Encoding probabilistic causal model in probabilistic action language

Nam Tran and **Chitta Baral**
Department of Computer Science and Engineering
Arizona State University, Tempe, AZ 85281
{chitta,namtran}@asu.edu

Abstract

Pearl's probabilistic causal model has been used in many domains to reason about causality. Pearl's treatment of actions is very different from the way actions are represented explicitly in action languages. In this paper we show how to encode Pearl's probabilistic causal model in the action language PAL thus relating this two distinct approaches to reasoning about actions.

Introduction and motivation

Normally an action when executed in a world changes the state of the world. Reasoning about actions is important in several 'intelligent' tasks such as planning, hypothetical reasoning, control generation and verification (for dynamical systems), and diagnosis. Often the effect of an action on the world is not deterministic but rather has an uncertainty associated with it. In recent years there have been several approaches to represent and reason with such actions. The first type of approaches include probabilistic generalization of formalisms for reasoning about actions; for example, planning (Kushmerick, Hanks, & Weld 1995; Littman 1997), situation calculus (Bacchus, Halpern, & Levesque 1999; Poole 1998; Reiter 2001; Mateus, Pacheco, & Pinto 2002), and action languages (Baral, Tran, & Le 2002; Eiter & Lukasiewicz 2003). The second type of approaches are based on frameworks of reasoning under uncertainty such as independence choice logic (Poole 1997; 1998) and probabilistic causal model (Pearl 1995; 1999; 2000).

In all these proposals, except in Pearl (1999; 2000), actions are explicitly defined (with names) and their effects on the world are described by various means. In Pearl (1999; 2000) the dynamics of the world is described through relationships between variables (which denote properties of objects in the world) that are expressed through functional relationships between them. Furthermore, probabilities are associated with a subset of variables called background (or exogenous) variables. Together they are referred to as *probabilistic causal models* (PCMs). The effect of actions are then formulated as "local surgery"on these models (Pearl 1995).

In this paper our goal is to study the relationship between reasoning about actions in PCMs and similar reasoning in the action description language PAL (Baral, Tran, & Le 2002) as a representative of the approaches where actions are named and have effects associated with it. The motivation behind studying this relationship is to objectively compare the expressiveness of these two formalisms vis-a-vis each other. We select PCM to study as it is the most successful representative of causal reasoning formalisms (Pearl 2000). We pick PAL as a representative of the high level action description languages (Gelfond & Lifschitz 1993; McCain & Turner 1995; Gelfond & Lifschitz 1998), for this is the most similar to PCM in the way it handles uncertainty among the action languages that have actions named distinctly. Moreover, PAL is inspired by PCM [1], so it is natural to question how they relate to each other. In fact, the formal analysis of this relation has provided us a better understanding of the two frameworks in term of their advantages and limitations.

The rest of this paper is organized as follows. First, we briefly recall the language PAL and PCM. We present an encoding of PCM in PAL together with a correctness result. We also provide the intuition of the encoding as well as of the result and its proof through a detailed example. Finally, we conclude with discussion on related and future works.

The language PAL: a brief overview

The alphabet of the language PAL (Baral, Tran, & Le 2002) consists of four non-empty disjoint sets $\mathbf{F}$, $\mathbf{U}_I$, $\mathbf{U}_N$, $\mathbf{A}$. The sets respectively contain fluents, *inertial* unknown variables, *non-inertial* unknown variables and actions. Intuitively, both fluents and unknown variables encode properties of the world. An action can have effects on the former but not on the latter. Moreover, inertial variables are unchanged through courses of actions. Unknown variables are assumed to be independent of each other. A *fluent literal* is a fluent or a fluent preceded by $\neg$. An unknown variable literal is an unknown variable or an unknown variable preceded by $\neg$. A *literal* is either a fluent literal or an unknown variable literal. A *formula* is a propositional formula constructed from

[1] Since the proposal of PAL in Baral, Tran, & Le (2002), other action languages have also been extended to allow actions with probabilistic effects; for example see Eiter & Lukasiewicz (2003).

literals.

A state s is an interpretation of fluents and unknown variables that satisfy certain conditions (to be mentioned while discussing semantics). For a state s, the sub-interpretations of s restricted to fluents, inertial unknown variables, and non-inertial unknown variables are denoted by s_F, s_I, and s_N respectively.

In the following we briefly review the components of PAL.

The domain description language: A *domain description* is a collection of propositions of the forms:

$$a \textbf{ causes } \psi \textbf{ if } \varphi \tag{1}$$

$$\theta \textbf{ causes } \psi \tag{2}$$

$$\textbf{impossible } a \textbf{ if } \varphi \tag{3}$$

where a is an action, ψ is a *fluent* formula, θ is a formula of fluents and *inertial* unknown variables, and φ is a formula of fluents and *unknown* variables. Propositions of the form (1), called *dynamic causal laws*, describe the direct effects of actions. Propositions of the form (2), called *static causal laws*, describe causal relations between fluents and unknown variables. Propositions of the form (3), called *executability conditions*, state when actions are not executable.

The semantics of domain descriptions is defined based on transition functions. In the following, let $\mathcal{D}$ be a domain description in the language PAL.

An *interpretation* I of the fluents and unknown variables in $\mathcal{D}$ is a maximal consistent set of literals of $\mathcal{D}$. A literal l is said to be true (resp. false) in I iff $l \in I$ (resp. $\neg l \in I$). The truth value of a formula in I is defined recursively over the propositional connective in the usual way. For example, $f \wedge g$ is true in I iff f is true in I and g is true in I. The formula ψ is said to hold in I (or I satisfies ψ), denoted by $I \models \psi$, if ψ is true in I.

A set of formulas from $\mathcal{D}$ is *logically closed* if it is closed under propositional logic (w.r.t. $\mathcal{D}$). Let V be a set of formulas and K be a set of static causal laws of the form θ **causes** ψ. V is said to be closed under K if for every rule θ **causes** ψ in K, if θ belongs to V then so does ψ. $Cn_K(V)$ denotes the least logically closed set of formulas from $\mathcal{D}$ that contains V and is also closed under K. A *state* s of $\mathcal{D}$ is an interpretation that is closed under the set of static causal laws of $\mathcal{D}$.

An action a is *prohibited (not executable)* in a state s if there exists in $\mathcal{D}$ an executability condition of the form **impossible** a **if** φ such that φ holds in s. The *effect of an action a* in a state s is the set of formulas:

$$E_a(\mathsf{s}) = \{\psi \mid \exists r \in \mathcal{D} : r = a \textbf{ causes } \psi \textbf{ if } \varphi\ ,\ \mathsf{s} \models \varphi\}.$$

Let $\mathcal{S}$ be the set of the states of $\mathcal{D}$. A *transition function* Φ is a function from $\mathcal{A} \times \mathcal{S}$ to $2^{\mathcal{S}}$. If a is prohibited in s, then $\Phi(a, \mathsf{s}) = \emptyset$; otherwise we have that

$$\Phi(a, \mathsf{s}) = \{\ \mathsf{s}' \mid \mathsf{s}'_{F,I} = Cn_R((\mathsf{s}_{F,I} \cap \mathsf{s}'_{F,I}) \cup E_a(\mathsf{s}))\}$$

where R is the set of the static causal laws of $\mathcal{D}$.

An extended transition function $\hat{\Phi}$ expresses the state transition due to a sequence of actions, which is defined as

$$\hat{\Phi}([a], \mathsf{s}) = \Phi(a, \mathsf{s});$$

$$\hat{\Phi}([a_1, \ldots, a_n], \mathsf{s}) = \bigcup_{\mathsf{s}' \in \hat{\Phi}(a_1, \mathsf{s})} \hat{\Phi}([a_2, \ldots, a_n], \mathsf{s}').$$

Let s be a state in a domain description $\mathcal{D}$. Then s *entails* φ **after** $a_1, \ldots, a_n$, written as $\mathsf{s} \models \varphi$ **after** $a_1, \ldots, a_n$, if φ is true in all states in $\hat{\Phi}([a_1, \ldots, a_n], \mathsf{s})$.

Probabilities of unknown variables: A probability description $\mathcal{P}$ of the unknown variables is a collection of propositions of the form

$$\textbf{probability of } u \textbf{ is } p\ ;$$

where u is an unknown variable, and $p \in [0, 1]$. Each proposition above directly gives us the probability distribution of the corresponding unknown variable as: $P(u) = p$.

For any state s, s_u denotes the interpretation of the unknown variables of s, that is, $\mathsf{s}_u = \mathsf{s}_I \cup \mathsf{s}_N$. The unconditional probability of the various states is defined as:

$$P(\mathsf{s}) = \frac{P(\mathsf{s}_u)}{|\{\mathsf{s}' | \mathsf{s}'_u = \mathsf{s}_u)\}|}$$

Example 1 Let us consider a domain of coin tossing, in which the coin can be fair or fake with some probability p. If it is a fair coin, then it will land head with probability q_1. If it is fake, it lands head with probability q_2. The PAL domain has an inertial unknown variable u describing the coin's fairness and two non-inertial variables v_1, v_2 describing the landing head of different coin types:

$$\textbf{probability of } u \textbf{ is } p$$
$$\textbf{probability of } v_1 \textbf{ is } q_1$$
$$\textbf{probability of } v_2 \textbf{ is } q_2$$

Besides, the domain has a fluent $head$ describing the outcome of action $toss$. We have that:

$$toss \textbf{ causes } head \textbf{ if } u, v_1$$
$$toss \textbf{ causes } \neg head \textbf{ if } u, \neg v_1$$
$$toss \textbf{ causes } head \textbf{ if } \neg u, v_2$$
$$toss \textbf{ causes } \neg head \textbf{ if } \neg u, \neg v_2$$

The query language: Let φ be a formula of fluents and unknown variables, a_i's be actions, and $p \in [0, 1]$. A query has the following form:

$$\textbf{probability of } [\varphi \textbf{ after } a_1, \ldots, a_n] \textbf{ is } p\ .$$

The entailment of queries is defined in several steps. First, we define the transitional probability between states:

$$P_{[a]}(\mathsf{s}'|\mathsf{s}) = P_a(\mathsf{s}'|\mathsf{s}) = \begin{cases} \frac{2^{|U_N|}}{|\Phi(a,\mathsf{s})|} P(\mathsf{s}'_N) \text{ if } \mathsf{s}' \in \Phi(a, \mathsf{s}) \\ 0, \text{ otherwise.} \end{cases}$$

The (probabilistic) correctness of a single action plan given an initial state state s is defined as follows.

$$P(\varphi \textbf{ after } a|\mathsf{s}) = \sum_{\mathsf{s}' \in \Phi(a,\mathsf{s}) \wedge \mathsf{s}' \models \varphi} P_a(\mathsf{s}'|\mathsf{s})$$

Next, the transitional probability due to a sequence of actions, is recursively defined starting with the base case.

$$P_{[\,]}(\mathsf{s}'|\mathsf{s}) = \begin{cases} 1 \text{ if } \mathsf{s} = \mathsf{s}' \\ 0, \text{ otherwise.} \end{cases}$$

$$P_{[a_1,\ldots a_n]}(\mathsf{s}'|\mathsf{s}) = \sum_{\mathsf{s}''} P_{[a_1,\ldots,a_{n-1}]}(\mathsf{s}''|\mathsf{s}) P_{a_n}(\mathsf{s}'|\mathsf{s}'')$$

Finally, we define the (probabilistic) correctness of a (multi-action) plan given an initial state s:

$$P(\varphi \textbf{ after } \alpha|\mathsf{s}) = \sum_{\mathsf{s}' \in \hat{\Phi}([\alpha],\mathsf{s}) \wedge \mathsf{s}' \models \varphi} P_{[\alpha]}(\mathsf{s}'|\mathsf{s}) \qquad (4)$$

The observation language: An observation description $\mathcal{O}$ is a collection of proposition of the form

$$\psi \textbf{ obs\_after } a_1, \ldots a_n \ ;$$

where ψ is a fluent formula, and $a_1, \ldots, a_n$ are actions. When $n = 0$, it is simply written as **initially** ψ. The probability $P(\varphi$ **obs_after** $\alpha|\mathsf{s})$ is computed by the right hand side of (4).

Using the Bayes' rule, the conditional probability of a state given some observations is given as follows.

$$P(\mathsf{s}_i|\mathcal{O}) = \begin{cases} \frac{P(\mathcal{O}|\mathsf{s}_i)P(\mathsf{s}_i)}{\sum_{\mathsf{s}_j} P(\mathcal{O}|\mathsf{s}_j)P(\mathsf{s}_j)} \text{ if } \sum_{\mathsf{s}_j} P(\mathcal{O}|\mathsf{s}_j)P(\mathsf{s}_j) \neq 0 \\ 0, \text{ otherwise .} \end{cases}$$

The (probabilistic) correctness of a (multi-action) plan given only some observations is defined by

$$P(\varphi \textbf{ after } \alpha|\mathcal{O}) = \sum_{\mathsf{s}} P(\mathsf{s}|\mathcal{O}) \times P(\varphi \textbf{ after } \alpha|\mathsf{s}) \qquad (5)$$

An action theory $\mathcal{T}$ in PAL consists of a domain description $\mathcal{D}$, a probability description $\mathcal{P}$ of the unknown variables, and an observation description $\mathcal{O}$: $\quad \mathcal{T} = \mathcal{D} \cup \mathcal{P} \cup \mathcal{O}$.
Let Q be a query:

$$Q = \textbf{ probability of } [\varphi \textbf{ after } a_1, \ldots, a_n] \textbf{ is } p \ .$$

Then $\mathcal{T}$ entails Q, written as $\mathcal{T} \models_A Q$, if and only if $P(\varphi$ **after** $a_1, \ldots, a_n|\mathcal{O}) = p$. The entailment can be written in the shorter form as

$$\mathcal{T} \models_A P(\varphi \textbf{ after } a_1, \ldots, a_n|\mathcal{O}) = p \ .$$

In the next section, we briefly review PCM (Pearl 2000).

Probabilistic causal models

Causal model: A *causal model* is a triple $M = \langle U, V, F\rangle$ where

* U is a set of *background* variables, (also called *exogenous*), that are determined by factors outside the model;
* V is a set $\{V_1, V_2, \ldots V_n\}$ of variables, called *endogenous*, that are determined by variables in the model - that is, variables in $U \cup V$; and
* F is a set of functions $\{f_1, f_2, \ldots f_n\}$, such that each f_i is a mapping from $U \cup (V \setminus V_i)$ to V_i, and such that the entire set F forms a mapping from U to V. In other words, each f_i tells us the value of V_i given the values of all other variables in $U \cup V$, and the entire set F has a unique solution for V, given a realization of U. Symbolically, the set of equations F can be represented by writing

  $$V_i = f_i(PA_i, U_i) \ i = 1, \ldots, n$$

  where PA_i is a subset of variables in $V \setminus V_i$ and U_i stands for a subset of variables in U.

Example 2 We have a simple causal model consisting of a background variable U, endogenous variables A and B, and the set of the following functions:

$$A = U \wedge B;$$
$$B = U \vee A.$$

For each value of U, there is a unique solution A, B. That is, if $U = 1$, then $A = B = 1$; otherwise $A = B = 0$. □

Submodel: Let M be a causal model, X be a set of variables in V, and x be a particular realization of X. A *submodel* M_x of M is the causal model $M_x = \langle U, V, F_x\rangle$ where $F_x = \{f_i : V_i \notin X\} \cup \{X = x\}$.

Submodels are useful for representing the effect of local actions and hypothetical changes. M_x represents the model that results from a minimal change to make $X = x$ hold true under any realization of U.

Probabilistic causal model: A *probabilistic causal* model (PCM) is a pair $\langle M, P\rangle$ where M is a causal model and P is a probability distribution over the domain of U.

Note that because the set of functional equations forms a mapping from U to V, the probability distribution P also induces a probability distribution over the endogenous variables. Hence, given any subsets X and E of $U \cup V$, the conditional probability $P(x|e) = P(X = x|E = e)$ is well-defined in the model $\langle M, P\rangle$.

Probabilistic queries and their entailment in PCM

* Given a PCM $\mathcal{M} = \langle M, P\rangle$, the probability of x given an observation e is the conditional probability $P(x|e)$. If $P(x|e) = p$, we write $\mathcal{M} \models_C P(x|e) = p$.
* Given a PCM $\mathcal{M} = \langle M, P\rangle$, the probability of x given an intervention $do(y)$, denoted by $P(x|do(y))$, is the probability of x computed w.r.t the submodel $\mathcal{M}_y = \langle M_y, P\rangle$. If $P(x|do(y)) = p$, we write $\mathcal{M} \models_C P(x|do(y)) = p$.
* Given a PCM $\mathcal{M} = \langle M, P\rangle$, the probability of x given observation e and intervention $do(y)$, denoted by $P(x|e, do(y))$, is the probability $P(x|do(y))$ that is computed w.r.t the modified causal model $\mathcal{M}' = \langle M_y, P_e\rangle$. Here, P_e is the conditional probability $P(\_\,|e)$ computed w.r.t the model $\mathcal{M} = \langle M, P\rangle$. If $P(x|e, do(y)) = p$ we write $\mathcal{M} \models_C P(x|e, do(y)) = p$.

From the above definition it follows that $\langle M, P\rangle \models_C P(x|do(y)) = p$ if and only if $\langle M_y, P\rangle \models_C P(x) = p$; and $\langle M, P\rangle \models_C P(x|e, do(y)) = p$ if and only if $\langle M_y, P_e\rangle \models_C P_e(x) = p$.

Example 3 The PCM of the *firing squad example* (Pearl 1999; 2000) has two exogenous variables U and W, and endogenous variables A, B, C, and D; which stand for

$U =$ court orders the execution;
$C =$ captain gives a signal;
$A =$ rifle A shoots;
$B =$ rifle B shoots;
$D =$ the prisoner dies;
$W =$ rifle A pulls the trigger out of nervousness.

The causal relationships between the variables are described by the following functional equations:

$$C = U; \quad A = C \vee W; \quad B = C; \quad D = A \vee B.$$

The goal is to compute the probability $P(\neg D|D, do(\neg A))$, which expresses the *counterfactual* probability that the prisoner would be alive if A had not shot, given that the prisoner is in fact dead. It is shown that the PCM entails:

$$P(u, w|D) = \begin{cases} \frac{P(u,w)}{1-(1-p)(1-q)} & \text{if } u = U \text{ or } w = W \\ 0 & \text{if } u = \neg U \text{ and } w = \neg W \end{cases}$$
$$P(\neg D|D, do(\neg A)) = \frac{q(1-p)}{1-(1-p)(1-q)}$$

where $p = P(U)$ and $q = P(W)$. $\square$

Encoding PCM in PAL

In this section we give an encoding of PCM in PAL, illustrate the encoding with an example, and show the correspondence between query entailment in PCM and query entailment of the corresponding encoding in PAL.

General encoding of PCM in PAL: Given a PCM $\mathcal{M} = \langle M, P \rangle$ and assuming that the functions $f_i(PA_i, U_i)$ in M are logical functions, we construct a PAL action theory $D(\mathcal{M})$ as follows:

- There are no non-inertial unknown variables.
- The inertial unknown variables are the exogenous variables in M with the same probability distributions:

$$\textbf{probability of } \ u \ \textbf{is} \ \ P(u).$$

- The endogenous variables in M are fluents in $D(\mathcal{M})$. Moreover, for every fluent V_i, there is an additional fluent $ab(V_i)$ in $D(\mathcal{M})$.
- For each functional equation of the form $V_i = f_i(PA_i, U_i)$ in M, the following static causal rule is in $D(\mathcal{M})$:

$$\neg ab(V_i) \ \textbf{causes} \ \ V_i \ \Leftrightarrow \ f_i(PA_i, U_i).$$

- For every fluent V_i, $D(\mathcal{M})$ has actions '$make(V_i)$', '$make(\neg V_i)$' with the following effects:

$$\begin{array}{rl} make(V_i) & \textbf{causes} \ \ \{ab(V_i), V_i\} \\ make(\neg V_i) & \textbf{causes} \ \ \{ab(V_i), \neg V_i\}. \end{array}$$

The main intuition is that functional equations of the form $V_i = f_i(PA_i, U_i)$ need to be encoded as $\neg ab(V_i)$ **causes** $V_i = f_i(PA_i, U_i)$ instead of the straightforward encoding $true$ **causes** $V_i = f_i(PA_i, U_i)$. Then the equation $V_i = f_i(PA_i, U_i)$ can be properly inactivated by action $make(V_i)$.

Let us construct the PAL encoding of the PCM in Example 3. The action theory contains inertial unknown variables U and W, fluents A, B, C, D, $ab(A), ab(B), ab(C)$, and $ab(D)$. Translated into PAL, the functional equations become the following static causal laws:

$$\begin{array}{lll} \neg ab(C) & \textbf{causes} & C \Leftrightarrow U \\ \neg ab(A) & \textbf{causes} & A \Leftrightarrow C \vee W \\ \neg ab(B) & \textbf{causes} & B \Leftrightarrow C \\ \neg ab(D) & \textbf{causes} & D \Leftrightarrow A \vee B \end{array} \tag{6}$$

We now relate the probabilities entailed by the PCM in Example 3 with those entailed by its PAL encoding. (In the following, χ denotes the indicator function, that is, $\chi(X) = 1$ if X is true and $\chi(X) = 0$ if X is false.)

Proposition 1 Let $\mathcal{M} = \langle M, P \rangle$ be the PCM of firing squad in Example 3, and let us denote its encoding in PAL by $D(\mathcal{M})$.

Let us denote that

$$init_{\neg ab} = \{\textbf{initially } \neg ab(A), \neg ab(B), \neg ab(C), \neg ab(D)\} \ .$$

Then for any u and w literals of U and W:

$$P(\textbf{initially } \{u, w\}|init_{\neg ab}, \textbf{initially } D) = \begin{cases} \frac{P(u,w)}{1-(1-p)(1-q)} & \text{if } u = U \text{ or } w = W, \\ 0 & \text{if } u = \neg U \text{ and } w = \neg W. \end{cases} \tag{7}$$

$$P(\neg D \textbf{ after } make(\neg A)|init_{\neg ab}, \textbf{initially } D) = \frac{q(1-p)}{1-(1-p)(1-q)} \tag{8}$$

Proof (*sketch*).

First we use the Bayes' rule to compute as follows.

$$P(\textbf{initially } \{u, w\}|init_{\neg ab}, \textbf{initially } D) = \frac{P(\textbf{initially } \{u, w\}, \textbf{initially } D|init_{\neg ab})}{P(\textbf{initially } D|init_{\neg ab})} \tag{9}$$

Because of the static causal laws (6), given $init_{\neg ab}$, the variables U, W and the fluents A, B, C, D in the initial state satisfy that

$$\begin{array}{llllll} C & \Leftrightarrow & U, & A & \Leftrightarrow & C \vee W, \\ B & \Leftrightarrow & C, & D & \Leftrightarrow & A \vee B. \end{array} \tag{10}$$

It follows from (10) that $\neg D \Leftrightarrow \neg U \wedge \neg W$ (in the initial state). Hence,

$$\begin{aligned} & P(\textbf{initially } \neg D|init_{\neg ab}) \\ & = P(\textbf{initially } \neg U, \textbf{initially } \neg W|init_{\neg ab}) \\ & = P(U = 0, W = 0) = (1-p)(1-q). \end{aligned}$$

Therefore, we have:

$$\begin{aligned}&P(\textbf{initially } D|init_{\neg ab})\\&\quad= 1 - P(\textbf{initially } \neg D|init_{\neg ab})\\&\quad= 1-(1-p)(1-q).\end{aligned}\tag{11}$$

Because $\neg D \Leftrightarrow \neg U \wedge \neg W$ in the initial state, we also have that: **initially** $D \Leftrightarrow$ **initially** $U \vee$ **initially** W. Consequently, if $u = U$ or $w = W$ then:

$$\begin{aligned}&P(\textbf{initially } \{u,w\}, \textbf{initially } D|init_{\neg ab})\\&\quad= P(\textbf{initially } \{u,w\}|init_{\neg ab}) = P(u,w).\end{aligned}$$

Otherwise, if $u = \neg U$ and $w = \neg W$ then:

$$P(\textbf{initially } \{u,w\}, \textbf{initially } D|init_{\neg ab}) = 0.$$

Finally, we have that:

$$\begin{aligned}&P(\textbf{initially } \{u,w\}, \textbf{initially } D|init_{\neg ab})\\&\quad= \begin{cases} P(u,w) & \text{if } u = U \text{ or } w = W\\ 0 & \text{if } u = \neg U \text{ and } w = \neg W.\end{cases}\end{aligned}\tag{12}$$

It is easy to see that (7) follows from (9), (11) and (12).

For proving (8), we use the formula:

$$\begin{aligned}&P(\neg D \textbf{ after } make(\neg A)|init_{\neg ab}, \textbf{initially } D) =\\&\sum_{\mathsf{s}} P(\neg D \textbf{ after } make(\neg A)|\mathsf{s})P(\mathsf{s}|init_{\neg ab}, \textbf{initially } D)\end{aligned}\tag{13}$$

Observe that $P(\mathsf{s}|init_{\neg ab}, \textbf{initially } D) = 0$ if $init_{\neg ab}$ does not hold in s (that is, $\mathsf{s} \not\models init_{\neg ab}$). Hence, the right hand side depends only on the terms containing s such that $\mathsf{s} \models init_{\neg ab}$. In the following we will consider only s such that $\mathsf{s} \models init_{\neg ab}$. Let us assume that u and w are the literals of U and W that hold in the initial state s. The variables and fluents in the initial state s satisfy the functions in (10). Consequently, the variables and fluents in s are uniquely determined by the values of U and W, that is, by u and w. So s is also uniquely determined by u and w. Thus we have:

$$\begin{aligned}&P(\mathsf{s}|init_{\neg ab}, \textbf{initially } D)\\&\quad= P(\textbf{initially } \{u,w\}|init_{\neg ab}, \textbf{initially } D)\end{aligned}\tag{14}$$

Assume that we reach the state s' by executing $make(\neg A)$ in the initial state s. The action causes $ab(A)$ and $\neg A$ to be true. Then it follows from the static causal laws (6) that in the state s': $C \Leftrightarrow U$, $B \Leftrightarrow C$ and $D \Leftrightarrow B$. Therefore, in the state s', we have that $D \Leftrightarrow U$. Because U is an inertial unknown variable, its values are the same in s and s'. Consequently,

$$\begin{aligned}&\neg D \textbf{ after } make(\neg A) \Leftrightarrow \neg D \in \mathsf{s}'\\&\quad\Leftrightarrow \neg U \in \mathsf{s}' \Leftrightarrow \neg U \in \mathsf{s} \Leftrightarrow \textbf{initially } \neg U\end{aligned}$$

Since s uniquely depends on u, w:

$$P(\textbf{initially } \neg U|\mathsf{s}) = P(\textbf{initially } \neg U|u,w)$$

Thus we have:

$$\begin{aligned}&P(\neg D \textbf{ after } make(\neg A)|\mathsf{s})\\&\quad= P(\textbf{initially } \neg U|\mathsf{s}) = \chi(u = \neg U)\end{aligned}\tag{15}$$

From (13), (14) and (15), we have that:

$$\begin{aligned}&P(\neg D \textbf{ after } make(\neg A)|init_{\neg ab}, \textbf{initially } D)\\&= \sum_{u,w} \chi(u = \neg U)P(\textbf{initially } \{u,w\}|init_{\neg ab}, \textbf{initially } D)\end{aligned}$$

Note that $\chi(u = \neg U) \neq 0$ only if $u = \neg U$. Furthermore, because of (7), if $u = \neg U$ then $P(\textbf{initially } \{u,w\}|init_{\neg ab}, \textbf{initially } D) \neq 0$ only if $w = W$. So the only possible positive term in the above sum corresponds to the pair $u = \neg U, w = W$. Then:

$$\begin{aligned}&P(\neg D \textbf{ after } make(\neg A)|init_{\neg ab}, \textbf{initially } D)\\&\quad= P(\textbf{initially } \{\neg U, W\}|init_{\neg ab}, \textbf{initially } D)\\&\quad= \frac{P(\neg U, W)}{1-(1-p)(1-q)} = \frac{q(1-p)}{1-(1-p)(1-q)}.\end{aligned}$$

Relating PCM and PAL: the main result

We generalize Proposition 1 to the following general result.

Theorem 1 *Given a PCM $\mathcal{M} = \langle M, P\rangle$, let $D(\mathcal{M})$ be its respectively constructed PAL action theory. Let $init_{\neg ab} = \{$**initially** $\neg ab(v)|v \in V\}$. Let u be a subset of background variable, v and w be subsets of endogenous variables. Then we have the following relations between entailments in PCM and PAL:*

- *$\mathcal{M}$ entails $P(u|w) = p$ if and only if $D(\mathcal{M})$ entails*
$$P(\textbf{initially } u|init_{\neg ab}, \textbf{initially } w) = p$$
- *$\mathcal{M}$ entails $P(w|do(v)) = p$ if and only if $D(\mathcal{M})$ entails*
$$P(w \textbf{ after } do(v)|init_{\neg ab}) = p$$
- *$\mathcal{M}$ entails $P(\neg w|w, do(\neg v)) = p$ if and only if $D(\mathcal{M})$ entails*
$$P(\neg w \textbf{ after } make(\neg v)\ |init_{\neg ab}, \textbf{initially } w) = p$$

Due to lack of space, we do not present the proof Theorem 1 in detail. This proof is based on a reasoning similar to that of the proof of Proposition 1.

Related works and discussion

The works closely related to ours include Poole (1997), Thielscher (1999), Finzi & Lukasiewicz (2003)[2]. These works also provide formal translations between probabilistic formalisms for reasoning about actions.

Poole (1997) introduces *independent choice logic* (ICL) of independent choices and logic programs that specify the consequence of choices and actions. Poole (1997) has shown that many formalisms can be translated into ICL, including influence diagrams, Markov decision problems (MDPs), strategic form of games and (dynamic) Bayesian networks. The relation between ICL and PAL is still unknown.

Thielscher (1999) translates causal models (without probabilities) into Fluent Calculus. Like the fluents $ab(V_i)$, their generic fluent $Denied(x)$ is used to simulate how an action inactivates respective functional equations. Nevertheless, the

[2]We thank anonymous reviewers for pointing this out.

semantics of counterfactuals by Thielscher (1999) is essentially non-probabilistic. We also plan to relate this semantics to PAL in the future extension of our work.

Finzi & Lukasiewicz (2003) provide translations between PCM and ICL then extends the notions of structural causes and explanations to ICL-theories. The translations are restricted to a special class of PCMs whose causal graphs are *acyclic*. This restriction is probably due to the fact that ICL is defined with acyclic logic programs. Moreover, their and our method of analysis are orthogonal. In Finzi & Lukasiewicz (2003), the extended notions are first translated into PCM together with the ICL; then they are semantically interpreted in the translated PCM model. In our case, probabilistic queries can be interpreted separately in PCM and PAL semantics. Our contribution was showing that the translation from PCM to PAL is semantically correct. Finally, counterfactual reasoning was a major point in our work (see (iii) in Theorem 1), which has not been studied in Finzi & Lukasiewicz (2003).

Baral, Tran, & Le (2002) discuss differences between PAL and PCM in detail. With regard to our work in this paper, it is noticeable that we have used only inertial variables in the encoding of PCM. Another important aspect of the PAL encoding is that as the world progresses if we want to reactivate a previously inactivated functional equation $V_i = f_i(PA_i, U_i)$ all we need to do is make $ab(V_i)$ false. (Reactivation need to be done with care though.) In PCMs once a functional equation is inactivated it can no longer be activated.

Finally, the proof of Proposition 1 (as well as that of Theorem 1) has referred to an important assumption of causal model. It is the assumption that the set of functional equations uniquely determines the values of the endogenous variables, given the values of the exogenous variables. Moreover, counterfactual probabilities are not well-defined in PCM without the assumption. While it is not clear how to do extend counterfactual reasoning in PCM without this assumption, it is straightforward to do so in PAL framework.

Conclusion

In this paper we have presented a translation of probabilistic causal models into the action description language PAL. The translation shows that the language of PCMs is semantically equivalent to a sub-language of PAL, which contains only *inertial* unknown variables and a special set of actions acting on fluents '$ab(V_i)$'. Our result has several important implications: (i) PAL is more expressive than PCM; (ii) PCM does not capture probabilistic effects of actions, which are encoded by *non-inertial* unknown variables in PAL; (iii) PAL is more suitable than PCM for reasoning about actions.

Acknowledgments

The authors acknowledge the support of NSF through grant number 0070463 and of NASA through grant number NCC2-1232.

References

Bacchus, F.; Halpern, J. Y.; and Levesque, H. J. 1999. Reasoning about noisy sensors and effectors in the situation calculus. *AI journal* 111(1-2):171–208.

Baral, C.; Tran, N.; and Le, T. 2002. Reasoning about actions in a probabilistic setting. In *Proc. AAAI'02*, 507–512.

Eiter, T., and Lukasiewicz, T. 2003. Probabilistic reasoning about actions in nonmonotonic causal theories. In *Proc. UAI'03*, 192–199.

Finzi, A., and Lukasiewicz, T. 2003. Structure-based causes and explanations in the independent choice logic. In *Proc. UAI'03*, 225–232.

Gelfond, M., and Lifschitz, V. 1993. Representing actions and change by logic programs. *Journal of Logic Programming* 17:301–323.

Gelfond, M., and Lifschitz, V. 1998. Action languages. *ETAI* 3(16).

Kushmerick, N.; Hanks, S.; and Weld, D. 1995. An algorithm for probabilistic planning. *AI journal* 76(1-2):239–286.

Littman, M. 1997. Probabilistic propositional planning: representations and complexity. In *Proc. of AAAI'97*, 748–754.

Mateus, P.; Pacheco, A.; and Pinto, J. 2002. Observations and the probabilistic situation calculus. In *Proc. KR'02*, 327338.

McCain, N., and Turner, H. 1995. A causal theory of ramifications and qualifications. In *Proc. of IJCAI'95*, 1978–1984.

Pearl, J. 1995. Action as a local surgery. In *Working Notes of AAAI Spring Symposium Series*, 157–162.

Pearl, J. 1999. Reasoning with cause and effect. In *Proc. of IJCAI'99*, 1437–1449.

Pearl, J. 2000. *Causality*. Cambridge University Press.

Poole, D. 1997. The independent choice logic for modelling multiple agents under uncertainty. *AI journal* 94:7–56.

Poole, D. 1998. Decision theory, the situation calculus and conditional plans. *ETAI* 2(1-2):105158.

Reiter, R. 2001. *Knowledge in action: logical foundation for describing and implementing dynamical systems*. MIT Press.

Thielscher, M. 1999. In *KI'99*, 137–148.