by making the cause of the coin toss outcome a non-inertial unknown variable. *Overall, our formulation can be considered as a generalization of Pearl's formulation of* causality *to a dynamic setting with a more elaboration tolerant representation and with two kinds of unknown variables.*

We now start with the syntax and the semantics our language PAL, which stands for *probabilistic action language*.

The Language PAL

The alphabet of the language PAL (denoting probabilistic action language) – based on the language $\mathcal{A}$ (Gelfond & Lifschitz 1993) – consists of four non-empty disjoint sets of symbols **F**, $\mathbf{U}_I$, $\mathbf{U}_N$ and **A**. They are called the set of fluents, the set of inertial unknown variables, the set of non-inertial unknown variables and the set of actions. 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 literals.

Unknown variables represent unobservable characteristics of environment. As noted earlier, there are two types of unknown variables: *inertial* and *non-inertial*. *Inertial* unknown variables are not affected by agent's actions and are independent of fluents and other unknown variables. *Non-inertial* unknown variables may change their value respecting a given probability distribution, but the pattern of their change due to actions is neither known nor modeled in our language.

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, we denote the sub-interpretations of s restricted to fluents, inertial unknown variables, and non-inertial unknown variables by s_F, s_I, and s_N respectively. We also use the shorthand such as $\mathsf{s}_{F,I} = \mathsf{s}_F \cup \mathsf{s}_I$. An n-state is an interpretation of only the fluents. That is, if s is a state, then $s = \mathsf{s}_F$ is an n-state. A u-state (s_u) is an interpretation of the unknown variables. For any state s, by s_u we denote the interpretation of the unknown variables of s. For any u-state s_u, $I(s_u)$ denotes the set of states s, such that $\mathsf{s}_u = s_u$. We say $\mathsf{s} \models s$, if the interpretation of fluents in s is same as in s.

PAL has four components: a domain description language PAL_D, a language to express unconditional probabilities about the unknown variables PAL_P, a language to specify observations PAL_O, and a query language.

PAL_D: The domain description language

Syntax Propositions in PAL_D are of the following forms:

$$a \textbf{ causes } \psi \textbf{ if } \varphi \quad (0.1)$$

$$\theta \textbf{ causes } \psi \quad (0.2)$$

$$\textbf{impossible } a \textbf{ if } \varphi \quad (0.3)$$

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. Note that the above propositions guarantee that values of unknown variables are not affected by actions and are not dependent on the fluents. But the effect of an action on a fluent may be dependent on unknown variables; also only inertial unknown variables may have direct effects on values of fluents.

Propositions of the form (0.1) describe the direct effects of actions on the world and are called *dynamic causal laws*. Propositions of the form (0.2), called *static causal laws*, describe causal relation between fluents and unknown variables in a world. Propositions of the form (0.3), called *executability conditions*, state when actions are not executable.

A *domain description* $\mathcal{D}$ is a collection of propositions in PAL_D.

Semantics of PAL_D: Characterizing the transition function A domain description given in the language of PAL_D defines a transition function from actions and states to a set of states. Intuitively, given an action (a), and a state (s), the transition function (Φ) defines the set of states ($\Phi(a, \mathsf{s})$) that may be reached after executing the action a in state s. If $\Phi(a, \mathsf{s})$ is an empty set it means that a is not executable in s. We now formally define this transition function.

Let $\mathcal{D}$ be a domain description in the language of PAL_D. An *interpretation* I of the fluents and unknown variables in PAL_D is a maximal consistent set of literals of PAL_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, $f \wedge q$ is true in I iff f is true in I and q is true in I. We say that ψ holds in I (or I satisfies ψ), denoted by $I \models \psi$, if ψ is true in I.

A set of formulas from PAL_D is *logically closed* if it is closed under propositional logic (w.r.t. PAL_D).

Let V be a set of formulas and K be a set of static causal laws of the form θ **causes** ψ. We say that V is closed under K if for every rule θ **causes** ψ in K, if θ belongs to V then so does ψ. By $Cn_K(V)$ we denote the least logically closed set of formulas from PAL_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 $E_a(\mathsf{s}) = \{\psi \mid \mathcal{D}$ contains a law a **causes** ψ **if** φ and φ holds in $\mathsf{s}\}$.

Given a domain description $\mathcal{D}$ containing a set of static causal laws R, we follow (McCain & Turner 1995) to formally define $\Phi(a, \mathsf{s})$, the set of states that may be reached by executing a in s as follows.

If a is not prohibited (i.e., executable) in s, then

$$\Phi(a, \mathsf{s}) = \{\ \mathsf{s}' \mid \mathsf{s}'_{F,I} = Cn_R((\mathsf{s}_{F,I} \cap \mathsf{s}'_{F,I}) \cup E_a(\mathsf{s}))\ \}; \quad (0.4)$$

If a is prohibited (i.e., not executable) in s, then $\Phi(a, \mathsf{s})$ is $\emptyset$.

We now state some simple properties of our transition function.

Proposition 1 *Let $U_N \subseteq U$ be the set of non-inertial variables in U.*

1. *If $\mathsf{s}' \in \Phi(a, \mathsf{s})$ then $\mathsf{s}'_I = \mathsf{s}_I$. That is, the inertial unknown variables are unchanged through state transitions.*

2. *For every* $s' \in \Phi(a, s)$ *and for every interpretation* w *of* U_N*, we have that* $(s'_{F,I} \cup w) \in \Phi(a, s)$.

Every domain description $\mathcal{D}$ in a language PAL_D has a unique transition function Φ, and we say Φ is the transition function of $\mathcal{D}$.

We now define an extended transition function (with a slight abuse of notation) that expresses the state transition due to a sequence of actions.

Definition 1 $\Phi([a], s) = \Phi(a, s)$;
$\Phi([a_1, \ldots, a_n], s) = \bigcup_{s' \in \Phi(a_1, s)} \Phi([a_2, \ldots, a_n], s')$

Definition 2 Given a domain description $\mathcal{D}$, and a state s, we write $s \models_{\mathcal{D}} \varphi$ **after** $a_1, \ldots, a_n$,
if φ is true in all states in $\Phi([a_1, \ldots, a_n], s)$.
(Often when it is clear from the context we may simply write $\models$ instead of $\models_{\mathcal{D}}$.)

PAL_P: Probabilities of unknown variables

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

$$\textbf{probability of } u \textbf{ is } n \tag{0.5}$$

where u is an unknown variable, and n is a real number between 0 and 1.

Semantics Each proposition above directly gives us the probability distribution of the corresponding unknown variable as: $P(u) = n$.

Since we assume (as does Pearl (Pearl 2000)) that the values of the unknown variables are independent of each other defining the joint probability distribution of the unknown variables is straight forward.

$$P(u_1, \ldots, u_n) = P(u_1) \times \ldots \times P(u_n) \tag{0.6}$$

Note: $P(u_1)$ is a short hand for $P(U_1 = true)$. If we have multi-valued unknown variables then $P(u_1)$ will be a short hand for $P(U_1 = u_1)$.

Since several states may have the same interpretation of the unknown variables and we do not have any unconditional preference of one state over another, the unconditional probability of the various states can now be defined as:

$$P(s) = \frac{P(s_u)}{|I(s_u)|} \tag{0.7}$$

PAL_Q: The Query language

Syntax A query is of the form:

$$\textbf{probability of } [\varphi \textbf{ after } a_1, \ldots, a_n] \textbf{ is } n \tag{0.8}$$

where φ is a formula of fluents and unknown variables, a_i's are actions, and n is a real number between 0 and 1. When $n = 1$, we may simply write: φ **after** $a_1, \ldots, a_n$, and when $n = 0$, we may simply write $\neg\varphi$ **after** $a_1, \ldots, a_n$.

Semantics: Entailment of Queries in PAL_Q We define the entailment in several steps. First we define the transitional probability between states due to a single action.

$$P_{[a]}(s'|s) = P_a(s'|s) \quad = \frac{|\Phi(a,s)|}{2^{|U_N|}} P(s'_N) \text{ if } s' \in \Phi(a, s); \quad = 0, \text{ otherwise .} \tag{0.9}$$

The intuition behind (0.9) is as follows: Since inertial variables do not change their value from one state to the next, $P_a(s'|s)$ will depend only on the conditioning of fluents and non-inertial variables: $P_a(s'|s) = P_a(s'_{F,N}|s)$. Since non-inertial variables are independent from the transition, we have $P_a(s'_{F,N}|s) = P_a(s'_F|s) * P(s'_N)$. Since there is no distribution associated with fluents, we assume that $P_a(s'_F|s)$ is uniformly distributed. Then $P_a(s'_F|s) = \frac{|\Phi(a,s)|}{2^{|U_N|}}$, because there are $\frac{|\Phi(a,s)|}{2^{|U_N|}}$ possible next states that share the same interpretation of unknown variables.

We now define the (probabilistic) correctness of a single action plan given that we are in a particular state s.

$$P(\varphi \textbf{ after } a|s) = \sum_{s' \in \Phi(a,s) \wedge s' \models \varphi} P_a(s'|s) \tag{0.10}$$

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

$$P_{[\,]}(s'|s) = 1 \text{ if } s = s'; \text{ otherwise it is } 0. \tag{0.11}$$

$$P_{[a_1, \ldots a_n]}(s'|s) = \sum_{s''} P_{[a_1, \ldots, a_{n-1}]}(s''|s) P_{a_n}(s'|s'') \tag{0.12}$$

We now define the (probabilistic) correctness of a (multi-action) plan given that we are in a particular state s.

$$P(\varphi \textbf{ after } \alpha|s) = \sum_{s' \in \Phi([\alpha], s) \wedge s' \models \varphi} P_{[\alpha]}(s'|s) \tag{0.13}$$

PAL_O: The observation language

Syntax An observations description $\mathcal{O}$ is a collection of proposition of the following form:

$$\psi \textbf{ obs\_after } a_1, \ldots, a_n \tag{0.14}$$

where ψ is a fluent formula, and a_i's are actions. When, $n = 0$, we simply write **initially** ψ. *Intuitively, the above observation means that* ψ *is true after a particular – because actions may be non-deterministic – hypothetical execution of* $a_1, \ldots, a_n$ *in the initial state.* The probability $P(\varphi$ **obs_after** $\alpha|s)$ is computed by the right hand side of (0.13). Note that observations in $\mathcal{A}$ and hence in PAL_O are hypothetical in the sense that they did not really happen. In a later section when discussing narratives we consider real observations.

Semantics: assimilating observations in PAL_O We now use Bayes' rule to define the conditional probability of a state given that we have some observations.

$$P(s_i|\mathcal{O}) \quad = \frac{P(\mathcal{O}|s_i)P(s_i)}{\sum_{s_j} P(\mathcal{O}|s_j)P(s_j)} \text{ if } \sum_{s_j} P(\mathcal{O}|s_j)P(s_j) \neq 0; \quad = 0, \text{ otherwise .} \tag{0.15}$$

Queries with observation assimilation

Finally, we define the (probabilistic) correctness of a (multi-action) plan given only some observations. This corresponds to counter-factual queries of Pearl (Pearl 2000) when the observations are about a different sequence of actions than the one in the hypothetical plan.

$$P(\varphi \textbf{ after } \alpha|\mathcal{O}) = \sum_{s} P(s|\mathcal{O}) \times P(\varphi \textbf{ after } \alpha|s) \tag{0.16}$$

Using the above formula, we now define the entailment between a theory (consisting of a domain description, a probability description of the unknown variables, and an observation description) and queries:

Definition 3 $\mathcal{D} \cup \mathcal{P} \cup \mathcal{O} \models$ **probability of** $[\varphi$ **after** $a_1, \ldots, a_n]$ **is** n iff $P(\varphi$ **after** $a_1, \ldots, a_n | \mathcal{O}) = n$

Since our observations are hypothetical and are about a particular hypothetical execution, it is possible that[2] $P(\varphi$ **after** $\alpha | \varphi$ **obs_after** $\alpha) < 1$, when α has non-deterministic actions. Although it may appear unintuitive in the first glance, it is reasonable as just because a particular run of α makes φ true does not imply that *all* run of α would make φ true.

Examples

In this section we give several small examples illustrating the reasoning formalized in PAL.

Ball drawing

We draw a ball from an infinitely large "black box". Let $draw$ be the action, red be the fluent describing the outcome and u be an unknown variable that affects the outcome. The domain description is as follow:

$draw$ **causes** red **if** u. $draw$ **causes** $\neg red$ **if** $\neg u$.
probability of u **is** 0.5.

Let $\mathcal{O} = red$ **obs_after** $draw$ and $Q = red$ **after** $draw, draw$. Different assumptions about the variable u will lead to different values of $p = P(Q|\mathcal{O})$.

Let $\mathsf{s}_1 = \{red, u\}$, $\mathsf{s}_2 = \{red, \neg u\}$, $\mathsf{s}_3 = \{\neg red, u\}$ and $\mathsf{s}_4 = \{\neg red, \neg u\}$.

1. Assume that the balls in the box are of the same color, and there are 2 possibilities: the box contains either all red or all blue balls. Then u is an inertial unknown variable. We can now show that $P(Q|\mathcal{O}) = 1$. Here, the initial observation tells us all about the future outcomes.

2. Assume that half of the balls in the box are red and the other half are blue. Then u is a non-inertial unknown variable. We can show that $P(\mathsf{s}_1|\mathcal{O}) = P(\mathsf{s}_3|\mathcal{O}) = 0.5$ and $P(\mathsf{s}_2|\mathcal{O}) = P(\mathsf{s}_4|\mathcal{O}) = 0$. By (0.13), $P(Q|\mathsf{s}_j) = 0.5$ for $1 \leq j \leq 4$. By (0.16), $P(Q|\mathcal{O}) = 0.5 * \sum_{\mathsf{s}_j} P(\mathsf{s}_j|\mathcal{O}) = 0.5$. Here, the observation $\mathcal{O}$ does not help in predicting the future.

The Yale shooting

We start with a simple example of the Yale shooting problem with probabilities. We have two actions $load$ and $shoot$, and two fluents $loaded$ and $alive$. To account for the probabilistic effect of the actions, we have two inertial unknown variables u_1 and u_2. The effect of the actions $shoot$ and $load$ can now be described by $\mathcal{D}_1$ consisting of the following:

$shoot$ **causes** $\neg alive$ **if** $loaded, u_1$
$load$ **causes** $loaded$ **if** u_2

The probabilistic effects of the action $shoot$ and $load$ can now be expressed by $\mathcal{P}_1$, that gives probability distributions of the unknown variables.

probability of u_1 **is** p_1. **probability of** u_2 **is** p_2.

Now suppose we have the following observations $\mathcal{O}_1$.

initially $alive$; **initially** $\neg loaded$

We can now show that $\mathcal{D}_1 \cup \mathcal{P}_1 \cup \mathcal{O}_1 \models$ **probability of** $[alive$ **after** $load, shoot]$ **is** $1 - p_1 \times p_2$.

[2]We thank an anonymous reviewer for pointing this out.

Pearl's example of effects of treatment on patients

In (Pearl 2000), Pearl gives an example of a joint probability distribution which can be expressed by at least two different causal models, each of which have a different answer to a particular counter-factual question. We now show how both models can be modeled in our framework. In his example, the data obtained on a particular medical test where half the patients were treated and the other half were left untreated shows the following:

treated alive	true true	true false	false true	false false
fraction	.25	.25	.25	.25

The above data can be supported by two different domain descriptions in PAL, each resulting in different answers to the following question involving counter-factuals. *"Joe was treated and he died. Did Joe's death occur due to the treatment. I.e., Would Joe have lived if he was not treated.*

Causal Model 1: The domain description $\mathcal{D}_2$ of the causal model 1 can be expressed as follows, where the actions in our language are, $treatment$ and $no\_treatment$.

$treatment$ **causes** $action\_occurred$
$no\_treatment$ **causes** $action\_occurred$
$u_2 \wedge action\_occurred$ **causes** $\neg alive$
$\neg u_2 \wedge action\_occurred$ **causes** $alive$

The probability of the inertial unknown variable u_2 can be expressed by $\mathcal{P}_2$ given as follows:

probability of u_2 **is** 0.5

The probability of the *occurrence* of $treatment$ and $no\_treatment$ is 0.5 each. (Our current language does not allow expression of such information. Although, it can be easily augmented, to accommodate such expressions, we do not do it here as it does not play a role in the analysis we are making.)

Assuming u_2 is independent of the occurrence of $treatment$ it is easy to see that the above modeling agrees with data table given earlier.

The observations $\mathcal{O}_2$ can be expressed as follows:

initially $\neg action\_occurred$ **initially** $alive$
$\neg alive$ **obs_after** $treatment$

We can now show that $\mathcal{D}_2 \cup \mathcal{P}_2 \cup \mathcal{O}_2 \not\models Q_2$, where Q_2 is the query: $alive$ **after** $no\_treatment$; and $\mathcal{D}_2 \cup \mathcal{P}_2 \cup \mathcal{O}_2 \models \neg alive$ **after** $no\_treatment$

Causal Model 2: The domain description $\mathcal{D}_3$ of the causal model 2 can be expressed as follows:

$treatment$ **causes** $\neg alive$ **if** u_2
$no\_treatment$ **causes** $\neg alive$ **if** $\neg u_2$

The probabilities of unknown variables ($\mathcal{P}_3$) is same as given in $\mathcal{P}_2$. The probability of occurrence of $treatment$ and $no\_treatment$ remains 0.5 each. Assuming u_2 is independent of the occurrence of $treatment$ it is easy to see that the above modeling agrees with data table given earlier.

The observations $\mathcal{O}_3$ can be expressed as follows:

initially $alive$ $\neg alive$ **obs_after** $treatment$

Unlike in case of the causal model 1, we can now show that $\mathcal{D}_3 \cup \mathcal{P}_3 \cup \mathcal{O}_3 \models Q_2$.

The state transition vs the n-state transition

Normally an MDP representation of probabilistic effect of actions is about the n-states. In this section we analyze the transition between n-states due to actions and the impact of observations on these transitions.

The transition function between n-states

As defined in (0.9) the transition probability $P_a(\mathsf{s}'|\mathsf{s})$ has either the value zero or is uniform among the s' where it is non-zero. This is counter to our intuition where we expect the transition function to be more stochastic. This can be explained by considering n-states and defining transition functions with respect to them.

Let s be a n-state. We can then define $\Phi_n(a,s)$ as:

$$\Phi_n(a,s) = \{\ s' \ \mid\ \exists \mathsf{s},\mathsf{s}' : (\mathsf{s} \models s) \wedge (\mathsf{s}' \models s') \wedge \mathsf{s}' \in \Phi(a,\mathsf{s})\ \}.$$

We can then define a more stochastic transition probability $P_a(s'|s)$ where s and s' are n-states as follows:

$$P_a(s'|s) = \sum_{\mathsf{s}_i \models s} \left(\frac{P(\mathsf{s}_i)}{P(s)} \sum_{\mathsf{s}'_j \models s'} P_a(\mathsf{s}'_j|\mathsf{s}_i)\right) \qquad (0.17)$$

The above also follows from (0.16) by having φ describing s', $\alpha = a$ and $\mathcal{O}$ expressing that the initial state satisfies s.

Impact of observations on the transition function

Observations have no impact on the transition function $\Phi(a,\mathsf{s})$ or on $P_a(\mathsf{s}'|\mathsf{s})$. But they do affect $\Phi(a,s)$ and $P_a(s'|s)$. Let us analyze why.

Intuitively, observations may tell us about the unknown variables. This additional information is monotonic in the sense that since actions do not affect the unknown variables there value remains unchanged. Thus, in presence of observations $\mathcal{O}$, we can define $\Phi_{\mathcal{O}}(a,s)$ as follows:

$\Phi_{\mathcal{O}}(a,s) = \{s' : s'$ is the interpretation of the fluents of a state in $\bigcup_{\mathsf{s} \models s \& \mathsf{s} \models \mathcal{O}} \Phi(a,\mathsf{s})\}$

As evident from the above definition, as we have more and more observations the transition function $\Phi_{\mathcal{O}}(a,s)$ becomes more deterministic. On the other hand, as we mentioned earlier the function $\Phi(a,\mathsf{s})$ is not affected by observations. Thus, we can accurately represent two different kind of non-deterministic effects of actions: the effect on states, and the effect on n-states.

Extending PAL to reason with narratives

We now discuss ways to extend PAL to allow actual observations instead of hypothetical ones. For this we extend PAL to incorporate narratives (Miller & Shanahan 1994), where we have time points as first class citizens and we can observe fluent values and action occurrences at these time points and do tasks such as reason about missing action occurrences, make diagnosis, plan from the current time point, and counter-factual reasoning about fluent values if a different sequence of actions had happened in a past (not just initial situation) time point. Here, we give a quick overview of this extension of PAL which we will refer to as $PALN$.

$PALN$ has a richer observation language $PALN_O$ consisting of propositions of the following forms:

$$\varphi \ \textbf{at} \ \ t \qquad (0.18)$$

$$\alpha \ \textbf{between} \ \ t_1, t_2 \qquad (0.19)$$

$$\alpha \ \textbf{occur\_at} \ \ t \qquad (0.20)$$

$$t_1 \ \textbf{precedes} \ \ t_2 \qquad (0.21)$$

where φ is a fluent formula, α is a (possibly empty) sequence of actions, and t, t_1, t_2 are time points (also called situation constants) which differ from the current time point t_C.

A narrative is a pair $(\mathcal{D}, \mathcal{O}')$, where $\mathcal{D}$ is a domain description and $\mathcal{O}'$ is a set of observations of the form (0.18-0.21). Observations are interpreted with respect to a domain description. While a domain description defines a transition function that characterize what states may be reached when an action is executed in a state, a narrative consisting of a domain description together with a set of observations defines the possible histories of the system. This characterization is done by a function Σ that maps time points to action sequences, and a sequence Ψ, which is a finite trajectory of the form $\mathsf{s}_0, a_1, \mathsf{s}_1, a_2, \ldots, a_n, \mathsf{s}_n$ in which $\mathsf{s}_0, \ldots, \mathsf{s}_n$ are states, $a_1, \ldots, a_n$ are actions and $\mathsf{s}_i \in \Phi(a_i, \mathsf{s}_{i-1})$ for $i = 1, \ldots, n$. Models of a narrative $(\mathcal{D}, \mathcal{O}')$ are interpretations $\mathcal{M} = (\Psi, \Sigma)$ that satisfy all the facts in $\mathcal{O}'$ and minimize unobserved action occurrences. (A more formal definition is given in (Baral, Gelfond, & Provetti 1997).) A narrative is *consistent* if it has a model. Otherwise, it is *inconsistent*. When $\mathcal{M}$ is a model of a narrative $(\mathcal{D}, \mathcal{O}')$ we write $(\mathcal{D}, \mathcal{O}') \models \mathcal{M}$.

Next we define the conditional probability that a particular pair $\mathcal{M} = (\Psi, \Sigma) = ([\mathsf{s}_0, a_1, \mathsf{s}_1, a_2, \ldots, a_n, \mathsf{s}_n], \Sigma)$ of trajectories and time point assignments is a model of a given domain description $\mathcal{D}$, and a set of observations. For that we first define the weight of a $\mathcal{M}$ (with respect to $\mathcal{D}$ which is understood from the context) denoted by $Weight(\mathcal{M})$ as:

$$\begin{aligned} Weight(\mathcal{M}) \ &= 0 \text{ if } \Sigma(t_C) \neq [a_1, \ldots, a_n]; and \\ &= P(\mathsf{s}_0) \times P_{a_1}(\mathsf{s}_1|\mathsf{s}_0) \times \ldots \times P_{a_n}(\mathsf{s}_n|\mathsf{s}_{n-1}) \\ &\ \text{otherwise.} \end{aligned}$$

Given a set of observation $\mathcal{O}'$, we then define

$$\begin{aligned} P(\mathcal{M}|\mathcal{O}') \ &= 0 \text{ if } \mathcal{M} \text{ is not a model of } (\mathcal{D}, \mathcal{O}'); \\ &= \frac{Weight(\mathcal{M})}{\sum_{(\mathcal{D},\mathcal{O}') \models \mathcal{M}'} Weight(\mathcal{M}\prime)} \text{ otherwise.} \end{aligned}$$

The probabilistic correctness of a plan from a time point t with respect to a model $\mathcal{M}$ can then be defined as

$$\begin{aligned} P(\varphi \ \textbf{after} \ \ \alpha \ \textbf{at} \ t|\mathcal{M}) \ &= \textstyle\sum_{\mathsf{s}' \in \Phi([\beta],\mathsf{s}_0) \wedge \mathsf{s}' \models \varphi} P_{[\beta]}(\mathsf{s}'|\mathsf{s}_0) \\ &\ \text{where } \beta = \Sigma(t) \circ \alpha \end{aligned}$$

Finally, we define the (probabilistic) correctness of a (multi-action) plan from a time point t given a set of observations. This corresponds to counter-factual queries of Pearl (Pearl 2000) when the observations are about a different sequence of actions than the one in the hypothetical plan.

$$\begin{aligned} P(\varphi \ \textbf{after} \ \ \alpha \ \textbf{at} \ t|\mathcal{O}') \ &= \textstyle\sum_{(\mathcal{D},\mathcal{O}') \models \mathcal{M}} P(\mathcal{M}|\mathcal{O}') \\ &\times P(\varphi \ \textbf{after} \ \ \alpha \ \textbf{at} \ t|\mathcal{M}) \end{aligned}$$

One major application of the last equation is that it can be used for action based diagnosis (Baral, McIlraith, & Son 2000), by having φ as $ab(c)$, where c is a component. Due to lack of space we do not further elaborate here.

Related work, Conclusion and Future work

In this paper we showed how to integrate probabilistic reasoning into 'reasoning about actions'. the key idea behind our formulation is the use of two kinds of unknown variables: inertial and non-inertial. The inertial unknown variables are similar to the unknown variables used by Pearl. The non-inertial unknown variables plays a similar role as the role of nature's action in Reiter's formulation (Chapter 12 of (Reiter 2001)) and are also similar to Lin's magic predicate in (Lin 1996). In Reiter's formulation a stochastic action is composed of a set of deterministic actions, and when an agent executes the stochastic action nature steps in and picks one of the component actions respecting certain probabilities. So if the same stochastic action is executed multiple times in a row an observation after the first execution does not add information about what the nature will pick the next time the stochastic action is executed. In a sense the nature's pick in our formulation is driven by a non-inertial unknown variable. We are still investigating if Reiter's formulation has a counterpart to our inertial unknown variables.

Earlier we mentioned the representation languages for probabilistic planning and the fact that their focus is not from the point of view of elaboration tolerance. We would like to add that even if we consider the Dynamic Bayes net representations as suggested by Boutilier and Goldszmidt, our approach is more general as we allow cycles in the causal laws, and by definition they are prohibited in Bayes nets.

Among the future directions, we believe that our formulation can be used in adding probabilistic concepts to other action based formulations (such as, diagnosis, and agent control), and execution languages. Earlier we gave the basic definitions for extending PAL to allow narratives. This is a first step in formulating action-based diagnosis with probabilities. Since our work was inspired by Pearl's work we now present a more detailed comparison between the two.

Comparison with Pearls' notion of causality

Among the differences betweens his and our approaches are:

(**1**) Pearl represents causal relationships in the form of deterministic, functional equations of the form $v_i = f_i(pa_i, u_i)$, with $pa_i \subset U \cup V \setminus \{v_i\}$, and $u_i \in U$, where U is the set of unknown variables and V is the set of fluents. Such equations are only defined for v_i's from V.

In our formulation instead of using such equations we use static causal laws of the form (0.2), and restrict ψ to fluent formulas. I.e., it does not contain unknown variables. A set of such static causal laws define functional equations which are embedded inside the semantics. The advantage of using such causal laws over the equations used by Pearl is the ease with which we can add new static causal laws. We just add them and let the semantics take care of the rest. (This is one manifestation of the notion of 'elaboration tolerance'.) On the other hand Pearl would have to replace his older equation by a new equation. Moreover, if we did not restrict ψ to be a formula of only fluents, we could have written $v_i = f_i(pa_i, u_i)$ as the static causal law $true$ **causes** $v_i = f_i(pa_i, u_i)$.

(**2**) We see one major problem with the way Pearl reasons about actions (which he calls 'interventions') in his formulation. To reason about the intervention which assigns a particular value v to a fluent f, he proposes to modify the original causal model by removing the link between f and its parents (i.e., just assigning v to f by completely forgetting the structural equation for f), and then reasoning with the modified model. This is fine in itself, except that if we need to reason about a sequence of actions, one of which may change values of the predecessors of f (in the original model) that may affect the value of f. Pearl's formulation will not allow us to do that, as the link between f and its predecessors has been removed when reasoning about the first action.

Since actions are first class citizens in our language we do not have such a problem. In addition, we are able to reason about executability of actions, and formulate indirect qualification, where static causal laws force an action to be inexecutable in certain states. In Pearl's formulation, all interventions are always possible.

Acknowledgment This research was supported by the grants NSF 0070463 and NASA NCC2-1232.

References

Baral, C.; Gelfond, M.; and Provetti, A. 1997. Representing Actions: Laws, Observations and Hypothesis. *Journal of Logic Programming* 31(1-3):201–243.

Baral, C.; McIlraith, S.; and Son, T. 2000. Formulating diagnostic problem solving using an action language with narratives and sensing. In *KR 2000*, 311–322.

Boutilier, C., and Goldszmidt, M. 1996. The frame problem and bayesian network action representations. In *Proc. of CSCSI-96*.

Boutilier, C.; Dean, T.; and Hanks, S. 1995. Planning under uncertainty: Structural assumptions and computational leverage. In *Proc. 3rd European Workshop on Planning (EWSP'95)*.

Gelfond, M., and Lifschitz, V. 1993. Representing actions and change by logic programs. *Journal of Logic Programming* 17(2,3,4):301–323.

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

Lin, F. 1996. Embracing causality in specifying the indeterminate effects of actions. In AAAI 96.

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

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

McCarthy, J. 1998. Elaboration tolerance. In *Common Sense 98*.

Miller, R., and Shanahan, M. 1994. Narratives in the situation calculus. *Journal of Logic and Computation* 4(5):513–530.

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

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

Poole, D. 1997. The independent choice logic for modelling multiple agents under uncertainty. *Artificial Intelligence* 94(1-2):7–56.

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

Sandewall, E. 1998. Special issue. *Electronic Transactions on Artificial Intelligence* 2(3-4):159–330. http://www.ep.liu.se/ej/etai/.