Causal Analysis of Events Occurring in Trajectories of Dynamic Domains*

Michael Gelfond and Evgenii Balai ✉

Texas Tech University, Lubbock, Texas 79409, USA
{michael.gelfond,evgbalai}@ttu.edu.

Abstract. In this paper we define the notion of causes of events in trajectories of dynamic domains from the standpoint of an agent acting in this domain. We assume that the agent's knowledge about the domain is axiomatized in P-log with consistency restoring rules – a powerful knowledge representation language combining various forms of logical and probabilistic reasoning. The proposed model of causality is tested on a number of examples of causal domains frequently used in the literature.

Keywords: causality, answer set programming, P-log

1 Introduction

This paper contributes to a research program aimed at finding precise mathematical formalization of substantial parts of commonsense knowledge and developing commonsense reasoning methods in knowledge representation languages based on Answer Set Prolog (ASP). We concentrate on *causal reasoning*, which seems to be of vital importance for our understanding of the world. The nature of causality and various causal relations has, for a long time, been debated by philosophers, physicists, statisticians, researchers in AI, etc. For recent work see, for instance, [5,6,13,16]. But despite the amazing progress, we do not yet have fully adequate understanding of the subject. There are still different interpretations of the intuitive meaning of causality, answers provided to causal questions by various formalisms do not always match the intuition, and some "causal stories" simply cannot be expressed in existing languages. In our approach we address these problems by using rich knowledge representation language capable of expressing non-trivial causal relations as well as various forms of commonsense background knowledge. We opted for logic programming language P-log with consistency-restoring rules (cr-rules) [1,3,10]. It is an extension of ASP with well known methodology for representing defaults and their direct and indirect exceptions, recursive definitions, probability, direct and indirect effects of actions (including parallel and non-deterministic actions), time, etc. Its non-monotonic reasoning system combines standard ASP reasoning, abduction, and probabilistic computation. We are primarily interested in dynamic domains and, as in

many theories of action and change, view the agent's knowledge base as a description of possible trajectories of the domain. The events in these trajectories are caused by actions. This is different from a large body of work in which the agent's knowledge is represented by structural equations, causal logic or other formalisms emphasizing purely causal reasoning at the expense of background knowledge. Usually, but not always, these works provide counterfactual account of causality. There is, however, a number of recent approaches (see, for instance, [4,7,8,14]) which seem to share our philosophy. There are, however, many substantial differences related to the power of our KR-language and other factors. The multiplicity of interpretations of the word *cause* is partially addressed by concentrating on what is often referred to as actual causes. In our approach time (or at least ordering of events) is an integral part of this notion. We further deal with this problem by dividing causes of events into those which consist of deliberate actions, those which contain at least one accidental (but known) action, and those which include some exogenous actions not native to the agent's model of the world. The methods of testing our definitions and KR methodology are determined by our goal. We view our work as a step in an attempt to pass what J.Pearl calls *Mini-Turing Test* [17]: "*The idea is to take a simple story, encode it on a machine in some way, and then test to see if the machine can correctly answer causal questions that a human can answer.*" So, naturally, we use this to test accuracy of our modeling and relationship with other approaches. (Of course, only a few of such examples are presented in this paper.) To make sure that wrong answers to these questions are not caused by inadequate representation of the problem, we pay serious attention to developing KR methodology which properly combines causal and background knowledge about the domain. The paper is organized as follows. We assume some knowledge of P-log and define notions of *background theory* T and *scenario* S. The former contains general knowledge about the domain while the latter describes a particular story to be analyzed. Together they form the knowledge base of an agent, $T(S)$, referred to as *causal theory*. Next section contains definitions of three types of causes accompanied by some explanatory examples. This is continued by analyses of causal relations in several simple stories, followed by conclusion and future work.

2 Representing Agent's Knowledge

Knowledge of an agent will be represented by a P-log program tailored for reasoning about effects of actions and causes of events, Regular function symbols of a program will be partitioned into *fluent*, *action*, *static* and *auxiliary* and used to form fluent, action, static and auxiliary terms respectively. We assume that actions are Boolean. The last parameter of functions from the first two groups is of a special sort called *time-step* (usually represented by natural numbers); time-steps are not allowed in statics. Recall that P-log terms formed by *random* are of the form $random(m, f(\bar{x}), p)$. This expression can also be viewed as an atom (a shorthand for $random(m, f, p) = true$), which states that, as the result of a random experiment m which is either genuine or deliberately interfered with,

$f(\bar{x})$ should take the value from $\{Y : p(Y) \cap range(f)\}$. In addition, we require that for every time steps t_1 and t_2 occurring in m, f respectively, $t_2 > t_1$ if f is a fluent, and $t_2 \geq t_1$ if f is an action. Finally, both m and $random(m, f(\bar{x}), p)$ are viewed as action terms. Sometimes we say that $f(\bar{x}, t)$ is an instance of abstract fluent (action) $f(\bar{x})$ and that the value y of $f(\bar{x}, t)$ is the value of abstract fluent (action) $f(\bar{x})$ at time-step t. Atoms formed by action terms are called action atoms. Similarly for fluent and static atoms. We are especially interested in properties of *events* – statements describing occurrences of actions and values of fluents at given time-steps. More precisely an *action event* is a collection of action atoms. Similarly for *fluent events*. An *event* is a fluent event or an action event. The *causal theory* representing the agent's knowledge consists of a particular *story* (also called *domain scenario* or *recorded history of the domain*) to be analyzed and *background theory* representing agent's *general knowledge*.

Scenario is a recorded history of time-stepped observations and deliberate (intended) actions which happened in the domain up to the current time-step. The initial time-step of a scenario is usually assumed to be 0. Observations are expressions of the form $obs(atom)$, actions are of the form $\mathbf{do}(a(\bar{x}, t) = y)$; $\mathbf{do}(a(\bar{x}, t) = true)$ indicates a deliberate execution of action $a(\bar{x}, t)$; $\mathbf{do}(a(\bar{x}, t) = false)$ – a deliberate refusal of such execution. Another form of **do**-operator is similar to *do*-operator of the original P-log [3] and Pearl's causal networks [16]. If $random(m, e(\bar{x}, t), p)$ is a random experiment and y is a possible value of $e(\bar{x}, t)$ then $\mathbf{do}(m, y)$ represents an intervention into random experiment m which forces $e(\bar{x}, t)$ to take value y. If the value y of $e(\bar{x}, t)$ is simply observed this is recorded as $obs(e(\bar{x}, t) = y)$. Whenever possible, we omit **do** from the description of actions in our scenarios.

Background theory is a P-log program T which contains no deliberate actions and observations and whose regular part (obtained from T by removing cr-rules) is consistent. T contains a set of *causal mechanisms* or *causal laws* of the form

$$m : \ a(\bar{x}) = y \leftarrow body, \ \text{not } ab(m), \ \text{not } interfere(a(\bar{x}), y)^{1} \qquad (*)$$

if a is an action and

$$m : \ a(\bar{x}) = y \leftarrow body, \ \text{not } ab(m)$$

if a is a fluent. In both rules *body* is non-empty and contains no default negation, m is the mechanism's name, ab and $interfere$ are auxiliary functions; $interfere(a(\bar{x}), y)$ holds if action $a(\bar{x})$ is deliberately assigned value different from y; $ab(m)$ captures indirect exceptions to m. Each causal mechanism is accompanied by *Contingency Axiom*

$$ab(m) \stackrel{+}{\leftarrow} body$$

[1] If $a(\bar{x})$ is formed by $random(r, f(\bar{u}), p)$, then m is omitted and the causal mechanism is named r. Random experiments are normally named by action terms.

If the causal mechanism is not defeasible, then the contingency axiom and the corresponding "*not* $ab(m)$" from the body of a causal law can be omitted. Intuitively, the first two rules say that normally *body* is a sufficient cause for *head*. The guard $interfere(a(\bar{x}), y)$ present in rule ($*$) allows deliberate actions to defeat triggering defaults. We will use shorthand $interfere(a(\bar{x}))$ to denote $interfere(a(\bar{x}), true)$. The Contingency Axiom for n allows causal mechanism m to be defeated by observations. It is a *consistency-restoring rule* of a version of ASP called *CR-Prolog* [2]. It says that "causal mechanisms m may be disabled, but such a possibility is very rare and, whenever possible, should be ignored". For more details, see [9]. In addition, a rule r of a causal theory must satisfy the following conditions:

- If a time-step t occurs in r then some time-step occurs in $head(r)$.
- If t occurs in $head(r)$ then (a) if $head(r)$ is a fluent atom then time-steps of fluents and actions in $body(r)$ do not exceed t and $t-1$ respectively, (b) if $head(r)$ is an action atom then no time-step in $body(r)$ exceeds t.

Let us illustrate the notion of causal theory by formalizing two informal examples frequently used in the literature on causation.

Example 1 (Firing Squad). A certain chain of events is required for a lawful execution of a prisoner. First, the court must order the execution. The order goes to a captain, who signals the soldiers on the firing squad (denoted by a and b) to shoot. We'll assume that they are obedient and expert marksmen, so they only shoot on command, and if either of them shoots, the prisoner dies.

Background theory FS for this example contains abstract actions *court_order*, *captain_order*, $shoot(a)$ and $shoot(b)$, inertial abstract fluent *dead* and standard auxiliary symbols ab and $interfere$. FS consists of causal mechanisms:

$$[m_1(T)] : captain\_order(T+1) \leftarrow court\_order(T), \text{not } ab(m_1(T)), \text{not } interfere(captain\_order(T+1)) \tag{1a}$$

which is a defeasible version of dynamic causal law used in actions languages. Two other rules:

$$[m_2(G,T)] : shoot(G,T+1) \leftarrow captain\_order(T), \text{not } ab(m_2(G,T)), \text{not } interfere(shoot(G,T+1)) \tag{1b}$$

$$[m_3(G,T)] : dead(T+1) \leftarrow shoot(G,T), \text{not } ab(m_3(G,T)) \tag{1c}$$

are defeasible triggers. We also have the contingency axioms:

$$ab(m_1(T)) \overset{+}{\leftarrow} court\_order(T) \tag{2a}$$

$$ab(m_2(G,T)) \overset{+}{\leftarrow} captain\_order(T) \tag{2b}$$

$$ab(m_3(G,T)) \overset{+}{\leftarrow} shoot(G,T) \tag{2c}$$

and the closed world assumptions (CWA) for actions:

$$\neg shoot(G,T) \leftarrow \text{ not } shoot(G,T) \tag{3a}$$
$$\neg captain\_order(T) \leftarrow \text{ not } captain\_order(T) \tag{3b}$$
$$\neg court\_order(T) \leftarrow \text{ not } court\_order(T) \tag{3c}$$

The CWA for deliberate action *court_order* will be accompanied by an indirect exception:

$$court\_order(T) \stackrel{+}{\leftarrow} \tag{4}$$

We also need inertia axioms for *dead*:

$$\neg dead(T+1) \leftarrow \neg dead(T), \text{not } dead(T+1) \tag{5a}$$
$$dead(T+1) \leftarrow dead(T), \text{not } \neg dead(T+1) \tag{5b}$$

Despite the fact that the story insists that the guards *only* shoot on command, the corresponding causal law is defeasible. This is essential since we would like to consider scenarios in which guards may refuse to follow the orders or simply fail to do so by unspecified reasons. Similarly for other causal mechanisms.

Example 2 (Flipping a Coin). Theory TC has "transient" fluent *agreed_to_play* (players agreed to start the game), and a "transient" fluent h (the coin landed heads). Transient fluents are partial functions which do not satisfy inertia. TC also contains action $flip$ (flip the coin); h is defined at time-steps immediately following $flip$. Causal mechanism

$$random(flip(T), h(T+1)) \leftarrow agreed\_to\_play(T), \text{not } ab(m(T)), \\ \text{not } interfere(random(flip(T), h(T+1)))$$

states that *agreed_to_play* triggers a random experiment $flip$ which ends in heads or in tails. We also need the contingency axiom and CWA for actions.

Agent's knowledge base and its models: *A scenario S is* encoded in P-log *as follows: $obs(A)$, where A is time-stepped by* 0 *is encoded by A; if the time-step of A is greater than* 0 *then $obs(A)$ is encoded by a constraint: "$\leftarrow$ not A". Do-statements of S remain unchanged. We denote this encoding by $enc(S)$.* Agent's knowledge base is given by causal theory

$$T(S) =_{def} T \cup enc(S) \cup DO$$

where S is a scenario of T and DO is a collection of axioms enforcing the intuitive meaning of **do**:

- for every $\mathbf{do}(a(\bar{x},t) = y)$ from S axioms:

$$a(\bar{x},t) = y \leftarrow \mathbf{do}(a(\bar{x},t) = y)$$
$$interfere(a(\bar{x},t), Y) \leftarrow Y \neq y, \mathbf{do}(a(\bar{x},t) = y)$$

  where the former axiom connects **do** with an actual occurrence of an action, and the latter allows a deliberate action interfere with a defeasible trigger assigning values to action $a(\bar{x})$;

– for every $\mathbf{do}(m, y)$ from S, where m is the name of random experiment $random(m, a(\bar{x}, t), p)$, axioms

$$do(m, a(\bar{x}, t), y) \leftarrow \mathbf{do}(m, y)$$

$$interfere(a(\bar{x}, t), Y) \leftarrow Y \neq y, \mathbf{do}(m, y)$$

where the former axioms guarantees that on random experiments **do** coincides with the original *do* of P-log and the latter defines interference with random experiments.

We only consider S for which $T(S)$ is a coherent P-log program in which multiplicity of models can only be a result of general axiom (19) from [1] for random. As any such program, $T(S)$ comes with the definition of its possible worlds and probability function. A possible world W of $T(S)$ can be viewed as a possible *trajectory* of a dynamic system associated with the program and written as

$$W = \langle \sigma(t_f), \alpha(t_f), \ldots, \sigma(i), \alpha(i), \ldots \alpha(t_l - 1), \sigma(t_l) \rangle$$

where $\alpha(i)$ is the set of all action events from W time-stepped by i and $\sigma(i)$ is the set of all fluent atoms of W time-stepped by i and statics from W. Note that though all actions from $\alpha(i)$ start at i, their effects may manifest themselves at different time-steps.

Definition 1 (Model). A *model* of scenario S of T is a possible world of $T(S)$.

Let us demonstrate this notion by going back to the Firing Squad example.

Example 3 (Firing Squad: models). Let us fix FS from Example 1. Then, in the model of the scenario $S_0 = \langle obs(\neg dead(0)) \rangle$ the prisoner is alive at every time step of the model. There are no actions. Scenario
$S_1 = \langle obs(\neg dead(0)), court\_order(0) \rangle$ has one model, M[2]:

$\neg dead(0), court\_order(0), \neg dead(1), captain\_order(1)$
$\neg dead(2), shoot(a, 2), shoot(b, 2), dead(3), interfere(court\_order(0), false)$.

When displaying a model we usually omit negations of actions derived by the CWA and the **do** statements.
Scenario $S_2 = \langle obs(\neg dead(0)), court\_order(0), \neg shoot(a, 2), \neg shoot(b, 2) \rangle$ is more interesting. Deliberate actions $\neg shoot(a, 2), \neg shoot(b, 2)$ from S together with axioms from DO cancel axiom $m_2(G, 2)$. The only model M of S_2 contains

$\neg dead(0), court\_order(0), \neg dead(1), captain\_order(1), \neg shoot(a, 2), \neg shoot(b, 2),$
$\neg dead(2), interfere(shoot(a, 2)), interfere(shoot(b, 2)), \neg dead(3)$

(In what follows we omit atoms formed by $interfere$ in the models). Next consider scenario $S_3 = \langle obs(\neg dead(0)), court\_order(0), obs(\neg dead(3)) \rangle$ with a non-initial observation which contradicts the effects of our causal mechanisms. The

[2] The model can be computed using our prototype P-log solver. For more details, refer to https://github.com/iensen/plog2.0/tree/master/plogapp/tests/causality.

contradiction can be resolved by assuming that the captain was not able to follow the court order or that his order could not have been executed by the guards. This is done by the Contingency Axioms. In CR-Prolog the contradiction can be avoided by finding *abductive support* - a minimal collection R of cr-rules whose application restores consistency of the program, i.e., the regular part Π^r of program Π together with the result, $\alpha(R)$, of replacing $\stackrel{+}{\leftarrow}$ in rules from R by $\leftarrow$ has an answer set. We define $\Pi^R =_{def} \Pi^r \cup \alpha(R)$. M is a model of Π if it is a model of Π^R for some abductive support R of Π. In this case we say that R *generates* M. There are different ways to compare abductive supports. In what follows we mainly assume that support A is better than B if $A \subset B$. In our case there are two ways to resolve the contradiction. One abductive support is R_1 consisting of contingency axioms for $m_2(a,1)$ and $m_2(b,1)$. The axioms derive $ab(m_2(a,1))$ and $ab(m_2(b,1))$ and hence disable $m_2(a,1)$ and $m_2(b,1)$. By inertia, FS^{R_1} will conclude $\neg dead(3)$ which leads to the first model M_1 of FS:

$ab(m_2(a,1)), ab(m_2(b,1)), \neg dead(0), court\_order(0), \neg dead(1), captain\_order(1)$
$\neg dead(2), \neg shoot(a,2), \neg shoot(b,2), \neg dead(3)$

Contradiction can also be avoided by abductive support R_2 consisting of the contingency axiom for $m_1(1)$. In FS^{R_2} causal connection between court and captain orders will be disabled and, by CWA, no order will be given by the captain. This will lead to the second model M_2 of FS:

$ab(m_1(0)), \neg dead(0), court\_order(0), \neg dead(1), \neg captain\_order(1)$
$\neg dead(2), \neg shoot(a,2), \neg shoot(b,2), \neg dead(3)$

In scenario $S_4 = \langle obs(\neg dead(0)), obs(dead(3))\rangle$ the only way to satisfy the last observation is to use rule (4) of FS and assume $court\_order(0)$.The resulting model, M, consists of all atoms from the model of S_1 except for those formed by **do** and $interfere$.

Example 4 (Flipping a Coin (Models)). It is easy to see that all models of scenario $S_0 = \langle agreed\_to\_play(0)\rangle$ contain $random(flip(0), h(1))$ (which we shorten to $flip(0)$). The experiment generates two possible outcomes $h(1)$ and $\neg h(1)$. Thus, S_0 has two models: $M_1 = \{agreed\_to\_play(0), flip(0), h(1)\}$ and $M_2 = \{agreed\_to\_play(0), flip(0), \neg h(1)\}$. Scenario $S_1 = \langle agreed\_to\_play(0), obs(h(1))\rangle$ has only one model M_1. Finally, $S_2 = \langle agreed\_to\_play(0), \mathbf{do}(flip(0), true)\rangle$ has only one model $M_1 \cup \{do(flip(0), h(1), true)\}$.

3 The Definitions

Our framework is based on the following **assumptions**: *An agent is supplied with a (fixed) background theory T and a scenario S with the last time step n, which consists of observations and the complete collection of deliberate actions which occurred in the domain up to that time. In addition, we assume that every uninterrupted random experiment of a scenario is immediately followed by the observation of its outcome.* Scenarios which do not satisfy these assumptions

will be called *illegal*. In what follows we introduce three different types of cause: *deliberate*, *accidental* and *exogenous*. The best explanation of an event is given by finding its deliberate cause. If this is impossible we attempt to find causes which include accidental (not deliberate) actions. As the last resort we allow causes containing unknown exogenous actions.

Deliberate Cause: Our definition of a deliberate cause can be viewed as a formalization of the following intuition:

> "*A cause of $e(\bar{x}, k) = y$ is a deliberate action event α which initiates a chain of events bringing about $e(\bar{x}, k) = y$. Moreover, α must be in some sense minimal, i.e. no parts of α can be removed without loss of causal information about $e(\bar{x}, k) = y$.*"

An expression "chain of events" from our informal description will be modeled by notion of proof [3] M be a set of ground literals, and $M^- = \{\text{not } l : l \text{ is not satisfied by } M\}$.

Definition 2 (Proof).

- A sequence $P = \langle r_0, l_0, r_1, l_1 \ldots, r_n, l_n \rangle$ where ls are literals from M and rs are rules or names of random experiments of program Π is called a *proof of l_n in M from Π ($M, \Pi \vdash l_n$)* if
  - For every i, $body(r_i)$ is satisfied by $\{l_0, \ldots, l_{i-1}\} \cup M^-$,
  - l_i is the head of r_i or
  - l_i is $e(k) = y$ and l_{i-1} is $random(n, e(k), p)$ representing a non-interrupted random experiment (i.e., there is no y such that $do(n, e(k) = y)$ is in S), r_i is n, and $p(y) \in \{l_0, \ldots, l_{i-1}\}$
  - No proper sub-sequence of P satisfies the above conditions.

  If M is a possible world of Π we simply say that P is a proof of l_n in M;
- A scenario S of background theory T *derives* event $e(k) = y$ in M (or simply $T(S)$ *derives* $e(k) = y$) if there is a proof of $e(k) = y$ in M from $T(S)$; S *derives* $e(k) = y$ if S derives $e(k) = y$ in every model of S.

The idea of a deliberate (or intentional) action is formalized as follows.

Definition 3 (Deliberate Action). Let S' be the result of removing action event $a(i) = y$ from a scenario S of T. We say that $a(i) = y$ is *deliberate* in $T(S)$ if no model of S' contains $a(i) = y$.

To define a cause of an event $e(k) = y$ in a scenario S of T we introduce a notion of the event's *inflection point - a time-step $i \leq k$ such that $i-1$ is the last time-step of S in which the value of $e(k)$ is not predicted to be y*. To make this

[3] A similar notion of a proof is given in [7]. Important and substantial differences include special treatment for P-log literals formed by *do* and *random*, literals with default negation, and cr-rules.

precise we need some notation: Let $S[i]$ consists of observations of S made at points not exceeding i and all actions S which occurred at steps not exceeding $i-1$, i.e. $S[i] =_{def} \{obs(f(j) = y) \in S : j \leq i\} \cup \{a(j) = y \in S : j < i\}$.

Definition 4 (Inflection Point). Step i is the *inflection point* for $e(k) = y$ in $T(S)$ if (a) $T(S[i-1])$ does not derive $e(k) = y$ and (b) for every $j \in [i, k]$, $T(S[j])$ derives $e(k) = y$.

Definition 5 (Deliberate Cause). Let S be a scenario of T, $obs(e(k) = y) \in S$, M be a model of S generated by a (possibly empty) abductive support R, and i be the inflection point of $e(k) = y$ in $T^R(S)$. A non-empty set α of actions from S is called *a cause* of $e(k) = y$ (with respect to $T(S)$) in M if

(a) $T^R(S[i-1] \cup \alpha)$ derives $e(k) = y$ in M,
(b) for every $\beta \subset \alpha$ there is a proof of $e(k) = y$ from $T^R(S[i-1] \cup \alpha)$ in M which is not a proof of $e(k) = y$ from $T^R(S[i-1] \cup \beta)$ in M.

We say that α *is a possible cause of* $e(k) = y$ *in* S if it is a cause of $e(k) = y$ in some model of S; α *causes* $e(k) = y$ *in* S if it causes $e(k) = y$ in all models of S.

Consider scenario S_1 from Example 3. Clearly, the inflection point for $dead(3)$ in $S_1 \cup \{obs(dead(3))\}$ is 1. It's cause is $court\_order(0)$ with the proof consisting of applications of causal mechanisms of FS. The same is true for $dead(4)$. Now consider a theory T with action a, inertial fluent f and causal law $f(T) \leftarrow a(T-1)$. Clearly $M = \{\neg f(0), m(0), \neg f(1), a(1), f(2)\}$ is the only model of scenario $S = \langle obs(\neg f(0)), random(m(0), a(1)), obs(a(1))\rangle$. The only deliberate action $random(m(0), a(1))$ of S is the only cause of $f(2)$ in M. We often read this as: $f(2)$ *is caused by the outcome* $a(1)$ *of random experiment* $m(0)$. This reading will be used throughout the paper.

Causes Containing Accidental Actions. To see the need for causes with accidental actions consider scenario $S_4 = \{obs(\neg dead(0)), obs(dead(3))\}$ from Example 3. Since S_4 contains no deliberate actions, $dead(3)$ has no deliberate cause. One can, however, argue that action $court\_order(0)$ should be a cause of $dead(3)$ in the unique model M of S_4 even though it is not deliberate. This intuition can be justified by the following informal principle:

> *Execution of a non-deliberate (accidental) action could be a part of an event's cause if it's deliberate execution could.*

One should, however, be careful in formalizing this intuition. An attempt to do that by simply adding the action to the original scenario does not work. Since $court\_order$ is derived by FS in the model of S_4, it is not deliberate in scenario $S_5 = S_4 \cup \{court\_order(0)\}$ and hence such a scenario is not legal.

One way to avoid the difficulty is to remove from FS rule (4) and consider S_5 to be a scenario of a new theory FS^*; $court\_order(0)$ is deliberate in $FS^*(S_5)$. After this "surgery", somewhat reminiscent of Pearl's surgery on causal networks,

$dead(3)$ would indeed be caused by $court\_order(0)$. Another small operation is required in scenarios containing random events. The new scenario should also be extended by the observation of the outcome of this experiment taken from the corresponding model. These observations lead to the following definition. Let M be a model of a scenario S of T, E be an action atom accidental in $T(S)$, $c(E, M)$ be E if E is regular, or $random(m, f)$ followed by observation of outcome $f = y$ of m in M if E is $random(m, f)$. If α is an action event then $c(\alpha, M) =_{def} \{c(E, M) : E \in \alpha\}$. We say that a rule r *generates* the value of term $a(i)$ if $head(r)$ is $a(i) = y$ for some y or is of the form $random(m, a(i), p)$. By $surg(T, \alpha)$ we denote a theory obtained from T as follows. For every $a(i)$ such that α contains $random(m, a(i), p)$ or $a(i) = y$ for some y remove from T every rule generating the value of $a(i)$.

Definition 6 (Causes Containing Accidental Actions). Given a model M of scenario S of T and action event $\alpha \subset M$, α *is an accidental cause of* $e(k) = y$ *in* M with respect to $T(S)$ if there is an action event $\beta \subset M$ such that

- M is a model of scenario $S^* =_{def} S \cup c(\beta, M)$ of $T_1 = surg(T, \beta)$ (Note that, by construction, all actions of S^* are deliberate), and
- α is a deliberate cause of $e(k) = y$ in M with respect to $T_1(S^*)$.

Now let us go back to scenario S_4 from Example 3 and show that the cause of $dead(3)$ in model M of S_4 is $court\_order(0)$. The model contains $court\_order(0)$ included in it by cr-rule (4) of theory FS. Let $\alpha = \beta = \{court\_order(0)\}$. It is easy to check that $court\_order(0)$ is a deliberate cause of $dead(3)$ in scenario $S_4^* = S_4 \cup \{court\_order(0)\}$ of $surg(FS, \beta)$ obtained from FS by removing rule (4). Note, that $captain\_order(1)$ will not be a cause of $dead(3)$ in M since M is not a model of $S_4 \cup \{captain\_order(1)\}$. Finally, consider scenario S_3 from Example 3. As was shown there it has two models: M_1 in which guards refuse to shoot and M_2 in which captain does not follow his order. One can check that refusal to shoot is the cause of $\neg dead(3)$ in M_1. In M_2 the cause is $\neg captain\_order(1)$.

To see how the definition works for random events consider the unique model M_1 of scenario $S_1 = \langle obs(agreed\_to\_play(0)), obs(h(1))\rangle$ from Example 4. The inflection point for $h(1)$ in S_1 is 1. The scenario contains no deliberate actions and hence $h(1)$ has no deliberate cause. It is, however, not difficult to show that, by Definition 6, $h(1)$ is caused by a random experiment $flip(1)$. In the model M_2 of S_2, however, $h(1)$ is the result of deliberate intervening action $do(flip(1), true)$.

The difference between Pearl's actions and observations is important not only for computing probability of events but also for discovering their causes. So far, we have been looking for causes of events occurring in a model M among actions from M. Now we allow to search for causes among a specific type of actions not present in M.

Causes Containing Exogenous Actions.

If an event has no cause consisting of deliberate and accidental actions then it may be useful to admit causes containing unknown, exogenous actions responsible for bringing about events in the initial state of the scenario and/or abnormality relations in causal mechanisms of the theory.

The definition is omitted due to space limitation. We simply illustrate the intended behavior. Consider scenario $S_0 = \langle obs(\neg dead(0)), obs(\neg dead(3))\rangle$ of FS from Example 3. The cause of $\neg dead(3)$ in the only model M_0 of S_0 is the exogenous action which brought about $\neg dead(0)$. The cause of $\neg f(1)$ in the only model of scenario $S = \langle obs(\neg f(0)), a(0), obs(\neg f(1))\rangle$ of theory

$$m(T) : f(T) \leftarrow a(T-1), \text{not } ab(m(T))$$

is the exogenous action which brought about $\neg f(0)$.

4 Examples

This example, often used to highlight difficulties with counterfactual approach to causation, (see, for instance, [12]) deals with so called "late preemption".

Example 5 (Breaking the Bottle). Suzy and Billy both throw rocks at a bottle. Suzy's rock arrives first and shatters the bottle. Billy's arrives second and so does not shatter the bottle. Both throws are accurate: Billy's would have shattered the bottle if Suzy's had not.
The background theory, TS of the story, with steps ranging from 0 to 2 contains actions $throw(suzy)$ and $throw(bill)$ with static attributes $duration(suzy) = 1$ and $duration(bill) = 2$ for durations of agents' throws, causal mechanism $m(A, T_1)$

$$shattered(T_2) \leftarrow throw(A, T_1), duration(A) = D, T_2 = T_1 + D, \\ \neg shattered(T_2 - 1), \text{not } ab(m(A, T_1))$$

determining the effect of throwing, contingency axiom for this rule and the inertia axiom for *shattered*. Consider scenario
$S_0 = \langle obs(\neg shattered(0)), throw(suzy, 0), throw(bill, 0)\rangle$. The only model M_0 of S_0 is: $\{\neg shattered(0), throw(suzy, 0), throw(bill, 0), shattered(1), shattered(2)\}$.

Step 1 is the inflection point for $shattered(1)$ and $shattered(2)$ in M_0 and their only cause is $throw(suzy, 0)$. Next consider $S_1 = S_0 \cup \{obs(\neg shattered(1))\}$ and its unique model

$$M_1 = M_0 \cup \{\neg shattered(1), ab(m(suzy, 0))\}) \setminus \{shattered(1)\}$$

Step 1 is still the inflection point for $shattered(2)$ in M_1, which is now caused by $throw(bill, 0)$; $\neg shattered(1) \in M_1$ is caused by an exogenous action which brought about $\neg shattered(0)$. Next consider
$S_2 = \langle obs(\neg shattered(0)), throw(bill, 0), throw(suzy, 1)\rangle$. It is easy to check that in its only model $shattered(2)$ has two causes: $throw(bill, 0)$ and $throw(suzy, 1)$.
Here is another example, taken from [11].

Example 6 (Hall's Neuron Net). Consider a neuron net from Figure 1.

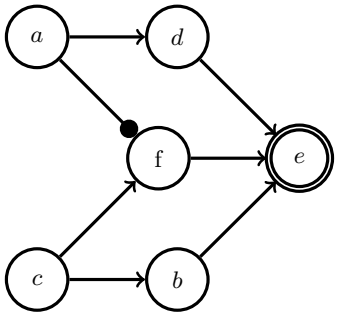

Fig. 1.

If a link from neuron n_1 to neuron n_2 is ended by an arrow, then n_1 stimulates n_2; if it is ended by a bullet, then n_1 inhibits n_2; e is a "stubborn" neuron, requiring two stimulatory signals to fire. For other neurons one stimulating signal is sufficient.

A background theory NN for this example will have sorts for neurons, an action $stim(S)$ which stimulate neurons from set S, Boolean fluents *stimulated* and *inhibited*, and statics *link* and *stubborn*. The net will be represented by a collection of atoms $link(a, d, stm)$, $link(a, f, inh)$, etc., where $link(X, Y, stm)$ / $link(X, Y, inh)$ indicates that X stimulates/inhibits Y, and facts $stubborn(e) \cup \{\neg stubborn(N) : N \neq e\}$. We will need two time steps: 0 and 1 with 0 being used for the execution of actions and 1 for their effects, and two inputs of action *stim*: s_1 and s_2 defined by statics: $member(c, s_1)$, $member(c, s_2)$, $member(a, s_2)$.

The causal mechanisms of NN are

$$[m_0(X,S)]: \ stimulated(X,1) \leftarrow stim(S,0), member(X,S)$$

$$[m_1(X,Y)]: stimulated(Y,1) \leftarrow \neg stubborn(Y), \neg inhibited(Y,1), link(X,Y,stm), stimulated(X,1)$$

$$[m_2(Y)]: stimulated(Y,1) \leftarrow stubborn(Y), \neg inhibited(Y,1), card\{X : link(X,Y,stm), stimulated(X,1)\} > 1$$

$$[m_3(X,Y)] \ : \ inhibited(Y,1) \leftarrow link(X,Y,inh), stimulated(X,1)$$

We assume that all neuron directly stimulated by *stim* are included in its parameter S, which eliminates the possibility of parallel *stim* actions, i.e. we have

$$\neg stim(S_1, I) \leftarrow stim(S_2, I), S_1 \neq S_2$$

Finally, we need the inertia axiom for the fluents, and CWA with indirect exceptions for action *stim*: $\neg stim(S, 0) \leftarrow$ not $stim(S, 0)$ and $stim(S, 0) \stackrel{+}{\leftarrow}$. Let us

consider NN together with a scenario $S_0 = init \cup \{obs(stimulated(e,1))\}$ where $init = \{obs(\neg stimulated(X,0)), obs(\neg inhibited(X,0)) : neuron(X)\}$. The regular part of $NN(S_0)$ is inconsistent. There are two ways to restore consistency using the cr-rule, and therefore two models of S_0: M_1 containing $stim(s_1,0)$ in which e is stimulated via neurons c, f, and b and M_2 which contains $stim(s_2,0)$. In M_2 neuron f is inhibited and e is stimulated via neurons a, c, d, and b. Clearly, in the first model $stimulated(e,1)$ is caused by $stim(s_1,0)$ and in the second by $stim(s_2,0)$. Hence in S_0, $stimulated(e,1)$ has two possible causes. One can argue that $stim(s_2,0)$ shall not be an actual cause of $stimulated(e,1)$ in S_0 since there is a better "minimally sufficient" candidate $stim(s_1,0)$. Indeed, since $s_1 \subset s_2$, action $stim(s_1)$ is simpler than $stim(s_2)$ but we believe that this does not preclude $stim(s_2,0)$ from being viewed as a valid possible cause of $stimulated(e,1)$ in S_0. This seems to agree with the Hall's view.

Example 7 (Adopted from [15] to our language).
Consider background theory with actions e_1, e_2, inertial fluents d_1, d_2, d_3, l, causal mechanisms:

$$[m1] \ : \ d_1(1) \leftarrow e_1(0) \quad [m2] \ : \ d_2(1) \leftarrow e_1(0) \quad [m3] \ : \ d_3(1) \leftarrow e_2(0)$$

rules:

$$l(1) \leftarrow d_1(1) \quad l(1) \leftarrow d_2(1), d_3(1)$$

and inertia axioms for fluents. Consider scenario:

$$S = \{e_1(0), e_2(0), obs(\neg d_1(0)), obs(\neg d_2(0)), obs(\neg d_3(0)), obs(\neg l(0))\}$$

Our definition agrees with the author of [15] and produce two causes of $l(1)$: $C_1 = \{e_1(0)\}$ and $C_2 = \{e_1(0), e_2(0)\}$. However, consider now background theory T_2 obtained from T by adding CR-rules $e_1(0) \stackrel{+}{\leftarrow}$ and $e_2(0) \stackrel{+}{\leftarrow}$ and a new scenario $S_2 = \{obs(\neg d_1(0)), obs(\neg d_2(0)), obs(\neg d_3(0)), obs(\neg l(0)), obs(l(1))\}$. Intuitively, we would expect C_1 and C_2 to also be the causes of $l(1)$ in $T_2(S_2)$. However, the subset-minimal preference relation does not produce this result. In the extended version we define a new preference relation which minimizes the number of applied cr-rules not relevant to actions and observations from the scenario. It produces desired results for this and other examples.

5 Conclusion

The paper outlines a new approach to analyzing causes of events in trajectories of dynamic domains. The fuller version, available from https://www.depts.ttu.edu/cs/research/krlab/documents/causal2020_extended.pdf, contains more examples and a more detailed comparison with other approaches. In future, we plan to conduct some mathematical investigation of causal theories. Even though in some respects our formalism is a more powerful modeling tool than that of structural equations and graphical models advocated by Pearl and many others it remains to be seen if it can also expand their computational power.

References

1. Balai, E., Gelfond, M., Zhang, Y.: P-log: refinement and a new coherency condition. Annals of Mathematics and Artificial Intelligence **86**(1-3), 149–192 (2019)
2. Balduccini, M., Gelfond, M.: Logic programs with consistency-restoring rules. In: International Symposium on Logical Formalization of Commonsense Reasoning, AAAI 2003 Spring Symposium Series. vol. 102 (2003)
3. Baral, C., Gelfond, M., Rushton, J.N.: Probabilistic reasoning with answer sets. Theory and Practice of Logic Programming **9**(1), 57–144 (2009)
4. Batusov, V., Soutchanski, M.: Situation calculus semantics for actual causality. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
5. Beckers, S., Vennekens, J.: Counterfactual dependency and actual causation in cp-logic and structural models: a comparison. In: Proceedings of the Sixth Starting AI Researchers Symposium. vol. 241, pp. 35–46. IOS Press (2012)
6. Bochman, A.: Actual causality in a logical setting. In: IJCAI. pp. 1730–1736 (2018)
7. Cabalar, P., Fandinno, J., Fink, M., Leuschel, M., Schrijvers, T.: Causal graph justifications of logic programs. Theory and Practice of Logic Programming **14**(4-5), 603 (2014)
8. Fandinno, J.: Towards deriving conclusions from cause-effect relations. Fundamenta Informaticae **147**(1), 93–131 (2016)
9. Gelfond, M., Kahl, Y.: Knowledge representation, reasoning, and the design of intelligent agents: The answer-set programming approach. Cambridge University Press (2014)
10. Gelfond, M., Rushton, N.: Causal and probabilistic reasoning in P-log. Heuristics, probabilities and causality. A tribute to Judea Pearl pp. 337–359 (2010)
11. Hall, N.: Two concepts of causation, pp. 225–276. MIT Press (2004)
12. Halpern, J.Y., Hitchcock, C.: Graded causation and defaults. The British Journal for the Philosophy of Science **66**(2), 413–457 (2014)
13. Halpern, J.: Actual Causality. MIT Press (2016)
14. LeBlanc, E., Baldiccini, M., Vennekens, J.: Explaining actual causation via reasoning about actions and change. In: JELIA (2019)
15. Leblanc, E.C.: Explaining Actual Causation via Reasoning About Actions and Change. Ph.D. thesis, Drexel University (2019)
16. Pearl, J.: Causality. Cambridge university press (2009)
17. Pearl, J., Mackenzie, D.: The Book of Why. Basic Books (2018)