ and $\phi_j \triangleright v_j \neq d_j$. A is divided into two types of actions, endogenous actions and exogenous actions. Intuitively endogenous actions describe those actions an agent is able to choose to take. Lindner et al. [2019] add the requirement that the set of endogenous actions always contains the empty action $\epsilon = (\top, \bot)$, this is not a strict requirement in this paper, but is worth noting as a possible simplifying extension. Exogenous actions describe actions which are outside of the control of the agent and must be performed whenever possible. Lindner et al. [2019] treat exogenous actions as a series of, non-conflicting, actions which are associated with discrete time points $t(a)$ and must be applied as soon as they become applicable and result in a state change. By contrast, this paper drops the requirement that exogenous actions be non-conflicting and instead introduces the permissive and strict approach to dealing with such actions.

The *state* of a classical planning task is the conjunction of all variable pairs. s_0 describes the initial state of the agent, before any actions have been performed. $s_\star$ describes the goal state of the agent, to be achieved by execution of some number of consecutive actions.

Semantics

A action $a =< pre, effect >$ is applicable in a state s iff $s \models pre$ @TODOref. For exogenous actions, it is further required that the current state is the t-th time state associated with that action.

The *change set* $[eff]_s$ of an action is the set of facts that are affected by the conditional. specifically $[eff]_s = \cup_{i=1}^{k} [\phi_i \triangleright v_i := d_i]$, where $[\phi \triangleright v := d]_s = \{v = d\}$ if $s \models phi$ and $\emptyset$ otherwise. The change set never contains contradictory facts.

Applying an actions a to state s results a state s' achieved by replacing all variables in s that occur in $[eff]_s$ with conditional values assigned by that change set.

As with Lindner et al. [2019], we assume *urgent semantics* for exogenous actions. That is, if an exogenous action is applicable, it must be applied immediately and no other action may precede it. However, because this paper allows conflicting exogenous actions, it is not obvious how to deal with cases where multiple exogenous actions are simultaneously applicable. There are three intuitive solutions to this problem:

1. *Ordering approach*: provide a strict ordering relationship among these actions and always apply the applicable action with the highest precedence first (provided it results in a state change).[1]

2. *Permissive approach*: select some random order of exogenous actions.[2]

3. *Strict approach*: consider every possible permutation of exogenous actions that can be performed.[3]

This paper will concern itself only with the third approach. $\Delta_{exo}(s)$ refers to the set of unique states that may be achieved by application of all relevant exogenous actions using one of the schemes above (Note that, unlike Lindner et al. [2019], $\Delta_{exo}(s)$ is not closed and does not guarantee a single unique state in the strict approach).

A *plan* $\pi = (a_1, ...a_n)$ is a sequence of endogenous actions. In a typical plan it is sufficient to apply a_1 to the initials state s_0 to achieve the first state s_1 and then apply a_2, etc. However, to incorporate exogenous actions, $\Delta_{exo}(s)$ must be calculated at each time point in which an exogenous action may occur. If each action in π (supplemented by appropriate intermediate states as a result of $\Delta_{exo}(s)$) is applicable to the preceding state, then the plan is called applicable.

It is worth noting that using the strict approach to exogenous actions above it does not simply suffice to show that some exogenous action in $\Delta_{exo}(s)$ will make the next endogenous action applicable. Instead all applicable exogenous actions must be considered, thus introducing branching complexity into the computation @TODOintroduceexampleofwhythismatters.

2.2 Multi-Agent Action Plans and Planning Tasks

Multi-agent planning tasks extend the intuition of single agent planning tasks as described @TODOsection to allow for consideration of a finite number of agents. Each agent may differ in their allowable actions and goals but are assumed share the same initial state and subsequent states after any endogenous or exogenous action occurs.

A multi-agent planning task Ξ for n agents is a 3-tuple as follows:

$$\Xi =< A, \Pi, T, E >$$

Where $A = (A_1, ...A_k)$ is an ordered list of all agent names in the planning task. Π is the ordered list of agent planning tasks for each agent in A. $\Pi =$

[1] Easy to do, linear time requirements, but forces a possibly unnatural order on events.

[2] Linear time requirements, good for generating a plan, but ignores other potential action sequences.

[3] Exhaustive, but computationally slow.

$(\Pi_1, ..., \Pi_n)$, $\Pi_i = (V, A, s_0, s_\star)$, and T is a scheduling function which determines which agents may act at any given time-point t.

E is the set of all exogenous actions which may occur. These exogenous actions are defined as described in the single-agent planning task context.

This definition intuitively aggregates a set of disparate planning tasks and makes them accessible via unique labels.

Multi-agent action plans are similar to single-agent action plans, but they also encode information about the agent which performed any given action [4]. Further, only multi-agent action plans in which the order of agent labels could feasibly be generated by T are considered valid.

A multi-agent action plan $\pi = ((A_x, \Pi_x[o]_1), ...(A_y, \Pi_y[o]_n))$. Where A_y is the label of the acting agent, and $\Pi_y[o]_n$ describes some applicable operator for agent y at time-point n.

For the purposes of readability, the rest of this paper will assume unique operator names and omit agent labels and specification of the operators available for that agent. Thus, multi-agent planning tasks will be treated as $\pi = (o_1, ...o_n)$, except where doing so would limit some required expressibility.

2.3 Counterfactual Reasoning in Planning

Counterfactual reasoning concerns the question: "What would happen if some feature of the world were other than it is?", and is an important part of many approaches to causality Halpern and Pearl [2005]. For example, we can intuitively grasp concepts such as: "*if the boy had not done his homework, he would not have passed*", which deal with a world different from the one in which we exist. Thus, without going into the minutia of defining causality, as will be done later, we have some intuition that not doing his homework, may have been cause of his failure.

But this reasoning is simplistic. What if the boy's grades were so bad that he would have failed anyway? Could we still consider not doing his homework to be a cause of failure? Our intuitions of cause differ from scenario to scenario, and for this reason we consider four distinct definitions of causality in this paper.

Lindner et al. [2019] introduce and justify a simple counterfactual reasoning procedure for single-agent plans. Specifically it asks what would have happened if some action had not occured and the agent had instead done nothing? In the context of a planning task the authors note that is not sufficient to replace the action slot pair (a_j, o_j) in question with the empty action $\epsilon = (\top, \bot)$ to make (a_j, ϵ), because that action may have caused variable assignment $v = d$ and without that assignment some later action a_{j+k} may no longer be permissible. The authors, instead, allow impermissible actions to be considered during counterfactual reasoning. That is, they assume that, provided the plan was initially

[4] I am aware that this information is trivially obtainable when agent actions have unique names, but I feel that it is more robust to also include explicit labelling information. It allows us, for example, to introduce an arbitrary number of identical new agents to a task without having to change their action labels

valid, we should allow all other actions to occur as they had before, regardless of violated preconditions.

For this paper we consider a slightly more robust choice of actions. Instead of introducing the empty action, we consider the full set of applicable actions available to an agent in a given state. In order to concisely capture the minutia of this kind of counterfactual reasoning we borrow intuition from the causality framework of Halpern and Pearl [2005] and introduce our own framework.

Given a single-agent planning task $\Pi = (V, A, s_0, s_\star)$ and plan $\pi < a_1, ..., a_n >$, we define the *causal plan setting* (CPS) (π, Π). Intuitively the CPS corresponds to the full set of information we have involving the problem. It describes the goal, constraints, and exogenous circumstances that drove the sequence of actions that actually occurred.

Let s_n denote the final state of the agent after each action in the planning task has been applied[5]. The variable assignment for which we would like to identify the cause ϕ is always such that $s_n \models \phi$ in the CPS. The relationship between the CPS and ϕ is described as

$$(\pi, \Pi) \models (\Diamond \phi)$$

Where $\Diamond\phi$ is a symbol borrowed from linear temporal logic (LTL) Galton [1987], and means that ϕ holds in the final state after execution of the plan (analogous to $s_n \models \phi$).

An *action slot* $q(\pi, k)$ denotes the domain of possible actions at a given position. For example, if $\pi = (a_1, a_2, a_3)$, $q(\pi, 1)$ identifies the position occupied by a_1, thus containing action a_1 and any other action that could have been performed by any agent at that moment.

Let $\pi' \subseteq \pi$ denote a situation in which all elements in π' are action slots in π which could have occurred in the same order or else are the empty action. Specifically, for all $a_k \in \pi'$, $a \in q(\pi, k) \cup \epsilon$. The operation $\pi' \subseteq \pi$ is henceforth abbreviated to the symbol π'.

A subplan π', $\pi' \subseteq \pi$ is subset of the actions slots in π. It is not a plan in itself because it describes only applicable actions at a given time, but does not bind itself to a single action as happens in plans. A possible subplan of $\pi = (a_1, a_2, a_3)$ is $\pi' = \{q(\pi, 1), q(\pi, 3)\}$ which tells us all actions which were permissible when the first and last action in π were fixed.

With this notation any plan can be described with the unappealing, but highly informative, use of action slots. So $\pi = (a_1, a_2, a_3)$ from above is equivalent to $\pi = \{(q(\pi, 1), a_1), (q(\pi, 2), a_2), (q(\pi, 3), a_3)\}$

$\pi' \leftarrow \vec{o'}$ is the setting of every non-ϵ action slot a_k to the singleton action $\vec{o'}[k]$.

A counterfactual consideration on a CPS asks the question "What if some subset of actions π' in π had been given the values $\vec{o'}$?". If changing the values of $\vec{o'}$ still results in the values ϕ in the final state, we write:

[5] Although we have stated that plans using conflicting exogenous actions do not necessarily result in a single deterministic final state, our intuition for causality allows us to treat the plan and final state as known quantities examined after the fact. Thus we ensure that there is only one state final state s_n

$$(\pi, \Pi) \models [\pi' \leftarrow \vec{o'}](\Diamond\phi)$$

If, on the other hand, ϕ no longer holds in the final state, we write:

$$(\pi, \Pi) \models [\pi' \leftarrow \vec{o'}](\neg\Diamond\phi)$$

A *multiagent causal plan setting* (MCPS) works in an analogous way. The MCPS (π, Ξ) is the multi-agent planning task combined with the executed plan. Actions slots now also encompass allowable actions by other agents at a given timeslot (remember that which agents may act at that time slot is determined by the scheduling function T).

A counterfactual consideration on a MCPS in which setting the actions slots described by π' to the actions $\vec{o'}$ still results in the values ϕ in the final state, we write:

$$(\pi, \Xi) \models [\pi' \leftarrow \vec{o'}](\Diamond\phi)$$

And if ϕ no longer holds in the final state, we write:

$$(\pi, \Xi) \models [\pi' \leftarrow \vec{o'}](\neg\Diamond\phi)$$

Finally, let $(\pi - \pi')$ denote the plan obtained by setting all non-ϵ actions slots in π' to ϵ in π. For example, given $\pi = (a_1, a_2, a_3)$ and $\pi' = (\epsilon, a_2, \epsilon)$, $(\pi - \pi') = (a_1, \epsilon, a_3)$, where a_i is an action slot.

3 Mathematical Preliminaries HP Causality

@TODOall Discuss the idea of causes. Provide details about the causal setting. Discuss counterfactual reasoning.

Halpern and Pearl Halpern and Pearl [2005] set out to define the cause of variable assignments in the final world. In order to do this they required a rigid and consistent set of operations on representations of the real world.

The first step in the HP approach is to define the *model*. Using this model, it is possible to make statements such as "A is the cause of B in model M according to some definition of causality X." Though we might disagree on precisely which definition of causality to use, we should still achieve unambiguous solutions when that definition is applied.

A model makes use of set of *variables* which may take on various values. In our running example, the boy doing his homework may be a variable HW where $HW = 1$ may denote the boy doing his homework and $HW = 0$ denotes that he has not done his homework. The concept of variables in HP-causality is analogous to that which we have described in planning. However, in HP-causality there is no separation of actions from variables. Thus, as well as having discrete values assigned *a priori*, variable assignments can also occur as the result of *structural equations*. When all variable domains are binary ($X \in \{0,1\}$), any structural equations can be expressed as $X = \phi$ where

$$\phi = \phi|\phi'|max(\phi,\phi')|min(\phi,\phi')$$

Intuitively the $max()$ and $min()$ operations capture the $\wedge$ and $\vee$ operations defined in planning, respectively. And, indeed, the same symbols are often used in HP-causality models.

Halpern and Pearl define *exogenous variables* and *endogenous variable*. Exogenous variables are those variables whose value is determined by factors outside the model. Endogenous variables, by contrast, are defined by structural equations within the model.

Halpern and Pearl enforce an independence criteria on structural equations, so that the sequence in which that are considered does not matter and the final variable assignments are still deterministically defined. To evaluate a causality model, one first assigns the exogenous variables, and then iterates through the structural equation (updating variables at each step) until no new variable assignments occur (@TODOdiagram).

A causal setting (M, u) is the world described by the structural equations of the model M and the exogenous variables u. For a given set of variables *phi* the causal setting exactly describes the value of all those variables after calculation of all structural equations. We write

$$(M, u) \models \phi$$

The final information needed to describe causality in causal settings is a way to represent counterfactual reasoning. To represent the idea of "What would happen if some feature of the world were other than it is", we use variable alternate variable assignments x' for some set of variables X. Thus, if changing X to x' prime still results in variable assignments ϕ we write

$$(M, u) \models [X \leftarrow x']\phi$$

And if ϕ no longer holds after the assignment we write

$$(M, u) \models [X \leftarrow x']\neg\phi$$

4 HP-Causality

This section briefly describes the notion of HP-causality and the definitions and interpretations of But-For Causality, Original Causality, Updated Causality, and Modified Causality.

4.1 But-for Causality

But-for causality is perhaps the simplest and most intuitive definition of causality. For a selected set of variables X it asks "If some, or all, of the variables in X were set to different values, would effect ϕ still occur?"

Definition 4.1. $X = x$ is a but-for cause of ϕ in the causal setting (M, u) iff the following 3 conditions hold:

1. $(M, u) \models (X = x)$ and $(M, u) \models \phi$.
2. $AC2(bf)$: There is a setting x' of the variables in X such that

$$(M, u) \models [X \leftarrow x']\neg\phi$$

3. X is minimal, there is no strict subset of X' of X such that $X' = x'$ satisfies the above conditions.

4.2 The Original definition of HP-Causality

But-for causality is very effective at capturing the most basic intuition about causality, but fails to account for cases where some variables X may have become but-for causes if the values of other variables in the causal setting were different.

The intuition of the original definition of causality (Halpern and Pearl [2005]) is somewhat more complex than that of but-for causality, but it captures a far more nuanced intuition of causality.

Definition 4.2. $X = x$ is an actual cause of ϕ in the causal setting (M, u) according to the original causality definition iff the following 3 conditions hold:

1. $(M, u) \models (X = x)$ and $(M, u) \models \phi$.
2. $AC2(a)$: There is a partition of V (the endogenous variables) into two disjoint subsets Z and W with $X' \subseteq Z'$ and a setting x' and w of the variables in X and W, respectively, such that

$$(M, u) \models [X \leftarrow x', W \leftarrow w]\neg\phi$$

3. $AC2(b^o)$: If $z^\star$ is such that $(M, u) \models Z = z^\star$, then for all subsets Z' of $Z - X$, we have

$$(M, u) \models [X \leftarrow x, W \leftarrow w, Z' \leftarrow z^\star]\phi$$

4. X is minimal, there is no strict subset of X' of X such that $X' = x'$ satisfies the above conditions.

4.3 The Updated definition of HP-Causality

Definition 4.3. $X = x$ is an actual cause of ϕ in the causal setting (M, u) according to the updated causality definition iff the following 3 conditions hold:

1. $(M, u) \models (X = x)$ and $(M, u) \models \phi$.

2. $AC2(a)$: There is a partition of V (the endogenous variables) into two disjoint subsets Z and W with $X' \subseteq Z'$ and a setting x' and w of the variables in X and W, respectively, such that

$$(M, u) \models [X \leftarrow x', W \leftarrow w]\neg\phi$$

3. $AC2(b^u)$: If $z^\star$ is such that $(M, u) \models Z = z^\star$, then for all subsets Z' of $Z - X$ and W' of W, we have

$$(M, u) \models [X \leftarrow x, W' \leftarrow w, Z' \leftarrow z^\star]\phi$$

4. X is minimal, there is no strict subset of X' of X such that $X' = x'$ satisfies the above conditions.

4.4 The Modified definition of HP-Causality

Definition 4.4. $X = x$ is an actual cause of ϕ in the causal setting (M, u) iff the following 3 conditions hold:

1. $(M, u) \models (X = x)$ and $(M, u) \models \phi$.

2. $AC2(a^m)$: There is a set $\vec{W}$ of variables in V and a setting x' of the variables in X such that if $(M, u) \models W = w^*$, then:

$$(M, u) \models [X \leftarrow x', W \leftarrow w^*]\neg\phi$$

3. X is minimal. There is no strict subset of X that satisfies the previous 2 conditions.

Intuitively the first two requirements are: the possible cause must actually have occurred, the final variable assignment under consideration must also have occurred; and changing the possible cause and fixing some number of other variable assignments can change the final variable assignment under consideration.

5 HP-Causality Planning

The HP causality definitions above are restricted to the identification of variable causes, and as such, are not suitable for use in a planning domain where questions posed may also require the identification of a subset of actions or agents as causes.

For this reasons I propose changed definitions of the HP-causality models which produce comparable conclusions given intuitively identical, but structural different problems. Specifically, my definitions allow for the use of actions plans rather than structural equations as original used.

But-for Planning

Definition 5.1. Given a multi-agent action plan π with a final state s_n, some action slots of π, $\pi' \subseteq \pi$, $\pi' \leftarrow o$ are a but-for cause of some final variable assignment $s_n \models \phi$ iff the following 3 conditions hold:

1. $o \subseteq \pi$, $\pi' \models o$ and $s_n \models \phi$.

2. $BF2$: There is a setting o' of the non-ϵ action slots in π' such that:

$$(\pi, \Xi) \models [\pi' \leftarrow o'](\neg\Diamond\phi)$$

3. π' is minimal. There is no strict subset of π' that satisfies the previous 2 conditions.

Original Planning

Definition 5.2. Given a multi-agent planning task Ξ and action plan π with a final state s_n, some actions slots π', $\pi' \subseteq \pi$, $\pi' \leftarrow o$ are a cause of some final variable assignment $s_n \models \phi$ according to the Original planning definition iff:

1. $o \subseteq \pi$, $\pi' \models o$ and $s_n \models \phi$.

2. $BF2(a)$: There is a partition of F (the *endogenous* actions) into two disjoint subplans $Z \subseteq \pi$ and $W \subseteq (\pi - Z)$ with $\pi' \subseteq Z$ and a setting o' and w' of the actions in π' and W respectively such that

$$(\pi, \Xi) \models [\pi' \leftarrow o', W \leftarrow w'](\neg\Diamond\phi)$$

3. $BF2(b^o)$: If $z^\star$ is such that $(\pi, \Xi) \models Z = z^\star$, then for all subplans $Z' \subseteq (Z - \pi')$ we have

$$(\pi, \Xi) \models [\pi \leftarrow o, W \leftarrow w', Z' \leftarrow z^\star](\Diamond\phi)$$

4. π' is minimal. There is no strict subset of π' that satisfies the previous 3 conditions.

(Where $Z \leftarrow z^\star$ denotes fixing all non-ϵ action slots in Z to their original actions.)

Updated Planning

Definition 5.3. Given a multi-agent planning task Ξ and action plan π with a final state s_n, some actions slots π', $\pi' \subseteq \pi$, $\pi' \leftarrow o$ are a cause of some final variable assignment $s_n \models \phi$ according to the Updated planning definition iff:

1. $o \subseteq \pi$, $\pi' \models o$ and $s_n \models \phi$.
2. $BF2(a)$: There is partition of F (the endogenous actions) into two disjoint subplans $Z \subseteq \pi$ and $W \subseteq (\pi - Z)$ with $\pi' \subseteq Z$ and a setting o' and w' of the actions in π and W respectively such that
$$(\pi, \Xi) \models [\pi' \leftarrow o', W \leftarrow w'](\neg\Diamond\phi)$$
3. $BF2(b^u)$: If $z^\star$ is such that $(\pi, \Xi) \models Z = z^\star$, then *for all* subplans $W' \subseteq W$ and $Z' \subseteq (Z - \pi')$ we have
$$(\pi, \Xi) \models [\pi \leftarrow o, W' \leftarrow w', Z \leftarrow z^\star](\Diamond\phi)$$
4. π' is minimal. There is no strict subset of π' that satisfies the previous 3 conditions.

(Where $Z \leftarrow z^\star$ denotes fixing all non-ϵ action slots in Z to their original actions.)

Modified Planning

Definition 5.4. Given a multi-agent action plan π with a final state s_n, some action slots π',$\pi' \subseteq \pi$, $\pi' \leftarrow o$ are a cause of some final variable assignment $s_n \models \phi$ according to the modified planning definition iff the following 3 conditions hold:

1. $o \subseteq \pi$, $\pi' \models o$ and $s_n \models \phi$.
2. $BF2$: There is a setting o' of the applicable actions in π', and a setting of $W \subseteq (\pi - \pi')$ such that:
$$(\pi, \Xi) \models [\pi' \leftarrow o', W \leftarrow w^\star](\neg\Diamond\phi)$$
3. π' is minimal. There is no strict subset of π' that satisfies the previous 2 conditions.

(Where $W \leftarrow w^\star$ denotes fixing all non-ϵ action slots in W to their original actions.)

Agent Causes

Sometimes we may desire to know if a specific agent or subset of agents caused a particular variable assignment to occur. This allows for statements such as "Person X is solely responsible for event ϕ".

Definition 5.5. A subset of agents $\vec{A}$ is a cause of ϕ iff there exists some cause $\vec{X} = \vec{x}$ of ϕ such that every agent in $\vec{A}$ is a label of at least one action in $\vec{X}$, further, only agents mentioned in $\vec{A}$ are action labels in $\vec{X}$.

Notes on the above definitions

- These definitions are strictly more inclusive than but-for cause checks. This is because when $\vec{W} = \{\}$, the $MC2(a^m)$ precisely models but-for causes (under minimality requirements for the but-for cause).
- The action plan obtained by replacing certain actions may no longer be valid. Much like in the normal HP definition of causality, where impossible variable assignments can be fixed during counter-factual reasoning.
- The most significant remaining concern relates to the treatment of exogenous events. Using a single exogenous event for all non-agent actions makes it very difficult to properly assign causality due to problems it causes with fixing variables. This will be detailed in the next section.
- One might wonder why my definitions do not refer to fixing variables at all, and only focus on actions. I have no strong argument for this decision except the observations that fixing some variable $X = x$ can be done in constant time by the inclusion of an action $s = (\top, X = x)$ in the action plan. This action can then be fixed. This addition can be done in $O(|X|)$ time.

6 Examples

6.1 Forest Fire

General Description

Imagine a scenario in which we are in a dry field. We drop a lit match just as lightning strikes the field. A forest fire occurs. What is the cause of the forest fire? Was it dropping the match? Was it the lightning strike? Was it both? Or was it perhaps neither?

Halpern notes that two intuitive possible relations between the forest fire and the lightning and matches make sense. It may be the case that either of the lightning or the dropped match is sufficient to cause the fire (the *disjunctive model*); or else it may be that both are required for the fire to start (the *conjunctive model*).

HP Causality

We use the variables FF, MD, and L to represent the forest fire, the match being dropped, and lightning striking, respectively. These are the endogenous variables. As always, $u = (u_1, u_2)$ is the vector of the exogenous variables. The because we wish to actively manipulate our potential causes, we assign them to the exogenous variables. Thus, $MD \leftarrow u_1$, and $L \leftarrow u_2$. The structural equation expressing whether a forest fire occurs is $FF \leftarrow (MD \vee L)$ in the disjunctive case and $FF \leftarrow (MD \wedge L)$ in the conjunctive case.

Disjunctive Model: Using the original definition of HP-causality and setting $X := MD$ and $W := L$ we find that setting $X \leftarrow \bot$ and $W \leftarrow \bot$ results in $FF = \bot$, satisfying $AC2(a)$. It follows that when X is allowed to retain its original value ($MD = \top$), and W is kept at the counterfactual value ($L = \bot$), then ϕ still occurs ($FF = \top$). This holds true for any value of Z' which can only be the empty set. We achieve the same result using the updated definition of HP-causality. But in the modified definition holding setting $X \leftarrow x'$ and fixing some other variables W to their original value will not affect the value of ϕ. Thus $MD = \top$ is not a cause of FF in this definition. However, the pair of variables $MD = \top$ and $L = \top$ together are a cause.

Conjunctive Model: In this example $AC2(a)$ is satisfied by simply changing the value of X and setting $X := MD$ and leaving W empty. $AC2(b^o)$ is satisfied by this assignment too. The updated version of HP causality holds with identical logic (because W is empty). And finally, (because X is a but-for cause) it follows that X is a cause according to the modified definition too.

@TODOincludeimages

@TODOpossibletable?

Planning Causality

Describing this example in a Planning Causality framework requires us to make some decisions. These decisions have a very significant influence over the results of the model. They can affect its computational complexity, and the intuitive meaning of any conclusions reached.

- *What are the variables?* The most fundamental planning consideration is which variables to include and which to exclude. Only a variable that is explicitly modelled or a composite of other variables can be considered in our definition of causality. In this example we may feel that the variables we use must at least describe whether the forest fire occurred as that is the event for which we desire a cause. Variable domains are just as important; in these examples variable domains will be binary wherever possible.

- *Who are the agents?* Should the arsonist be an agent? This seems fairly intuitive. But should nature also be an agent? Arguments can be made for or against this approach. One might argue that technically, nature has no agency and behaves in a complex, but deterministic, manner. @TODO defines agency as “Agency is the capacity of an actor to act in a given environment.” This definition would seem to imply that the lightning strike is not the act of an agent, but rather of the environment itself. But, using our definition of causality, which encapsulates the entire planning task and plan, it could be argued that the MCPS as a whole is the environment that facilitates action. There are many arguments along these lines that can be made for and against the agency of lightning strikes. For this example we will assume a natural disaster agent which is capable of causing lightning to strike, or of doing nothing.

- *What actions are available to each agent?* We only have knowledge of the causal path and final world that actually resulted. We know, at the very least, that lightning is capable of striking, that the arsonist is able to drop the match, and that a fire can start. What we do not know is what other actions any of these agents might have performed. Is it reasonable to say that, instead of dropping the match, the arsonist decided to do nothing, or make a sandwich, or chase butterflies? All of these actions, a many, many more, are consistent with our understanding of human capabilities, but do they represent different worlds in terms of the variables we wish to model? It may be that in reality making a sandwich or chasing butterflies are completely different activities, but in our model they are essentially the same. They both take us from a world in which the match was dropped, to a counterfactual world in which it was not. For this reason and following the example of Lindner et al. [2019] these examples will allow agents an action that results in nothing being done.

- *What are the exogenous actions?*: Exogenous actions follow similar intuition to endogenous ones. They should be plausible from a counterfactual reasoning perspective. And they should be sufficiently complex to describe preferences and interactions between different aspects of the MCSP. @TO-DOexample illustrates the importance of defining exogenous actions that allow for a more fine-grained approach to causality in planning.

- *What is the scheduling of actions?*: The sequence of allowable actions or actor turns is very important. Are actions simultaneous or parallel? When may exogenous actions occur? Can an agent’s actions affect this scheduling? All of these questions result in different causal models. And, in the absence of certain right answer, the best advice is often to keep the scheduling as simple as it can reasonably be, but no simpler.

Taking these questions into consideration, one possible description of the plan (π), and planning task ($\Xi = (A, \Pi, T, E)$) is $A = (Arsonist, Lightning)$, $\Pi = (\Pi_{arsonist}, \Pi_{lightning})$, $T = (Arsonist, Lightning)$.

$$\Pi_{Arsonist} = (V, A_{Arsonist}, s_0, s_{\star Arsonist})$$

$$V = \{matchDropped, lightningStruck, forestFire\}$$

$$A_{Arsonist} = \{DropMatch = (\top, matchDropped = \top), DontDropMatch = (\top, \bot)\}$$

$$s_0 = \{\neg matchDropped, \neg lightningStruck, \neg forestFire\}$$

$$s_{\star Arsonist} = \{forestFire\}$$

$$\Pi_{Lightning} = (V, A_{Lightning}, s_0, s_{\star Lightning})$$

$$A_{Lightning} = \{StrikeLightning = (\top, lightningStruck = \bot), DontStrikeLightning = (\top, \bot)\}$$

The choice of exogenous actions actions (E), at their simplest, mirrors the conjunctive and disjunctive case discussed in the standard HP setting. Either both *lightningStruck* and *matchDropped* must have occurred, or else only one of them is sufficient.

$$ForestFireCon = (matchDropped \wedge lightningStruck, forrestFire = \top)$$

$$ForestFireDis = (matchDropped \vee lightningStruck, forrestFire = \top)$$

And the plan followed in the actual world is as follows

$$\pi_1 = (DropMatch, StrikeLightning)$$

And, including the exogenous $ForestFire$ event which could be either $ForestFireCon$ or $ForestFireDis$, it becomes

$$\pi = (DropMatch, StrikeLightning, ForestFire)$$

The choice of $\phi := (forestFire)$, and $\pi' = q(\pi, 1)$ (the arsonist's action slot) is obvious from the phrasing of the original example. The conclusions of all three planning causality come to the same conclusions as standard HP causality. In the disjunctive model, $DropMatch$ is not a but-for cause or cause according to the modified definition of $forestFire = \top$, although $DropMatch \wedge StrikeLightning$ is. The original and updated definitions agree that $DropMatch$ is a cause of ϕ. Setting $W := q(\pi, 2)$ (lightning's action) and $w' = StrikeLightning$ and setting $o' := DontStrikeLightning$ satisfies @TODOequation. @TODOequation is satisfied because Z' can only the empty set, and when π' is returned to its original value, ϕ still occurs in the final state.

In the conjunctive case $DropMatch$ is a cause according to all four definition of causality. It is a cause according to the but-for definition of causality, and thus a cause according to the modified definition. It is a cause in the original and updated versions of planning causality because setting $o' := DontDropMatch$ is sufficient to satisfy @TODOequation. It then trivially follows that setting $\pi' \leftarrow o$ results in the real representation of the problem again and

$$(\Xi, \pi) \models (\Diamond \phi)$$

by definition.

Planning Causality

6.2 Rock Throwing

General Description

Imagine a scenario in which Bill and Sue come across a bottle sitting on a windowsill. Being naughty children, they both pick up stones and throw them at the bottle. The bottle shatters. Who is responsible in this scenario?

Intuitively this problem can be modelled in an identical manner to the Forest Fire example, but in Halpern and Pearl [2005] Halpern suggests a different way to model this problem which allows us to differentiate causes more clearly. He argues that one stone (we will assume Sue's) will certainly strike the bottle before the other and, as a consequence, there is a more complex model of this problem which allows differentiation between the two agents.

HP Causality

In his extended example Halpern uses the variables ST (Sue throws) and BT (Bill throws) as variables determined by the exogenous variable $u = (u_1, u_2)$. Additionally we introduce SH (Sue hits) and BH (Bill hits) to show whose rock actually hits the bottle, and BS (Bottle shatters) to indicate if the bottle actually breaks.

The structural equations governing these variables are: $ST \leftarrow u_1$, $BT \leftarrow u_2$, $SH \leftarrow ST$, $BH \leftarrow (\neg ST \wedge BT)$, $BS \leftarrow (BH \vee SH)$. These equations capture our intuition that Sue throws harder, as well as the fact that no matter whose stone hits, the bottle shatters.

All three variants of HP-causality identify ST as a cause of BS, they also agree that BT is not a cause. This is most intuitive in the modified definition where setting X (BT) to some value and fixing some arbitrary set of other variables W to their original value $w^\star$ will never cause ϕ to change because Suzy will still throw, whereas pretending Sue had not thrown and fixing that Billy did not hit will show that ST is a cause. In the original and updated definitions it is possible to set $W = (BT)$ and $w = \bot$, which satisfies $AC2(a)$ and also allows us to satisfy $AC2(b^u)$ by showing that, even when Bill does not throw (and we fix other world events that had occurred), Suzy throwing is still sufficient to cause $BS = \top$.

Planning Causality

To describe these examples in the multi-agent planning domain the scenario must be described in terms of an action plan, and multi-agent planning task. This information we can perform counterfactual reasoning on the CPS.

One possible representation of the multi-agent planning task $\Xi =< A, \Pi, T, E >$ is as follows. $A = (Bill, Sue)$, $\Pi = (\Pi_{Bill}, \Pi_{Sue})$, $T = (Bill, Sue)$ @TODOexogenousactionshere?.

$$\Pi_{Bill} = (V, A_{Bill}, s_0, s_{\star Bill})$$

$$V = billThrew, sueThrew, billHit, sueHit, bottleShattered$$

$$A_{Bill} = \{ThrowBill = (\top, billyThrew = \top), DontThrowBill = (\top, \bot)\}$$

$$s_0 = \{\neg billThrew, \neg sueThrew, \neg billHit, \neg sueHit, \neg bottleShattered\}$$

$$s_{\star Bill} = \{bottleShattered\}$$

$$\Pi_{Sue} = (V, A_{Sue}, s_0, s_{\star Sue})$$

$$A_{Sue} = \{ThrowSue = (\top, sueThrew = \top), DontThrowSue = (\top, \bot)\}$$

The exogenous variables in this situation are only applicable at time point 2 (after both Sue and Bill have taken an action) follows

$$HitsSue = (sueThrew, sueHit = \top)$$

$$HitsBill = (billThrew \land \neg sueThrew, billHit = \top)$$

$$BottleShatters = (sueHit \lor billHit, bottleShattered = \top)$$

The action plan describing the example is as follows

$$\pi_1 = (ThrowBill, ThrowSue)$$

The inclusion of exogenous actions results in the following action plan

$$\pi = (ThrowBill, ThrowSue, SueHits, BillHits, BottleShatters)$$

It is important to note that, if exogenous action sequences were unconstrained, any combination of any number of the actions $SueHits$, $BillHits$, and $BottleShatters$ could occur (provided they result in a state change).

With this formulation of the action plan and MCSP and knowledge that we wish to model causes that result in $bottleShattered$ ($\phi = bottleShattered$), it is now possible to model the actual world

$$(\Xi, \pi) \models (\Diamond \phi)$$

Using this formulation it is immediately apparent that $ThrowSue$ is not a but-for cause of phi in the causal setting, there is no setting of the first action slot (Sue's actions) in π that does not cause $bottleShattered = \top$.

As always, the original and updated models of causality are more complex. Both these definitions limit the action slots that can be manipulated to the endogenous actions of each agent. Thus, setting $Z = q(\pi, 1)$, $X = q(\pi, 1)$, $o' = DontThrowSue$, $W = q(\pi, 2)$, $w' = DontThrowBill$, and executing the exogenous actions as normal is sufficient to validate $BF2(a)$. $BF2(B^o)$ is easily satisfied because Z' is empty, and keeping $ThrowSue = \top$ while holding $W \leftarrow w'$ still results in $bottleShattered = \top$. The updated definition results in similar conclusions.

Our first real deviation from the conclusions of the standard HP causality model comes when we use the modified planning definition. *SueThrows* is *not* a cause according to this definition because there is no way to fix the remaining action slots to their original action which which would cause any value of π' to result in $\neg phi$.

Intuitively this occurs because although *BillHits* was inapplicable in the MCSP, there is no way of preventing it from firing when we counterfactually reason that Sue has not thrown in our current formulation. A simple extension to the modified planning definition allows us to solve this problem. By substituting an empty exogenous action $Empty = (\top, \bot)$ wherever an exogenous action is inapplicable in π, we can now show $ThrowSue$ is a cause of ϕ in the MCSP.

7 Conclusion

@TODOmentionifexogenousactionisanaplicableitstilloccursbutnothinghappens

References

Antony Galton. *Temporal logics and their applications*, volume 10. Academic Press London, 1987.

Joseph Y Halpern and Judea Pearl. Causes and explanations: A structural-model approach. part ii: Explanations. *The British journal for the philosophy of science*, 56(4):889–911, 2005.

Felix Lindner, Robert Matmuller, and Bernhard Nebel. Moral permissibility of action plans. *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.