ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230355

Dynamic Causality

Maksim Gladyshev[a]**, Natasha Alechina**[a]**, Mehdi Dastani**[a]**, Dragan Doder**[a] **and Brian Logan**[a,b]

[a]Utrecht University, Utrecht, The Netherlands
[b]University of Aberdeen, Aberdeen, UK
ORCiD ID: Maksim Gladyshev https://orcid.org/0000-0002-6657-4870,
Natasha Alechina https://orcid.org/0000-0003-3306-9891, Mehdi Dastani https://orcid.org/0000-0002-4641-4087,
Dragan Doder https://orcid.org/0000-0003-0067-3654, Brian Logan https://orcid.org/0000-0003-0648-7107

Abstract. There have been a number of attempts to develop a formal definition of causality that accords with our intuitions about what constitutes a cause. Perhaps the best known is the "modified" definition of actual causality, HP^m, due to Halpern. In this paper, we argue that HP^m gives counterintuitive results for some simple causal models. We propose Dynamic Causality (DC), an alternative semantics for causal models that leads to an alternative definition of causes. DC ascribes the same causes as HP^m on the examples of causal models widely discussed in the literature and ascribes intuitive causes for the kinds of causal models we consider. Moreover, we show that the complexity of determining a cause under the DC definition is lower than for the HP^m definition.

1 Introduction

Causal inference is central to artificial intelligence. For example, it can be used to infer causal structure from data or to infer the causal effect of a particular (hypothetical or actual) event or decision [22, 12]. Two kinds of causal inference can be distinguished. The first is termed 'type causality', and is critical in machine learning and for prediction purposes. This kind of causality concerns general statements such as 'smoking causes lung cancer', and can be used to predict, e.g., the probability that someone who smokes gets lung cancer. The second kind of inference is termed 'actual causality', and is essential in tracing and explaining the cause of a specific outcome, which in turn is essential for assigning responsibility for the outcome to a specific component or decision of an AI system [13, 14, 7, 20, 1, 26, 21]. This kind of inference can be used, for example, to identify the cause of a train accident and to assign the responsibility of the accident to specific decisions. In this paper, we focus on actual causality.

There have been many attempts to define the notion of actual causality and to characterize its corresponding inference system [12, 15, 3]. A 'naive' notion of actual causality is *but-for causality*: event A is the cause of event B if A happened and afterwards B happened, and if A did not happen, B would not have happened [11]. The problem with but-for causality is that sometimes there is some event C that would have caused B even if A did not happen. A classic example concerns the death of a person (event B) who jumps from a ten-story building to commit suicide (event C), but was shot when passing the 9th floor on the way down, killing him instantly (event A). Such considerations gave rise to the HP^m definition that considers a counterfactual state of affairs where A did not happen but also C has the same status as in the actual situation, that is, it did not affect B, and hence in this alternative state of affairs B did not happen. We state the HP^m definition formally below, but essentially it boils down to holding some features of the world to their actual values when they would have been affected by changing the value of the hypothetical cause A.

We argue that the HP^m definition of actual causality gives counterintuitive results for some simple models. One example is the well-known 'Switches' problem due to Hall [8], where a train approaches a switch in the railroad tracks. An engineer can divert the train to the left-hand track instead of the right. In both cases (with and without the intervention of the engineer) the train arrives at its destination, because the tracks reconverge up ahead. According to the HP^m definition of actual causality, one can infer that the engineer's intervention is the cause of the train arriving at its destination, which seems counterintuitive; the train arrives at its destination regardless of what the engineer does. This is because the HP^m definition allows us to assume, counterfactually, that the engineer did not intervene (the train is not on the left-hand track) while keeping 'the train is on the right-hand track' to its actual value which is 'false' (in the actual situation the engineer does intervene such that the train is on the left track). As in this case ('the train is on the left track' and 'the train is on the right track' are both false) the train *does not* arrive at its destination, the HP^m definition concludes that the engineer is the cause of its arrival. As we will see, the counterintuitive implications of the HP^m definition are not limited to this example.

In this paper we propose Dynamic Causality (DC), an alternative semantics for causal models that leads to an alternative definition of causes. DC considers the order in which variables are evaluated in a causal model. It ascribes the same causes as HP^m on the examples of causal models widely discussed in the literature (including those in [11]) and ascribes intuitive causes for the kinds of causal models we consider. Moreover, we show that the complexity of determining a cause under the DC definition is lower than for the HP^m definition.

The remainder of this paper is structured as follows. In Section 2 we recall the formal definition of causal models and the modified definition of actual causality HP^m. In Section 3 we present some counterexamples to the HP^m definition. Finally, in Section 4 we propose the dynamic interpretation of causal models, that gives rise to a new definition of Dynamic Causality (DC). We demonstrate how the DC definition deals with the counterexamples to the HP^m definition and prove that the complexity of verifying a cause under the DC definition is lower than for the HP^m definition. Finally, we discuss how our approach is related to other attempts to define actual causation proposed in recent years, e.g. [2], [4] and [3].

2 Preliminaries

In this section we briefly recall the necessary background on causal models and the HP^m definition of causality. The presentation below essentially follows that in [9, 11].

2.1 Causal Models

The idea of describing causal models as a collection of structural equations was introduced by Pearl [22]. In [10] Halpern provided axiomatizations for different classes of causal models, and in [13, 14] Halpern and Pearl developed formal definitions of *cause* and *explanation*.

The Halpern and Pearl approach (hereafter HP) assumes that the world is described in terms of *variables* and their *values*. Some variables may have a causal influence on others. This influence is modelled by a set of *modifiable structural equations*. The variables are split into two sets: the *exogenous* variables, whose values are determined by factors outside the model, and the *endogenous* variables, whose values are ultimately determined by the exogenous variables. The structural equations describe how the outcome is determined. Formally, a causal model is defined as:

Definition 1 (Causal Model)**.** *A signature is a tuple $S = (\mathcal{U}, \mathcal{V}, \mathcal{R})$, where $\mathcal{U}$ is a finite set of exogenous variables, $\mathcal{V}$ is a finite set of endogenous variables, and $\mathcal{R}$ associates with every variable $Y \in \mathcal{U} \cup \mathcal{V}$ a finite nonempty set $\mathcal{R}(Y)$ of possible values for Y, also called range of Y. A causal model over a signature $\mathcal{S}$ is a tuple $M = (\mathcal{S}, \mathcal{F})$, where $\mathcal{F}$ associates with every endogenous variable $X \in \mathcal{V}$ a function $\mathcal{F}_X$ such that $\mathcal{F}_X$ maps $\times_{Z \in (\mathcal{U} \cup \mathcal{V} - \{X\})} \mathcal{R}(Z)$ to $\mathcal{R}(X)$.*

Intuitively, $\mathcal{F}_X$ defines a structural equation that specifies how the value of the endogenous variable X is determined by the values of all other variables in $(\mathcal{U} \cup \mathcal{V}) - \{X\}$. For example, in a causal model with three variables X, Y and Z, the function $\mathcal{F}_X(Y, Z) = Y + Z$ defines the structural equation $X = Y + Z$, while $\mathcal{F}_Y(X, Z) = Z$ defines the structural equation $Y = Z$, etc. The later equation demonstrates that Y does not depend on X. Additionally, these equations can be written with an 'iff' notation, for example $X = 1$ iff $\min(\mathrm{Y}, \mathrm{Z}) = 0$, and $X = 0$ iff $\min(\mathrm{Y}, \mathrm{Z}) \neq 0$. For the case of binary variables it is often more convenient to define structural equations using boolean connectives, e.g. X = $\neg(Y \lor X)$. So, by structural equation for any endogenous variable X we understand the way of specifying how the value of X is determined by the values of other variables[1].

An assignment $\vec{\mathcal{U}} = \vec{u}$ of all exogenous variables is called *context*.[2] We will slightly abuse the notation and use $\vec{u}$ to refer to the context instead of $\vec{\mathcal{U}} = \vec{u}$.

The causal dependencies between variables can be visualised using a dependency graph, consisting of nodes representing variables and directed edges representing causal dependencies. Though dependency graphs do not provide any new information apart of those already contained in $\mathcal{M}$ they often serve as a good illustration of $\mathcal{M}$. In order to generate this graph for some causal model $\mathcal{M}$, we need to check the dependencies between our variables. We say that Y depends on X if there is some setting of all the variables other than X and Y such that varying the value of X in that setting results in a variation in the value of Y. More formally, for all $X, Y \in \mathcal{U} \cup \mathcal{V}$, and $\vec{Z} = (\mathcal{U} \cup \mathcal{V}) - (X \cup Y)$, we say that Y depends on X if there is some $\vec{z}$, such that $\mathcal{F}_Y(X = x, \vec{Z} = \vec{z}) \neq \mathcal{F}_Y(X = x', \vec{Z} = \vec{z})$, where $x \neq x'$. Additionally, note that all $\mathcal{F}_X$ take as input the assignment of all variables from $(\mathcal{U} \cup \mathcal{V}) - X$. But we will slightly abuse the notation and allow expressions $\mathcal{F}_X(\vec{Y} = \vec{y})$, where $\vec{Y} \subsetneq (\mathcal{U} \cup \mathcal{V}) - X$, i.e. $\vec{Y}$ is a proper subset of $(\mathcal{U} \cup \mathcal{V}) - X$. We say that $\mathcal{F}_X(\vec{Y} = \vec{y}) = x$ if $\mathcal{F}_X(\vec{Y} = \vec{y}, \vec{Z} = \vec{z}) = x$ for all $\vec{z}$, where $(\vec{Y} \cup \vec{Z}) = (\mathcal{U} \cup \mathcal{V})$. The choice of exogenous variables is usually trivial since they can be considered as 'dummy' variables enforcing the necessary values of those endogenous variables that depend only on these exogenous ones.

In this paper we restrict our attention to *recursive* models [11] only. In such models, the dependency graph is acyclic.

Binary causal models are models $\mathcal{M}$ for which $\mathcal{R}(Y)$ contains only two values for each $Y \in \mathcal{U} \cup \mathcal{V}$.

To illustrate the notion of a causal model, consider the following simple example, initially due to Lewis [19] which has been influential in the development of actual causality.

Example 1 (Rock-throwing example)**.** *Suzy and Billy both pick up rocks and throw them at a bottle (encoded as ST=1 and BT=1 respectively). Suzy's rock gets there first, shattering the bottle. We denote the fact that Suzy's rock hits the bottle as SH=1. Similarly, BH=0 denotes the fact that Billy's rock does not hit the bottle. Finally, BS=1 means 'the bottle shatters'. We also know that because both throws are perfectly accurate, Billy's would have shattered the bottle had it not been preempted by Suzy's throw. So, our endogenous variables $\mathcal{V}$ are $\{ST, BT, SH, BH, BS\}$. Structural equations are defined as follows:*

- *SH=ST;*
- *BH=(BT∧¬SH);*
- *BS=(SH∨BH).*

Our exogenous variables $\mathcal{U} = \{U_{\mathrm{ST}}, U_{\mathrm{BT}}\}$ determine the values of ST and BT variables respectively. For simplicity, we omit exogenous variables from the dependency graph.

Note that structural equations contain the information about any counterfactual scenario, not only about the actual one. For example, we know that if Suzy had thrown the rock, but by some reason her rock would not have hit the bottle, the bottle would have been still shattered, because in this scenario Billy's rock would have hit the bottle. So, causal models provide us a powerful tool for dealing with counterfactuals, even when alternative situations violate the causal structure of the actual situation, in this example, when they violate the equation SH=1 iff ST=1. In such cases, we can 'break' certain causal relations to explore causal dependencies between other variables. The dependency graph for Example 1 is presented in Figure 1.

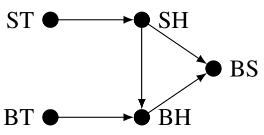

Figure 1. A dependency graph for the Rock-throwing example.

[1] The detailed overview can be found in [11].

[2] We use the notation $\vec{X} = \vec{x}$ to abbreviate $(X_1 = x_1 \land \cdots \land X_k = x_k)$, where $\vec{X} \subseteq \mathcal{U} \cup \mathcal{V}$ and $|\vec{X}| = k$. We also slightly abuse the notation and write $(X = x) \in (\vec{X} = \vec{x})$ meaning that $(X = x)$ is a conjunct of $(\vec{X} = \vec{x})$.

The main feature of causal models is their ability to express facts about any counterfactual scenario. As we mentioned before, we can say that if Suzy did not throw the rock, her rock would not hit the bottle, but the bottle would still be shattered. We can express it with a formula $[\mathrm{ST} \leftarrow 0](\mathrm{SH{=}0} \wedge \mathrm{BS{=}1})$. The operator $[\vec{X} \leftarrow \vec{x}]$ is called *intervention*. This intervention results in a new casual model denoted $\mathcal{M}_{\vec{X}\leftarrow\vec{x}}$. Informally, $\mathcal{M}_{\vec{X}\leftarrow\vec{x}}$ is model $\mathcal{M}$ in which functions $\mathcal{F}_X$ for any $X \in \vec{X}$ are replaced with a constant function $\mathcal{F}_X^{\vec{X}\leftarrow\vec{x}}$, which always returns X, where $X = x \in \vec{X} \leftarrow \vec{x}$ and the remaining functions remain unchanged. Note that an intervention $\vec{X} \leftarrow \vec{x}$ can be seen as a set of variable assignments $\{X_1 \leftarrow x_1, \dots, X_k \leftarrow x_k\}$ and we write $X = x \in \vec{X} \leftarrow \vec{x}$ if $X \leftarrow x$ appears in $\vec{X} \leftarrow \vec{x}$.

Definition 2 (Updated model)**.** *Given a causal model $\mathcal{M} = (\mathcal{S}, \mathcal{F})$ and assignment $\vec{X} = \vec{x}$ of any subset of $\mathcal{V}$, we call $\mathcal{M}_{\vec{X}\leftarrow\vec{x}} = (\mathcal{S}, \mathcal{F}^{\vec{X}\leftarrow\vec{x}})$ an updated model, where for all Y, $\vec{z}$*

$$\mathcal{F}_Y^{\vec{X}\leftarrow\vec{x}}(\vec{z}) = \begin{cases} \mathcal{F}_Y(\vec{z}), \ \textit{if } Y \notin \vec{X} \\ x', \ \textit{otherwise, where } (Y = x') \in \vec{X} = \vec{x} \end{cases}$$

Now we can formally define the syntax of the basic causal language, that allows us to reason about basic causal formulas of the form $(X = x)$, their boolean combinations and interventions $[\vec{Y} \leftarrow \vec{y}]\varphi$.

Definition 3 (Syntax)**.** *Given a signature $\mathcal{S} = (\mathcal{U}, \mathcal{V}, \mathcal{R})$, a primitive event is a formula of the form $X = x$, for $X \in \mathcal{V}$ and $x \in \mathcal{R}(X)$. A causal formula (over $\mathcal{S}$) is one of the form $[Y_1 \leftarrow y_1, \dots, Y_k \leftarrow y_k]\varphi$, where φ is a Boolean combination of primitive events, $\{Y_1, \dots, Y_k\} \subseteq \mathcal{V}$, $y_i \in \mathcal{R}(Y_i)$.*

Language for $\mathcal{S} = (\mathcal{U}, \mathcal{V}, \mathcal{R})$ consists of all Boolean combinations of causal formulas, where the variables in the formulas are taken from $\mathcal{V}$ and the sets of possible values of these variables are determined by $\mathcal{R}$.

The well-formed formulas of our language are only those constructed according to the above mentioned rules. It remains to define the truth relation $\models$ for causal formulas with respect to causal models. We call a pair $(\mathcal{M}, \vec{u})$ a *causal setting* for a model $\mathcal{M}$ and context $\vec{u}$. Given a causal setting $(\mathcal{M}, \vec{u})$ and a causal formula φ we interpret $(\mathcal{M}, \vec{u}) \models \varphi$ notation as 'formula φ is true at $(\mathcal{M}, \vec{u})$'. Finally, let $Sol(\vec{u})$ be a set of all $(X = x)$, such that X has a value x in the unique solution of equations in $\mathcal{M}$ for a context $\vec{u}$. Now we are ready to define the semantics of causal formulas.

Definition 4 (Semantics)**.** *For a causal model $\mathcal{M} = (\mathcal{S}, \mathcal{F})$, a context $\vec{u}$ and a causal formula φ we define the relation $\models$ inductively as follows:*
$(\mathcal{M}, \vec{u}) \models (X = x)$ *iff* $(X = x) \in Sol(\vec{u})$;
$(\mathcal{M}, \vec{u}) \models \neg\varphi$ *iff* $(\mathcal{M}, \vec{u}) \not\models \varphi$;
$(\mathcal{M}, \vec{u}) \models (\varphi \wedge \psi)$ *iff* $(\mathcal{M}, \vec{u}) \models \varphi$ *and* $(\mathcal{M}, \vec{u}) \models \psi$;
$(\mathcal{M}, \vec{u}) \models [\vec{Y} \leftarrow \vec{y}]\varphi$ *iff* $(\mathcal{M}_{\vec{Y}\leftarrow\vec{y}}, \vec{u}) \models \varphi$.

2.2 The HP^m Definition of Cause

Almost all existing approaches to causality are essentially counterfactual theories of causation [18]. And the most common of them and widely used in legal practice is so-called *but-for* causality. According to this approach, A is a cause of B if, but for A, B would not have happened. In other words, but-for definition says that A caused B iff A and B both occurred and had A not occurred, B would not have occurred.

Definition 5 (But-for cause)**.** *We say that $\vec{X} = \vec{x}$ is a but-for cause of φ in $(\mathcal{M}, \vec{u})$ if the following three conditions hold:*
$\mathbf{BC1}$. $(\mathcal{M}, \vec{u}) \models (\vec{X} = \vec{x})$ *and* $(\mathcal{M}, \vec{u}) \models \varphi$
$\mathbf{BC2}$. $(\mathcal{M}, \vec{u}) \models [\vec{X} \leftarrow \vec{x}']\neg\varphi$
$\mathbf{BC3}$. *$\vec{X}$ is minimal: no proper subset of $\vec{X}$ satisfies* $\mathbf{BC2}$.

BC1 condition ensures that both $\vec{X} = \vec{x}$ and φ hold in the actual context. BC2 is a but-for condition saying that if $\vec{X} = \vec{x}$ had not been true, φ would have been false. And BC3 guarantees that only essential elements of the conjunction $\vec{X} = \vec{x}$ are considered part of a cause while inessential variables are pruned. This definition in fact looks quite natural, but with the development of causal theory it became clear that Definition 5 cannot resist the pressure of various counterexamples. In many situations we intuitively agree that $\vec{X} = \vec{x}$ is a cause of φ, but $\vec{X} = \vec{x}$ and φ do not satisfy Definition 5. We can observe this problem in Example 1. If both Suzy and Billy throw the rock, then we know that it was exactly Suzy's rock hit the bottle. So, it is very natural to say that ST=1 is a cause of BS=1 in this example since it was exactly ST=1 that lead to SH=1 that made BS=1 true. But ST=1 fails the but-for test: $[\mathrm{ST} \leftarrow 0]\neg(BS = 1)$ does not hold in $(\mathcal{M}, \vec{u})$, since the bottle is shattered even if Suzy does not throw the rock.

The most famous and commonly used solution called HP definition was proposed by Judea Pearl and Joseph Halpern [22, 13, 11]. This approach is usually called *actual* causality. Its final version is introduced in [9] and is called HP^m (for modified) definition.

Definition 6 (HP^m cause[3])**.** *We say that $\vec{X} = \vec{x}$ is an actual cause of φ in $(\mathcal{M}, \vec{u})$ if the following three conditions hold:*
$\mathbf{AC1}$. $(\mathcal{M}, \vec{u}) \models (\vec{X} = \vec{x})$ *and* $(\mathcal{M}, \vec{u}) \models \varphi$
$\mathbf{AC2m}$. *There is a set $\vec{W}$ of variables in $\mathcal{V}$, such that if $(\mathcal{M}, \vec{u}) \models \vec{W} = \vec{w}^*$, then*

$$(\mathcal{M}, \vec{u}) \models [\vec{X} \leftarrow \vec{x}', \vec{W} \leftarrow \vec{w}^*]\neg\varphi$$

$\mathbf{AC3}$. *$\vec{X}$ is minimal: no proper subset of $\vec{X}$ satisfies* $\mathbf{AC2m}$.

It is easy to see that AC1 and AC3 conditions are identical to those from Definition 5. The main difference can be observed in AC2m. Intuitively, it says that $\vec{X} = \vec{x}$ is a cause of φ if we can find some subset $\vec{W}$ of endogenous variables, such that fixing the values of all variables in $\vec{W}$ to their original values in the actual context would make $\vec{X} = \vec{x}$ a but-for cause of φ. To illustrate this intuition, consider Example 1 once again. We want to check if HP^m definition determines Suzy as a cause of bottle shattering, on which but-for definition fails. Conditions AC1 and AC3 trivially hold. So, we need to find $\vec{W}$, such that $[\mathrm{ST} \leftarrow 0, \vec{W} \leftarrow \vec{w}^*]\neg(\mathrm{BS{=}1})$ to satisfy AC2m. And BH is the candidate we need. We know that in the actual context $\vec{u}$, BH=0 holds. So, fixing it to original values results in a formula $[\mathrm{ST} \leftarrow 0, \mathrm{BH} \leftarrow 0]\neg(\mathrm{BS{=}1})$ and it is easy to check that this formula holds in $(\mathcal{M}, \vec{u})$. So, ST=1 is a cause of BS=1 in our settings $(\mathcal{M}, \vec{u})$ according to HP^m definition. The Rock-throwing example is the an illustration of so-called *Late Preemption*.

[3] In fact, there are three versions of HP definition, all with the same basic structure. In this paper we discuss HP^m [11], which is the latest and simplest version. The other versions are the HP^o (original) and HP^u (updated) definitions (a detailed overview of all definitions can be found in [11]). Here we mention only that HP^m gives a more intuitive ascription of causes than HP^o and HP^u in some cases. For example, in situations where so-called *overdetermination* occurs. Consider a voting scenario with 11 voters and simple majority rule. Assume that Suzy wins 11-0. In this case both HP^o and HP^u claim each of the voters for Suzy to be a cause, while HP^m picks any subset of 6 voters.

3 Problems With The HP^m Definition

It can be fairly said that HP^m outperforms but-for approach in a sense that this definition provides us more natural solutions. In many cases it is easier to agree that HP^m solutions are what we would call a cause rather than but-for solutions. But this approach also has some drawbacks. To illustrate it consider the following example [8, 4].

Example 2 (Switches)**.** *An engineer is standing by a switch in the railroad tracks. A train approaches in the distance. She flips the switch, so that the train travels down the left-hand track (LT), instead of the right (RT). Since the tracks reconverge up ahead, the train arrives at its destination ($Dest$) all the same.*

The causal dependencies are represented by the following structural equations:

- $Dest := LT \vee RT;$
- $LT := Switches;$
- $RT := \neg Switches.$

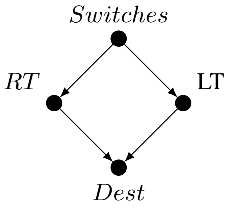

Figure 2. Dependency graphs for Example 2.

It seems reasonable that flipping the switch is not a cause for the train's arrival in this example, as the train would have arrived at its destination whether the switch was flipped or not. But HP^m definition violates this intuition. It is easy to check that $Switches = 1$ is an HP^m cause of $Dest = 1$: take $\vec{W} = \{\text{RT}\}$, so $(\mathcal{M}, \vec{u}) \vDash [\text{Switches} \leftarrow 0, RT \leftarrow 0]Dest = 0$, so (AC2m) is satisfied. AC1 and AC3 are trivially satisfied.

Another counterexample for HP^m definition is a kind of causal models, in which an outcome φ holds no matter how we manipulate the other variables.

Example 3. *A major AI conference has two phases of the reviewing process. In the first phase ($P1$), papers are judged by the suitability of their abstracts (U_a). A paper whose abstract is good ($U_a = 1$) passes the first phase ($P1 = 1$) and reaches the second phase ($P2$), where it is reviewed based on its content (U_c). Then it gets negative reviews ($P2 = 2$) if the content is not good ($U_c = 0$), and positive reviews ($P2 = 1$) if the content is good ($U_c = 1$). If a paper does not pass the first stage ($P1 = 2$) it doesn't get any reviews in the second phase ($P2 = 0$). Finally, the conference chair decides to accept ($CD = 1$) the papers that reach the second phase and receive positive reviews, and to reject ($CD = 0$) other papers. In this example variables P1 and CD are binary, and P2 has three values: P1 = 1 means that review is positive and P1 = 2 means that it is negative. Similarly for P2, but additionally there is an option that second review is not provided at all (P2=0). This situation is captured by the following structural equations:*

- *P1=1 if U_a=1; P1=2 if U_a=0;*
- *P2=0 if P1=2; P2=1 if (P1=1∧U_c=1);*
  P2=2 if (P1=1∧ U_c=0);
- *CD=1 if P1=1 and P2=1; CD=0 otherwise.*

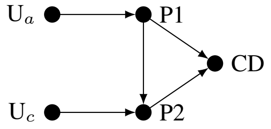

Figure 3. Dependency graph for Example 3.

Consider a submission with bad abstract and bad content, which determines the actual context $\vec{u}$: ($U_a = 0, U_c = 0$). The paper does not pass the first phase ($P1 = 2$), doesn't get any reviews in the second phase ($P2 = 0$) and gets rejected ($CD = 0$). It seems natural to claim that $P1 = 2$ (not passing the first phase) is the cause of rejection ($CD = 0$). But both but-for and HP^m definitions claim that $P1 = 2$ and $P2 = 0$ together are a cause of $CD = 0$, since $[P1 \leftarrow 1, P2 \leftarrow 1]CD = 1$ holds in $(\mathcal{M}, \vec{u})$ satisfying BC2 (and AC2m). In is also easy to check that $\{P1 = 1, P2 = 1\}$ is minimal: no proper subset of it satisfies BC2 (and AC2m).

4 Dynamic Causal Models

In this section we present dynamic causal models or, more precisely, the dynamic interpretation of causal models. The dynamic interpretation is based on the idea that not only the solution of structural equations must be taken into account, but also how the solution is computed based on the dependencies among variables. Below, we show how the order in which values can be assigned to variables in $\mathcal{V}$ given $(\mathcal{M}, \vec{u})$ may be used to determine actual causality.

Recall that in Example 1, in the actual context, the event 'Bottle shatters' depends on the event 'Suzy's rock hits the bottle', which in turn depends on 'Suzy throws the rock'. And since events in causal models are represented by the assignment of values to variables, we want to understand in which order values must be assigned to the corresponding variables. For example, it is clear from the causal structure of the model that the value of BS cannot be computed before the value of SH is known (given that $BH = 0$ in the actual context) and SH cannot be computed until the value of ST is available. This illustrates that causal models already contain all the information about the order in which values can be assigned to variables; we only need to extract and use this information in the definition of causality. In what follows, we specify the order in which value can be assigned to variables.

Recall that $\mathcal{F}$ is a finite set of $\mathcal{F}_X$ for $X \in \mathcal{V}$. Let $\vec{Y} = \vec{y}$ denote an assignment of a (possibly empty) set of endogenous variables in $\mathcal{V}$. Given $(\mathcal{M}, \vec{u})$, an assignment $\vec{Y} = \vec{y}$ substitutes $\vec{u}$ and $\vec{Y} = \vec{y}$ to $\mathcal{F}_X$ for all X in $\mathcal{M}$ and determines the values of a set of variables $\vec{Y'} = \vec{y'}$. Note that $(\vec{u}, \vec{Y} = \vec{y})$ is not necessarily a complete assignment for $\mathcal{U} \cup \mathcal{V}$. By a complete assignment we mean an assignment where all variables are assigned a value. The only variables X that will be assigned a value given $(\vec{u}, \vec{Y} = \vec{y})$ are those for which $\mathcal{F}_X$ returns the same value on any assignment extending $(\vec{u}, \vec{Y} = \vec{y})$ to a complete assignment to all variables apart from X. We say that $\vec{Y} = \vec{y}$ is *sufficient* for $X = x$ in $(\mathcal{M}, \vec{u})$ if $\mathcal{F}_X(\vec{u}, \vec{Y} = \vec{y}) = x$ for all complete assignments [3]. Finally, we stipulate that the computation of assignments is performed in a step-wise manner. Given $(\mathcal{M}, \vec{u})$, we start with an an empty assignment $\vec{Y} = \vec{y}$ and compute the assignment $\vec{Y'} = \vec{y'}$ of variables whose values can be determined given (only) $\vec{u}$. (Note that $\vec{Y'}$ is guaranteed to be non-empty by the recursiveness of $\mathcal{M}$.) We then compute the assignment $\vec{Y''} = \vec{y''}$ of variables whose values can be determined, given the assignment $\vec{Y'} = \vec{y'}$. We repeat this process until no new assignments can be made, i.e., until the assignment is sufficient for all $X \in \mathcal{V}$.

As an example, consider the number of steps necessary to achieve a sufficient assignment for all variables in the Rock-throwing example. At step 0 only the context $\vec{u}$ is known, and the only variables whose value can be determined are $(ST = 1, BT = 1)$. Given the assignment $(ST = 1, BT = 1)$, at step 1, we can determine the value of the variable $SH = 1$. Given the assignment $(ST = 1, BT = 1, SH = 1)$, at step 2 we can determine the value of the variables $BH = 0$ and $BS = 1$. This is because $SH = 1$ is enough to compute $BS = 1$, whatever the value of BH. So, the computation of a sufficient assignment for all variables in this example requires three steps (see Figure 4).

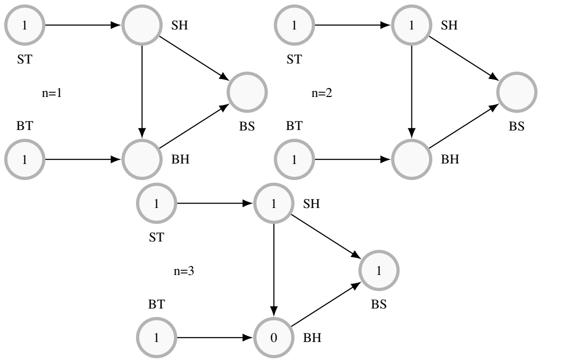

Figure 4. Dynamic interpretation of the Rock-throwing example. n denotes number of calls to $\mathcal{F}$, i.e. step of the computation.

More formally, assume that we are given a causal model $\mathcal{M} = (\mathcal{S}, \mathcal{F})$. A *computation* over $(\mathcal{M}, \vec{u})$, denoted by $\mathcal{C}$, is a function mapping $\mathbb{N}$ to assignments $\vec{Y} \times \mathcal{R}(\vec{Y})$, where $\vec{Y} \in 2^{\mathcal{U} \cup \mathcal{V}}$, which is constructed as follows. $\mathcal{C}(0) = \{U_1 = u_1, \dots, U_k = u_k\}$, where $(U_1 = u_1, \dots, U_k = u_k)$ is a context $\vec{u}$. For all $n > 0$ we require that: (1) the context $\vec{u}$ is contained in $\mathcal{C}(n)$, i.e. $U_1 = u_1 \in \mathcal{C}(n), \dots, U_k = u_k \in \mathcal{C}(n)$; and (2) for all $X \in \mathcal{V}, X = x \in \mathcal{C}(n)$ iff $\mathcal{F}_X(\mathcal{C}(n-1)) = x$. Since $\mathcal{C}(n-1)$ is not necessarily a complete assignment of all variables, $\mathcal{F}_X(\mathcal{C}(n-1))$ returns some value x only if $\mathcal{F}_X(\mathcal{C}(n-1), \vec{Y} = \vec{y'}) = x$ for all $\vec{Y} = \vec{y'}$ that complete $\mathcal{C}(n-1)$ to all variables. This procedure basically describes how the computation over $(\mathcal{M}, \vec{u})$ must be performed in a step-wise manner. Note also that since $\mathcal{M}$ is recursive, $\mathcal{C}$ behaves as follows: (1) once any $X = x$ appears at $\mathcal{C}(i)$, it remains in any $\mathcal{C}(j)$ for $j > i$ and (2) there always exists n', such that all variables appear in $\mathcal{C}(n')$ and then $\mathcal{C}(n'') = \mathcal{C}(n')$ for $n'' > n'$.

The introduction of computations requires a new definition of interventions. Recall that in HP^m, an intervention $\mathcal{F}^{X \leftarrow x}$ is the result of replacing $\mathcal{F}_X$ with a constant function and leaving the remaining functions unchanged. However, this approach is inconsistent with a dynamic interpretation of causal models. Replacing $\mathcal{F}_X$ with a constant function breaks the order of computation, because this constant function will always return the same value x for any input and thus X will be already defined at $\mathcal{C}(1)$, independently of when X was defined in the original computation. Instead, we define interventions with respect to a computation $\mathcal{C}$ as follows.

Given $(\mathcal{M}, \vec{u})$, let $\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}$ be a computation for an intervention $\vec{Y} \leftarrow \vec{y}$, such that $\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(0) = \vec{u}$ and for all $n > 0$ if $X \notin \vec{Y}$, then $X = x \in \mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n)$ iff $\mathcal{F}_X(\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n-1)) = x$, and if $Y \in \vec{Y}$, then $Y = y \in \mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n)$ iff $\exists y' \in \mathcal{R}(Y)$ such that $\mathcal{F}_Y(\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n-1)) = y'$ and $Y = y \in \vec{Y} \leftarrow \vec{y}$. In other words, at each step n of the computation we check if some X_i in the resulting assignment appears in $\vec{Y}$, and if yes, we replace $X_i = x$ with $X_i = x' \in \vec{Y} \leftarrow \vec{y}$ in $\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n)$. Otherwise, if X_i does not occur in $\vec{Y}$, we add $X_i = x$ to $\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n)$, where $\mathcal{F}_X(\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n-1)) = x$.

Finally, we give a revised definition of semantics of causal formulas for dynamic causal models. This definition is similar to the HP^m definition (Definition 4). The main difference is that it is no longer sufficient to evaluate the truth of formulas with respect to causal settings $(\mathcal{M}, \vec{u})$. Instead we do it with respect to *causal states* of the form $(\mathcal{M}, \vec{u}[n])$.

Definition 7 (Semantics)**.** *Given a causal model $\mathcal{M}$, a context $\vec{u}$, a natural number n and a causal formula φ we define the relation $\models$ inductively as follows:*
$(\mathcal{M}, \vec{u}[n]) \models X = x$ *iff* $X = x \in \mathcal{C}(n)$,
$(\mathcal{M}, \vec{u}[n]) \models \neg\varphi$ *iff φ can be evaluated to false in* $\mathcal{C}(n)$,
$(\mathcal{M}, \vec{u}[n]) \models (\varphi \wedge \psi)$ *iff* $(\mathcal{M}, \vec{u}[n]) \models \varphi$ *and* $(\mathcal{M}, \vec{u}[n]) \models \psi$,
$(\mathcal{M}, \vec{u}[n]) \models [\vec{Y} \leftarrow \vec{y}]\varphi$ *iff φ can be evaluated to true in* $\mathcal{C}^{\vec{Y} \leftarrow \vec{y}}(n)$.

Note that in the truth definition φ does not necessarily have a truth value at n.

In addition, we use $(M, \vec{u}) \models \varphi$ as an abbreviation for $(M, \vec{u}[\infty]) \models \varphi$. For this case, the relation $\models$ for the dynamic interpretation corresponds to that of Definition 4.

4.1 Dynamic Causality

Now that all technical details have been dealt with, we can present the dynamic interpretation of causality. Before giving the formal definition, we give a motivating example. Recall why Suzy fails the but-for test in the Rock-throwing example. If we think in terms of static causal models, then it is clear that $[\mathrm{ST} \leftarrow 0](BS = 1)$ holds in the actual context $(\mathcal{M}, \vec{u})$. The choice of the value for ST changes nothing in terms of the resulting value of BS, and so ST=1 is not a but-for cause of BS=1. We now consider the intervention $(\mathcal{M}_{\mathrm{ST} \leftarrow 0}, \vec{u})$ under the dynamic interpretation.

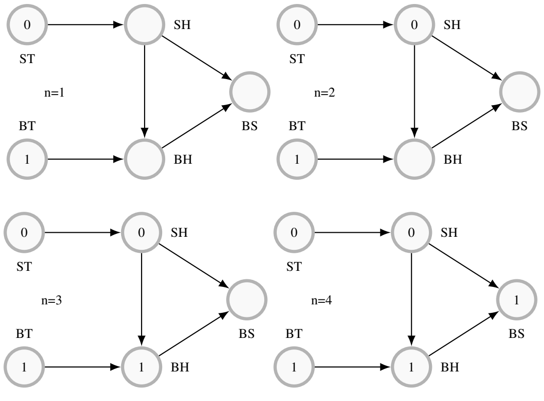

Figure 5. Computation for $(\mathcal{M}_{\mathrm{ST} \leftarrow 0}, \vec{u})$.

In this computation SH and BH are again computed on steps 2 and 3 respectively. However, while previously we could calculate BS given only SH (if SH=1 holds on the previous step, then BS=1 becomes defined at the next step), now we cannot do it, since SH=0 in the new computation. We have to 'wait' one more step to get the value of BH and then compute BS. Note that, although the intervention on ST does not prevent $BS = 1$ from occurring in $(\mathcal{M}, \vec{u})$, $ST \leftarrow 0$ 'pushes' the moment of computation of BS to a later step. In other words, $ST \leftarrow 0$ cannot prevent $BS = 1$, but it does prevent $BS = 1$ from being computed at the step (step 3) at which it would have been computed in the actual context.

Now we are ready to formalize these ideas and provide a new definition of actual cause with respect to a computation over a causal model. We use the notation n_φ to denote the first step of the calculation for $(\mathcal{M}, \vec{u})$ when φ becomes true; otherwise $n_\varphi = \infty$.

Definition 8 (Dynamic Cause)**.** *$\vec{X} = \vec{x}$ is a* DC *cause of φ in $(\mathcal{M}, \vec{u})$ if the following three conditions hold:*
$\mathbf{DC1}$. $(\mathcal{M}, \vec{u}) \models (\vec{X} = \vec{x})$ *and* $(\mathcal{M}, \vec{u}) \models \varphi$
$\mathbf{DC2}$. $(\mathcal{M}, \vec{u}[n_\varphi]) \not\models [\vec{X} \leftarrow \vec{x}']\varphi$
$\mathbf{DC3}$. *$\vec{X}$ is minimal: no proper subset of $\vec{X}$ satisfies* $\mathbf{DC2}$

Conditions DC1 and DC3 are the same as the corresponding conditions in the But-for and HP^m definitions. Condition DC2 is essentially a but-for condition, but with respect to a specific step of the computation. It says that $\vec{X} = \vec{x}$ satisfies the (dynamic) but-for condition if $[\vec{X} \leftarrow \vec{x}']$ would prevent φ from being evaluated to true at the step of computation at which it would have become true in the actual context.

DC ascribes the same causes as HP^m on the examples of causal models widely discussed in the literature. However DC and HP^m ascribe different causes in Examples 2 and 3 discussed above.

Proposition 1. *An* HP^m *cause is not a* DC *cause.*

Proof. Recall Example 2. In Example 2, HP^m picks $Switches = 1$ as a cause of $Dest = 1$: take $\vec{W} = \{RT\}$. Fixing $\vec{W}$ to the original value and intervening on $Switches$ will result in $Dest = 0$. So, $(\mathcal{M}, \vec{u}) \models [Switches \leftarrow 0, RT \leftarrow 0]Dest = 0$ holds and, thus, (AC2m) is satisfied. However, according to the DC definition, $Switches = 1$ is not a cause of $Dest = 1$, since $(\mathcal{M}, \vec{u}[n_{(Dest=1)}]) \models [\text{Switches} \leftarrow 0]Dest = 1$ holds, so DC2 is not satisfied.

□

Proposition 2. *A* DC *cause is not an* HP^m *cause.*

Proof. Recall Example 3. Both but-for and HP^m pick (P1=2∧P2=0) as a cause of CD=0, but it is not a cause for DC. DC picks only P1=2. It is trivial to check conditions DC1 and DC3. For DC2, in the original context CD=0 becomes true at step 2 of the computation, but in the computation for $(\mathcal{M}_{P1\leftarrow 1}, \vec{u})$ it becomes defined only at step 3. So, $(\mathcal{M}, \vec{u}[2]) \not\models [P1 \leftarrow 1](CD = 0)$ holds satisfying DC2. □

4.2 Complexity

Finally, we briefly discuss complexity results for the DC definition, and compare them to the HP^m definition. Given $(\mathcal{M}_{rec}, \vec{u}, \vec{X}, \vec{x}, \varphi)$ we want to check if $\vec{X} = \vec{x}$ is a *cause* of φ in $(\mathcal{M}, \vec{u})$ according to DC.

Theorem 1. *The complexity of determining whether $\vec{X} = \vec{x}$ is a cause of φ in $(\mathcal{M}_{rec}, \vec{u})$ under* DC *is:*
(a) D^P-complete in the general case;
*(b) co-*NP*-complete for binary causal models; and*
(c) in PTIME if $|\vec{X}| = 1$.

Proof. (a) The proof will appear in the extended version of the paper. (b) It is easy to see that the problem is in co-NP. Conditions DC1 and DC2 can be checked in polynomial time. We need the assumption of binary variables to guarantee there is a unique $\vec{x}'$ of values different from $\vec{x}$; otherwise we would need to check all possible $\vec{x}'$. Checking DC3 is in co-NP because we can guess a counterexample (a strict subset of $\vec{X}$ which satisfies DC2).

For co-NP hardness, consider the following reduction from the *minimal model* problem: given a propositional formula φ in conjunctive normal form and a propositional assignment α, is α a minimal model of φ? Minimality is defined with respect to the pointwise order on assignments, where $0 \leq 1$. In other words, an assignment is minimal if any assignment with strictly fewer 1s does not make the formula true. This problem was introduced and proved co-NP complete in [6]. Let φ be a propositional formula in conjunctive normal form over propositional variables $Y_1, \dots, Y_k$, and α an assignment of values to $Y_1, \dots, Y_k$. Consider a causal model M with exogeneous variable U and endogeneous variables $Y_1, \dots, Y_k$, where the structural equations are $Y_i = U$ for $i \in \{1, \dots, k\}$. Let $\{X_1, \dots, X_m\} \subseteq \{Y_1, \dots, Y_k\}$ be the set variables assigned 1 by α. Without loss of generality, $m > 0$, since an assignment of all 0s is trivially minimal. We claim that $\vec{X} = \vec{0}$ is a cause of $\neg\varphi$ in $(M, \vec{0})$ if, and only if, α is the minimal model of φ.

For the left to right direction: suppose that DC1 – DC3 hold:
$\mathbf{DC1}$. $(M, 0) \models (\vec{X} = \vec{0})$ and $(M, 0) \models \neg\varphi$
$\mathbf{DC2}$. $(M, 0[n_\varphi]) \not\models [\vec{X} \leftarrow \vec{1}]\neg\varphi$
$\mathbf{DC3}$. $\vec{X}$ is minimal: no proper subset of $\vec{X}$ satisfies $\mathbf{DC2}$ This entails that setting $\vec{X}$ to $\vec{1}$ is sufficient to prevent φ from being false at n_φ, and $\vec{X}$ is a minimal such set, that is, α is a minimal model of φ.

For the right to left direction, suppose α is a minimal model of φ. Then (by our assumption that α assigns 1 to at least one variable), an assignment of all 0s does not satisfy φ, so DC1 holds. Since it is a satisfying assignment, assigning 1s to the variables $\vec{X}$ prevents φ from being evaluated to false at n_φ. And from minimality of α, DC3 holds.

(c) It is clear that checking if DC1 holds can be done in polynomial time and DC3 holds trivially. For DC2 it is sufficient to check $|\mathcal{R}(X)| - 1$ candidates for x' and each candidate can be checked in polynomial time. □

For the HP^m definition (a) and (b) are D^P-complete [11] (the proof of D^P-hardness uses a binary model). The problem for unary causes (c) under the HP^m definition is NP-complete [11]. Hence the complexity of determining whether something is a cause for binary causal models and for unary causes is lower under DC than under the HP^m definition, and for unary causes, tractable.

5 Discussion and Future Work

The dynamic interpretation of causal models provides a new perspective on reasoning about causality. The DC interpretation can be seen as a modification of the satisfiability relation $\models$ for causal formulas. The definition of causal models and the syntax are unchanged, which means that many results from static causal models can be applied directly to dynamic ones. Secondly, dynamic causal models give rise to a natural definition of cause, which improves HP^m by identifying intuitive causes in some situations and behaving better than HP^m in terms of computational complexity, making the problem tractable in the case of single variables, which covers many interesting situations.

In the discussion and comparisons above, we focused on HP^m, as HP-style definitions are the most common in the field, and it could be argued that HP^m represents the 'state of the art' in such approaches. However, HP^m is not the only candidate for the definition of actual causation. Various alternative definitions have recently been proposed. In particular, Beckers and Vennekens [4] (hereafter BV) introduce several principles, such that counterfactual dependence and production, and propose a new definition of actual causation incorporating these principles. Another feature of their approach is that it extends a causal setting $(\mathcal{M}, \vec{u})$ with extra temporal information, called timing τ. Timing is a function that maps literals of the form $(X = x)$ to natural numbers, and represents the temporal order in which events happen in the actual context. That is, $\tau(X = x)$ represents the moment at which the event $X = x$ happens. Causality statements are expressed relative to a timing as tuples of the form $(\mathcal{M}, \vec{u}, \tau)$. While this is superficially similar to our notation $(\mathcal{M}, \vec{u}[n])$ (especially given the similarity between temporal moment $\tau(X = x)$ at which $X = x$ happens and the first moment n of the computation $\mathcal{C}(n)$ at which X can be assigned with value x), there are significant differences between BV and DC. Firstly, [4] assumes that the timing τ is given outside the model, but agrees with dependencies between variables (ensuring that the cause does not happen after effect in the timing). We provide a way to generate a computation $\mathcal{C}$ using only the information given in $(\mathcal{M}, \vec{u})$. DC is therefore applicable when no additional temporal information is available, which is not the case for BV. Secondly, we do not claim that the order of computation described by $\mathcal{C}$ represents any temporal information about the order in which events occurred in the actual context. Clearly, the same causal setting $(\mathcal{M}, \vec{u})$ always generates the same computation $\mathcal{C}$, but there may be multiple valid timings $\tau_1, \dots, \tau_k$ over $(\mathcal{M}, \vec{u})$, such that the BV definition picks different causes for different timings. However, if the order of variable assignments in the computation $\mathcal{C}$ is interpreted as a timing τ, it is one of the valid temporal orderings according to [4]. We believe that our approach provides a simple and elegant definition of a cause compared to both HP^m and BV, and remains applicable when only a model $\mathcal{M}$ and a context $\vec{u}$ are given. Additionally, we show that the problem of verifying a cause is tractable for DC for the important class of unary causes, while [4] do not consider issues of complexity.

Another recently proposed definition of actual cause is the CNESS definition [3]. CNESS can be seen as a compromise between the BV definition and Wright's *Necessary Element of a Sufficient Set* (NESS) [25] definition. The NESS definition states that $C = c$ NESS-causes $E = e$ w.r.t. $(\mathcal{M}, \vec{u})$ if there exists a chain of *direct* NESS causes from $C = c$ to $E = e$. $C = c$ is a *direct* NESS-cause of $E = e$ in $(\mathcal{M}, \vec{u})$ if there exists a witness $\vec{W} = \vec{w}$ so that: (1) $(\mathcal{M}, \vec{u}) \models C = c \wedge \vec{W} = \vec{w}$; (2) $\{C = c, \vec{W} = \vec{w}\}$ is *sufficient* for $E = e$; and (3) $\vec{W} = \vec{w}$ is not sufficient for $E = e$. Recall that $\vec{X} = \vec{x}$ is sufficient for $Y = y$ in $(\mathcal{M}, \vec{u})$ if $\mathcal{F}_Y(\vec{u}, \vec{X} = \vec{x}) = y$. The BV and CNESS definitions of cause are formulated in terms of NESS-causes as follows. $C = c$ BV-causes $E = e$ in $(\mathcal{M}, \vec{u})$ if $C = c$ *NESS-causes* $E = e$ in $(\mathcal{M}, \vec{u})$ and there exists a $c' \in \mathcal{R}(C)$ such that $C = c'$ does not NESS-cause $E = e$ in $(\mathcal{M}_{C \leftarrow c'}, \vec{u})$. $C = c$ *CNESS-causes* $E = e$ in $(\mathcal{M}, \vec{u})$ if $C = c$ NESS-causes $E = e$ along some path p and there exists a $c' \in \mathcal{R}(C)$ such that $C = c'$ does not NESS-cause $E = e$ along any subpath p' of p in $(\mathcal{M}_{C \leftarrow c'}, \vec{u})$. Where 'NESS-causes along some path' means that the values of the variables in p form a chain of direct NESS-causes from $C = c$ to $E = e$ (see [3] for details). Note that, similarly to HP-style definitions, NESS uses the witness $\{\vec{W} = \vec{w}\}$. This increases the complexity of both BV and CNESS, as they use the notion of a direct NESS-cause. As noted above, [3] does not provide complexity results. However, since a NESS-cause requires checking exponentially many candidates for a witness $\vec{W} = \vec{w}$ to verify if $C = c$ is a cause of $E = e$, our conjecture is that the problem of verifying NESS, BV or CNESS is at least NP-hard. In contrast, for DC this problem is in PTIME for single variables C and E.

NESS, BV and CNESS are similar in a sense that they are all based on the idea of checking chains of direct NESS-causes, while DC uses the idea of performing a step-by-step computation. However the idea of checking causal chains between a cause and effect can be compared to the idea of checking if the change in cause affects the moment of computation on which the effect can be computed. Another difference is that both BV and CNESS verify two conditions: (1) that $C = c$ causes $E = e$ in $(\mathcal{M}, \vec{u})$ and (2) $C = c'$ does not cause $E = e$ in $(\mathcal{M}_{C \leftarrow c'}, \vec{u})$. We believe that this feature is used to deal with 'Switches'-style models (see Example 2). As DC does not pick $Switches = 1$ as a cause of $Dest = 1$ in this type of model, so second conditions seems to be redundant in our approach. However, it is an open question whether these definitions agree on all examples or not, so a detailed comparison of DC with NESS, BV and CNESS is a promising direction for future work.

An alternative attempt to define NESS-style definition was made by Bochman in [5]. Another recent paper [24] proposes an actual cause definition for action languages which is also based on Wright's NESS test. There are also recent papers studying actual causality in situation calculus semantics [16, 2, 17], but as they use a significantly different formalism than DC, we leave a detailed comparison with these definitions for future work.

We conclude by briefly outlining some directions for future work that build on dynamic causal models. Firstly, in this paper we presented only example-based arguments in favour of DC, while there have been some recent proposals to define general principles of actual causation [4]. We believe that the study of this general principles is a priority direction for future work.

Another important direction is application of the proposed semantics to non-recursive models. For static models non-recursiveness means that the set of structural equations can have no (or multiple) solutions. One way to deal with this problem is to adapt the idea proposed in [11] and to take the truth of a primitive events $X = x$ to be relative not just to a context, but to a complete assignment: i.e., a complete description $(\vec{u}, \vec{v})$ of the values of both the exogenous and the endogenous variables. Formulas are evaluated with respect to tuples $(\mathcal{M}, \vec{u}, \vec{v}[n])$. Once the values of $(\vec{u}, \vec{v})$ are assigned, a computation $\mathcal{C}$ is unique, since every $\mathcal{F}_X(\vec{u}, \vec{v})$ will always return a value for X and this value will be unique. Then, even if a model is not recursive, i.e., some variables will change their values during the computation, we can still deal with it in a straightforward way. For some models, the computation $\mathcal{C}$ may never terminate, i.e., computation results in an infinite loop. Our proposed semantics can be used to reason about such models in the same way as discussed here. The only difference for this case is that the notation $(\mathcal{M}, \vec{u})$ abbreviating $(\mathcal{M}, \vec{u}[\infty])$ is no longer applicable: we always need to refer to the exact step of the computation to evaluate the truth of our formulas. Alternative way to deal with non-recursive models are so-called generalized structural equation models (GSEMs) [15]. But the dynamic interpretation of GSEMs remains an open problem.

Dynamic causal models can also be used to deal with a temporal dimension embedded in a causal model, e.g. time-indexed variables (see [22, 13, 4]) and causal reasoning in time series [23].

References

[1] Natasha Alechina, Joseph Y. Halpern, and Brian Logan, 'Causality, responsibility and blame in team plans', in *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems AAMAS 2017*, eds., S. Das, E. Durfee, K. Larson, and M. Winikoff, (5 2017).

[2] Vitaliy Batusov and Mikhail Soutchanski, 'Situation calculus semantics for actual causality', *Proceedings of the AAAI Conference on Artificial Intelligence*, **32**(1), (2018).

[3] Sander Beckers, 'The counterfactual NESS definition of causation', *Proceedings of the AAAI Conference on Artificial Intelligence*, **35**, 6210–6217, (5 2021).

[4] Sander Beckers and Joost Vennekens, 'A principled approach to defining actual causation', *Synthese*, **195**(2), 835–862, (Feb 2018).

[5] Alexander Bochman, 'Actual causality in a logical setting', in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 1730–1736. International Joint Conferences on Artificial Intelligence Organization, (7 2018).

[6] Marco Cadoli, 'The complexity of model checking for circumscriptive formulae', *Inf. Process. Lett.*, **44**(3), 113–118, (1992).

[7] Hana Chockler and Joseph Y. Halpern, 'Responsibility and blame: A structural-model approach', *Journal of Artificial Intelligence Research*, **22**, 93–115, (10 2004).

[8] N. Hall, 'Structural equations and causation', *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition*, **132**(1), 109–136, (2007).

[9] J. Y. Halpern, 'A modification of the Halpern-Pearl definition of causality', in *Proc. 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pp. 3022–3033, (2015).

[10] Joseph Y. Halpern, 'Axiomatizing causal reasoning', *Journal of Artificial Intelligence Research*, **12**, 317–337, (5 2000).

[11] Joseph Y. Halpern, *Actual Causality*, The MIT Press, 2016.

[12] Joseph Y. Halpern, 'Appropriate causal models and the stability of causation', *The Review of Symbolic Logic*, **9**, 76–102, (3 2016).

[13] Joseph Y. Halpern and Judea Pearl, 'Causes and explanations: A structural-model approach. part i: Causes', *The British Journal for the Philosophy of Science*, **56**(4), 843–887, (2005).

[14] 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).

[15] Joseph Y. Halpern and Spencer Peters, 'Reasoning about causal models with infinitely many variables', *Proceedings of the AAAI Conference on Artificial Intelligence*, **36**, 5668–5675, (6 2022).

[16] Mark Hopkins and Judea Pearl, 'Causality and Counterfactuals in the Situation Calculus', *Journal of Logic and Computation*, **17**(5), 939–953, (08 2007).

[17] Shakil Khan and Mikhail Soutchanski, 'Necessary and sufficient conditions for actual root causes', in *ECAI 2020 - 24th European Conference on Artificial Intelligence*, (11 2020).

[18] David Lewis, *Counterfactuals*, Blackwell, Oxford, 1973.

[19] David Lewis, 'Causation as influence', *Journal of Philosophy*, **XCVII**, 182–197, (2000).

[20] Emiliano Lorini, Dominique Longin, and Eunate Mayor, 'A logical analysis of responsibility attribution: emotions, individuals and collectives', *Journal of Logic and Computation*, **24**(6), 1313–1339, (12 2013).

[21] Pavel Naumov and Jia Tao, 'An epistemic logic of blameworthiness', *Artificial Intelligence*, **283**, 103269, (2020).

[22] Judea Pearl, *Causality: Models, Reasoning, and Inference*, Cambridge University Press, 2000.

[23] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf, *Elements of Causal Inference. Foundations and Learning Algorithms*, The MIT Press, Cambridge, Massachusetts, 2017.

[24] Camilo Sarmiento, Gauvain Bourgne, Katsumi Inoue, and Jean-Gabriel Ganascia, 'Action languages based actual causality in decision making contexts', in *PRIMA 2022: Principles and Practice of Multi-Agent Systems*, eds., Reyhan Aydoğan, Natalia Criado, Jérôme Lang, Victor Sanchez-Anguix, and Marc Serramia, pp. 243–259, Cham, (2023). Springer International Publishing.

[25] Richard W. Wright, 'The NESS account of natural causation: A response to criticisms', in *Critical Essays on "Causation and Responsibility"*, eds., Benedikt Kahmen and Markus Stepanians, pp. 13–66, Berlin, Boston, (2013). De Gruyter.

[26] Vahid Yazdanpanah, Mehdi Dastani, Wojciech Jamroga, Natasha Alechina, and Brian Logan, 'Strategic responsibility under imperfect information', in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '19, p. 592–600, Richland, SC, (2019). International Foundation for Autonomous Agents and Multiagent Systems.