A Model for Reasoning About Persistence and Causation

Thomas Dean and Keiji Kanazawa
Brown University Department of Computer Science
Providence, RI 02912, U.S.A.

Technical Report No. CS-89-04
February 1989

A Model for Reasoning About Persistence and Causation

Thomas Dean* and Keiji Kanazawa
Department of Computer Science
Brown University
Box 1910, Providence, RI 02912
401-863-7600
tld@cs.brown.edu and kgk@cs.brown.edu

Abstract

Reasoning about change requires predicting how long a proposition, having become true, will continue to be so. Lacking perfect knowledge, an agent may be constrained to believe that a proposition persists indefinitely simply because there is no way for the agent to infer a contravening proposition with certainty. In this paper, we describe a model of causal reasoning that accounts for knowledge concerning cause-and-effect relationships and knowledge concerning the tendency for propositions to persist or not as a function of time passing. Our model has a natural encoding in the form of a network representation for probabilistic models. We explore the computational properties of our model by considering recent advances in computing the consequences of models encoded in this network representation. Finally, we consider how our probabilistic model addresses certain classical problems in temporal reasoning (*e.g.,* the frame and qualification problems).

Keywords: temporal reasoning, causal reasoning, uncertainty, probabilistic models, probabilistic inference, belief networks

*This work was supported in part by the National Science Foundation under grant IRI-8612644 and by DARPA/AFOSR under contract F49620-88-C-0132DEF.

1 Introduction

The common-sense law of inertia (McCarthy 1986) states that a proposition once made true remains so until something makes it false. Given perfect knowledge of initial conditions and a complete predictive model, the law of inertia is sufficient for accurately inferring the persistence of effects. In most circumstances, however, our predictive models and our knowledge of initial conditions are less than perfect. The law of inertia requires that, in order to infer that a proposition ceases to be true, we must predict an event with a contravening effect. Such predictions are often difficult to make. Consider the following examples:

- a cat is sleeping on the couch in your living room
- you leave your umbrella on the 8:15 commuter train
- a client on the telephone is asked to hold

In each case, there is some proposition initially observed to be true, and the task is to determine if it will be true at some later time. The cat may sleep undisturbed for an hour or more, but it is extremely unlikely to remain in the same spot for more than six hours. Your umbrella will probably not be sitting on the seat when you catch the train the next morning. The client will probably hold for a few minutes, but only the most determined of clients will be on the line after 15 minutes. Sometimes we can make more accurate predictions (*e.g.,* a large barking dog runs into the living room), but, lacking specific evidence, we would like past experience to provide an estimate of how long certain propositions are likely to persist.

Events precipitate change in the world, and it is our knowledge of events that enables us to make useful predictions about the future. For any proposition P that can hold in a situation, there are some number of general sorts of events (referred to as *event types*) that can affect P (*i.e.,* make P true or false). For any particular situation, there are some number of specific events (referred to as *event instances*) that occur. Let O correspond to the set of events that occur at time t, A correspond to that subset of O that affect P, $K(O)$ that subset of O known to occur at time t, and $K(A)$ that subset of A whose type is known to affect P. Figure 1 illustrates how these sets might relate to one another in a specific situation. In many cases, $K(O) \cap K(A)$ will be empty while A is not, and it may still be

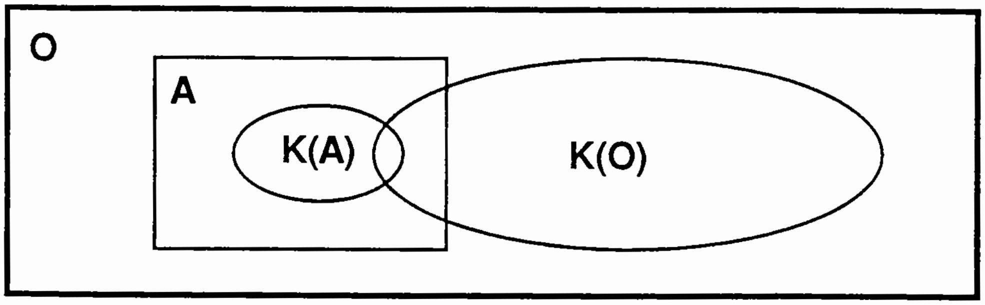

Figure 1: Events precipitate change in the world

possible to provide a reasonable assessment of whether or not P is true at t. In this paper, we provide an account of how such assessments can be made probabilistically.

2 Prediction and Persistence

In the following, we distinguish between two kinds of propositions: propositions, traditionally referred to as *fluents* (McCarthy and Hayes 1969), which, if they become true, tend to persist without additional effort, and propositions, corresponding to the occurrence of events, which, if true at a point, tend to precipitate or trigger change in the world. Let $holds(P,t)$ indicate that the fluent P is true at time t, and $occurs(E,t)$ indicate that an event of type E occurs at time t. We use the notation E_P to indicate an event corresponding to the fluent P becoming true.

Given our characterization of fluents as propositions that tend to persist, whether or not P is true at some time t may depend upon whether or not it was true at $t - \Delta$, where $\Delta > 0$. We can represent this dependency as follows:

$$\begin{aligned} p(holds(P,t)) &= p(holds(P,t) \mid holds(P,t-\Delta))p(holds(P,t-\Delta)) + \\ &\quad\ p(holds(P,t) \mid \neg holds(P,t-\Delta))p(\neg holds(P,t-\Delta)) \end{aligned} \tag{1}$$

where $\neg holds(P,t) \equiv holds(\neg P,t)$.

The conditional probability $p(holds(P,t) \mid holds(P,t-\Delta))$ is referred to as a ***survivor*** function in classical queuing theory (Syski 1979). Survivor functions capture the tendency of propositions to become false as a consequence of events with contravening effects; one

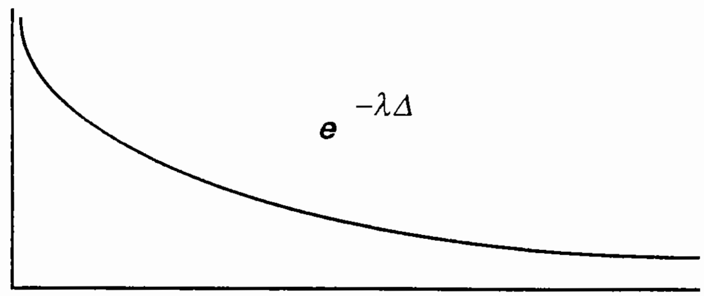

Figure 2: A survivor function with exponential decay

needn't be aware of a specific instance of an event with a contravening effect in order to predict that P will cease being true. As an example of a survivor function,

$$p(\mathit{holds}(P,t) \mid \mathit{holds}(P,t-\Delta)) = e^{-\lambda\Delta} \tag{2}$$

indicates that the probability that P persists drops off as a function of the time since P was last observed to be true at an exponential rate determined by λ (Figure 2). It is possible to efficiently construct an appropriate survivor function by tracking P over many instances of P becoming true (Dean and Kanazawa 1987). Referring back to Figure 1, survivor functions account for that subset of A corresponding to events that make P false, assuming that $K(A) = \{\}$.

If we have evidence concerning specific events known to affect P (*i.e.*, $K(A) \cap K(O) \neq \{\}$), (1) is inadequate. As an interesting special case of how to deal with events known to affect P, suppose that we know about all events that make P true (*i.e.*, we know $p(\mathit{occurs}(E_P,t))$ for any value of t), and none of the events that make P false. In particular, suppose that P corresponds to John being at the airport, and E_P corresponds to the arrival of John's flight. We're interested in whether or not John will still be waiting at the airport when we arrive to pick him up. Let $e^{-\lambda\Delta}$ represent John's tendency to hang around airports, where λ is a measure of his impatience. If $f(t) = p(\mathit{occurs}(E_P,t))$, then we can compute the probability of P being true at t by convolving f with $e^{-\lambda\Delta}$ as in

$$p(\mathit{holds}(P,t)) = \int_{-\infty}^{t} f(z)e^{-\lambda(t-z)}dz. \tag{3}$$

A shortcoming of (3) is that it fails to account for evidence concerning specific events known to make P false. Suppose, for instance, that $E_{\neg P}$ corresponds to Fred meeting John

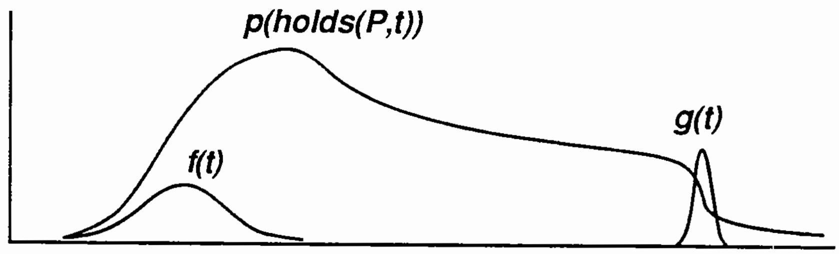

Figure 3: Probabilistic predictions

at the airport and giving him a ride to his hotel. If $g(t) = p(occurs(E_{\neg P}, t))$, then

$$p(holds(P,t)) = \int_{-\infty}^{t} f(z) e^{-\lambda(t-z)} \left[1 - \int_{z}^{t} g(x)dx \right] dz \tag{4}$$

is a good approximation in certain cases. Figure 3 illustrates the sort of inference licensed by (4).

There are some potential problems with equation (4). The survivor function $e^{-\lambda\Delta}$ was meant to account for all events that make P false, but (4) counts one such event, John leaving the airport with Fred, twice: once in the survivor function and once in g. In certain cases, this can lead to significant errors (*e.g.,* Fred always picks up John at the airport). To combine the available evidence correctly, it will help if we distinguish the different sorts of knowledge that might be brought to bear on estimating whether or not P is true. We will also reinterpret the event type E_P to mean an event *known* to make P true. The following equation makes the necessary distinctions and indicates how the evidence should be combined[1]:

[1] In order to justify our use of the generalized addition law in (5), we assume that $p(occurs(E_P, t) \wedge occurs(E_{\neg P}, t)) = 0$.

$$
\begin{aligned}
&p(\mathit{holds}(P,t)) = && (5)\\
&\quad p(\mathit{holds}(P,t) \mid \mathit{holds}(P,t-\Delta) \wedge \neg(\mathit{occurs}(E_P,t) \vee \mathit{occurs}(E_{\neg P},t))) && \text{(N1)}\\
&\qquad * \; p(\mathit{holds}(P,t-\Delta) \wedge \neg(\mathit{occurs}(E_P,t) \vee \mathit{occurs}(E_{\neg P},t)))\\
&\; + \; p(\mathit{holds}(P,t) \mid \mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_P,t)) && \text{(N2)}\\
&\qquad * \; p(\mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_P,t))\\
&\; + \; p(\mathit{holds}(P,t) \mid \mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_{\neg P},t)) && \text{(N3)}\\
&\qquad * \; p(\mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_{\neg P},t))\\
&\; + \; p(\mathit{holds}(P,t) \mid \neg\mathit{holds}(P,t-\Delta) \wedge \neg(\mathit{occurs}(E_P,t) \vee \mathit{occurs}(E_{\neg P},t))) && \text{(N4)}\\
&\qquad * \; p(\neg\mathit{holds}(P,t-\Delta) \wedge \neg(\mathit{occurs}(E_P,t) \vee \mathit{occurs}(E_{\neg P},t)))\\
&\; + \; p(\mathit{holds}(P,t) \mid \neg\mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_P,t)) && \text{(N5)}\\
&\qquad * \; p(\neg\mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_P,t))\\
&\; + \; p(\mathit{holds}(P,t) \mid \neg\mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_{\neg P},t)) && \text{(N6)}\\
&\qquad * \; p(\neg\mathit{holds}(P,t-\Delta) \wedge \mathit{occurs}(E_{\neg P},t))
\end{aligned}
$$

Consider the contribution of the individual terms corresponding to the conditional probabilities labeled N1 through N6 in (5). N1 accounts for *natural attrition*: the tendency for propositions to become false given no direct evidence of events known to affect P. N2 and N5 account for *causal accretion*: accumulating evidence for P due to events known to make P true. N2 and N5 are generally 1. N3 and N6, on the other hand, are generally 0, since evidence of $\neg P$ becoming true does little to convince us that P is true. Finally, N4 accounts for *spontaneous causation*: the tendency for propositions to suddenly become true with no direct evidence of events known to affect P.

By using a discrete approximation of time and fixing Δ, it is possible both to acquire the necessary values for the terms N1 through N6 and to use them in making useful predictions (Dean & Kanazawa 1987). If time is represented as the integers, and $\Delta = 1$, we note that the law of inertia applies in those situations in which the terms N1, N2, and N5 are always 1 and the other terms are always 0. In the rest of this paper, we assume that time is discrete and linear and that the time separating any two consecutive time points is some constant δ. Only evidence concerning events *known* to make P true is brought to bear on $p(\mathit{holds}(E_P,t))$. If $p(\mathit{holds}(E_P,t))$ were used to summarize all evidence concerning events that make P true, then N1 would be 1.

3 Reasoning About Causation

Before we consider the issues involved in making predictions using knowledge concerning N1 through N6, we need to add to our theory some means of predicting additional events. We consider the case of one event causing another event. Deterministic theories of causation often use implication to model cause-and-effect relationships. For instance, to indicate that the occurrence of an event of type E_1 at time t causes the occurrence of an event of type E_2 following t by some $\delta > 0$ just in case the conjunction $P_1 \wedge P_2 \ldots \wedge P_n$ holds at t, we might write

$$(holds(P_1 \wedge P_2 \ldots \wedge P_n, t) \wedge occurs(E_1, t)) \supset occurs(E_2, t + \delta).$$

If the caused event is of a type E_P, this is often referred to as *persistence causation* (McDermott 1982). In our model, we use the conditional probability

$$p(occurs(E_2, t + \delta) \mid holds(P_1 \wedge P_2 \ldots \wedge P_n, t) \ \wedge \ occurs(E_1, t)) = \pi$$

to indicate that, if an event of type E_1 occurs at time t, and P_1 through P_n are true at t, then an event of type E_2 will occur following t by some $\delta > 0$ with probability π.

In moving to a probabilistic model of causation, there are complications that we must consider. Consider, for example, the two rules:

$$(holds(P, t) \wedge occurs(E, t)) \supset occurs(E_R, t + \delta) \text{ and}$$

$$(holds(P \wedge Q, t) \wedge occurs(E, t)) \supset occurs(E_R, t + \delta).$$

These two rules pose no problems for the deterministic theory of causation, since P and Q are either true or false, and the rules either apply or they don't. In fact, the second rule is redundant. However, in a probabilistic model, P and Q are usually not unambiguously true or false. Therefore, in the probabilistic causal theory consisting of

$$p(occurs(E_R, t + \delta) \mid holds(P, t) \wedge occurs(E, t)) = \pi_1$$

$$p(occurs(E_R, t + \delta) \mid holds(P \wedge Q, t) \wedge occurs(E, t)) = \pi_2$$

the second rule can no longer be considered redundant. In fact, since the second rule is more specific than the first, it provides us with valuable additional information. Note that although the second rule provides additional information, it only tells us how P is affected by Q being true. A complete account of how P is affected by Q would also include the information

$$p(occurs(E_R, t+\delta) \mid holds(P \wedge \neg Q, t) \wedge occurs(E, t)) = \pi_3.$$

Such a complete account of how a proposition is affected by other propositions is important for efficient computation with probabilistic models, as we shall see in the sections below.

4 A Model for Reasoning About Change

In this section, we take a slight modification of equation (5) as the basis for a model of persistence. Equation (5) predicts $holds(P, t)$ on the basis of $holds(P, t - \Delta)$, $occurs(E_P, t)$, and $occurs(E_{\neg P}, t)$, where Δ is allowed to vary. In the model presented in this section, we only consider pairs of consecutive time points, t and $t + \delta$, and arrange things so that the value of a fluent at time t is completely determined by the state of the world at δ in the past. In equation (5), we interpret events of type E_P occurring at t as providing evidence for P being true at t. In our new model, we interpret events of type E_P occurring at t as providing evidence for P being true at $t + \delta$. This reinterpretation is not strictly necessary, but we prefer it since the expressiveness of the resulting models can easily be characterized in terms of the properties of Markov processes. In our new model, we predict $A \equiv holds(P, t + \delta)$ by conditioning on

$$\begin{aligned} C_1 &\equiv holds(P, t) \\ C_2 &\equiv occurs(E_P, t) \\ C_3 &\equiv occurs(E_{\neg P}, t) \end{aligned}$$

and specify a complete model for the persistence of P as

$$p(A) = \sum p(A \mid C_1 \wedge C_2 \wedge C_3)\ p(C_1 \wedge C_2 \wedge C_3) \tag{6}$$

where the sum is over the eight possible truth assignments for the variables C_1, C_2, and C_3. Note that this model requires that we have probabilities of the form $p(A \mid C_1 \wedge C_2 \wedge C_3)$ and $p(C_1 \wedge C_2 \wedge C_3)$ for all possible valuations of the C_i.

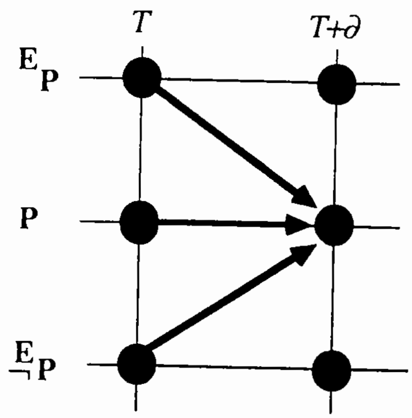

Figure 4: The evidence for P at time $T + \delta$

In the following, we will make use of a graphical representation for probabilistic models that will serve to clearly indicate the assumptions concerning dependence and independence underlying our models. The graphical representations that we will be using have been called *Bayes networks* (Pearl 1988), *belief networks* (Duda *et al.*1981), and *influence diagrams* (Shachter 1986). We will use the generic term *probability network* to refer to a network that satisfies the following basic properties common to all three of the above representations. A probability network represents the variables or propositions of a probabilistic theory as nodes in a graph. The variables in our networks correspond to propositional variables of the form $holds(P,T)$ and $occurs(E,T)$. Dependence between two variables is indicated by a directed arc between the two nodes associated with the variables.

Because dependence is always indicated by an arc, probability networks make it easy to identify the conditional independence inherent in a model simply by inspecting the graph. Two nodes which are linked via a common neighbor, but for which there are no other connecting paths are conditionally independent given the common node. For instance, in the model described in this section, $holds(P, t - \delta)$ is independent of $holds(P, t + \delta)$ given $holds(P, t)$; in this particular case, this corresponds to a Markov property. Probability networks make it easy to construct and verify the correctness and reasonableness of a model directly in terms of the corresponding graphical representation. Our model for persistence can be represented by the network shown in Figure 4. As soon as we provide a model for causation, we will show how this simple model for persistence can be embedded in a more complex model for reasoning about change over time.

Generally, we expect that the cause-and-effect relations involving E_P will be specified in terms of constraints of the form:

$$\begin{aligned}
p(occurs(E_P, t+\delta) \mid occurs(E_1, t) \wedge holds(Q_1, t)) &= \pi_1 \\
p(occurs(E_P, t+\delta) \mid occurs(E_2, t) \wedge holds(Q_2, t)) &= \pi_2 \\
&\ldots \\
p(occurs(E_P, t+\delta) \mid occurs(E_n, t) \wedge holds(Q_n, t)) &= \pi_n
\end{aligned}$$

However, to specify a complete model, we will need some more information. To predict $A \equiv occurs(E_P, t+\delta)$, we condition on

$$\begin{aligned}
C_1 &\equiv occurs(E_1, t) \wedge holds(Q_1, t)) \\
C_2 &\equiv occurs(E_2, t) \wedge holds(Q_2, t)) \\
&\ldots \\
C_n &\equiv occurs(E_n, t) \wedge holds(Q_n, t))
\end{aligned}$$

and specify a complete model as:

$$p(A) = \sum p(A \mid C_1 \wedge C_2 \wedge \ldots \wedge C_n)\ p(C_1 \wedge C_2 \wedge \ldots \wedge C_n)$$

Note that we need on the order of 2^n probabilities corresponding to the 2^n possible valuations of the propositional variables C_1 through C_n to specify this model. The associated probability network is shown in Figure 5. Similar networks would be constructed for event types other than those involving propositions becoming true or false.

Now we can construct a complete model for reasoning about change over time. Figure 6 illustrates the probability network for such a complete model. For each propositional variable of the form $holds(P,T)$ or $occurs(E,T)$, there is a node in the probability network. The arcs are specified according to the isolated models for persistence and causation illustrated in Figure 4 and Figure 5. Following Pearl (1988), we can write down the unique distribution corresponding to the model shown in Figure 6 as

$$p(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} p(x_i|S_i)p(S_i)$$

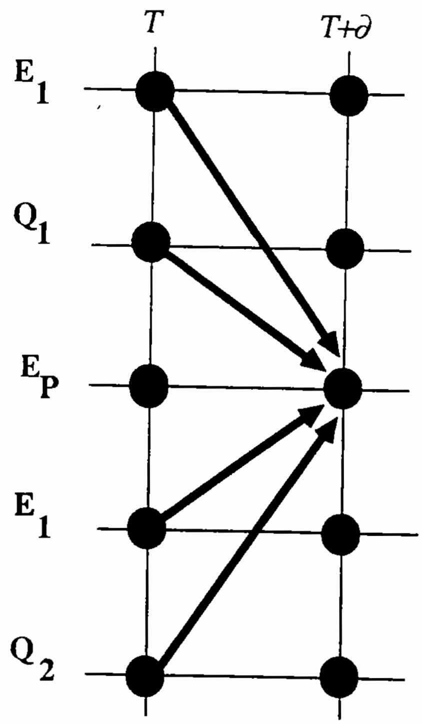

Figure 5: The evidence for E_P at time $T + \delta$

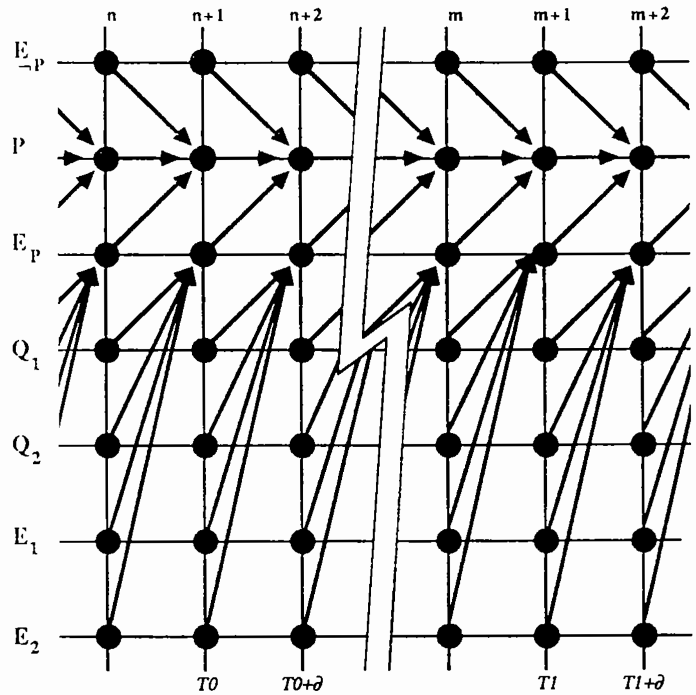

Figure 6: A Bayesian inference network for reasoning about time

where the x_i denote the propositional variables in the model, and S_i is the conjunction of the propositional variables associated with those nodes for which there exist arcs to x_i in the network. How we compute this distribution is the subject of Section 5.

It is straightforward to extend the model described above to account for new observations and updating beliefs. Suppose we have the observations $\sigma_1, \sigma_2, \ldots, \sigma_n$, where each observation is of the form $occurs(O, t)$. We assume some prior distribution specified in terms of constraints of the form:

$$p(occurs(O, t)) = 0.001$$

where O is an event type corresponding to a particular type of observation. There are also constraints indicating prior belief regarding the occurrence of events other than observations. For instance, we might have

$$p(occurs(E, t)) = 0.001.$$

Observations are related to events by constraints such as

$$p(occurs(E,t) \mid occurs(O,t)) = 0.70 \text{ and}$$

$$p(occurs(E,t) \mid \neg occurs(O,t)) = 0.025.$$

To update an agent's beliefs you can either change the priors:

$$p(occurs(O,t)) = 1.0$$

or you can compute the posterior distribution:

$$BEL(A) = p(A \mid \sigma_1, \sigma_2, \ldots, \sigma_n).$$

Most of the standard techniques for representing and reasoning about evidence in probability networks apply directly to our model.

This paper is primarily concerned with presenting a particular model for reasoning about change. Our current research involves applying this model to problems in robotics and exploring the expressive limitations of our model. While we have only begun the latter research, a word about expressive limitations is probably in order to give the reader some idea of the potential power of our model and to encourage analysis by others. Most of the initial work in analyzing the expressive power of our model has been concerned with relating the predictive power of our model to that of Markov processes and in particular Markov chains.

Suppose that the instantaneous state of the world can be completely specified in terms of a vector of values assigned to a finite set of boolean variables $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$, and suppose further that the environment can be accurately modeled as a Markov process in which time is discrete and the state space Ω corresponds to all possible valuations of the variables in $\mathcal{P}$. Given such a model including a transition matrix defined on Ω, we can generate a probability network that will enable us to compute the probability of any proposition in $\mathcal{P}$ being true at any time t given evidence concerning the values of variables in $\mathcal{P}$ at various times, and do so in accord with with transition probabilities specified in the Markov model. It would seem likely (though we have no proof as yet) that, given a model of the sort described in this section, one could construct a corresponding Markov model that would encode the same information. In fact, we conjecture that the class of models described in this section, or some restriction thereof, is equivalent in expressive power to the class of models defined as Markov chains.

The reason that one might use the probability network model rather than an equivalent Markov model is twofold. First, the probability network is designed to simplify computing the answers to questions that one generally needs for applications in planning and decision support: namely, answers to questions of the form, "What is the probability of P at t given everything else we know about the situation?" These same answers can be computed using the Markov model, but the process is somewhat less direct. Second, the probability network provides an encoding that, in most cases, is considerably more compact than the $O(2^{2n})$, $n = |\mathcal{P}|$, sized matrix of transition probabilities required for the Markov model. The models that we conjecture are equivalent to Markov models specify how each proposition at time t relates to each proposition at time $t - \delta$. This information is then replicated as many times as we have time points that we are interested in reasoning about. If m is the maximum number of incoming arcs for any node in the probability network, then the size of the resulting probability network, including all of the associated probabilities necessary to complete the model, is $O(2^m kn)$, where k is the number of time points and $n = |\mathcal{P}|$. In the next section, we will consider how hard it is to answer questions regarding the probability of P at t given a particular probability network.

5 Computing Probabilistic Predictions

In this section, we consider the computational issues concerned with our model of temporal reasoning. We will chiefly be concerned with computing the joint conditional probability distribution for a fully specified probability network, given the available information. The problem of computing the distribution for a partial, incomplete model will be considered briefly at the end of this section.

For networks with restricted topologies, there are efficient methods for computing the joint distribution. A graph is said to be *singly-connected* if there are no undirected cycles in the graph. In the case of probability networks, the underlying directed acyclic graph is singly-connected if there is at most one path linking any two nodes in the graph. In other words, the class of singly-connected graphs are basically those which form tree-like topologies; both proper trees and trees in which a node may have more than one parent (called a *generalized Chow tree* (Chow and Liu 1968)). Kim and Pearl (1983) have described an efficient mechanism for computing the joint distribution for singly-connected networks by local propagation of Bayes factors (*i.e.*, the likelihood ratios $p(x)/p(\neg x)$).

The Kim-Pearl algorithm is linear in the diameter of the network. However, it is restricted to singly-connected networks. Unfortunately, most interesting causal theories, such as those arising in medicine, and our own temporal models involve *multiply-connected* networks; networks which contain cycles. The general problem of probabilistic inference in probability networks, including multiply-connected networks, has been shown to be NP-hard (Cooper 1987). Although this points towards some possible problems in terms of computational efficiency, it should not discourage us from using probability networks for probabilistic inference. The NP-hardness result tells us that a search for an efficient exact algorithm for all cases is probably misguided, but it says little about what to expect in practice. In general, asymptotic complexity results should serve as a guide in algorithm design, not as a justification for abandoning a research area. Such results tell us to look for algorithms that have good expected-case performance and to consider possible tradeoffs in terms of efficiency, soundness, or completeness.

There have been three principal types of algorithms proposed for probabilistic inference with general probability networks; *exact*, *bounding*, and *stochastic* algorithms. The exact algorithms attempt to exploit the topology of the network; they include the method of *conditioning* (Pearl 1988), and the Lauritzen-Spiegelhalter (L-S) algorithm (Lauritzen and Spiegelhalter 1988). In a multiply-connected network, if a node which has multiple children is instantiated to a given value, then the values of the children become independent of one another. The path(s) connecting the children through the parent node become "blocked", and the connectivity of the graph changes. If enough nodes are instantiated, then the resulting network becomes singly-connected, so that the Bayes propagation algorithm can be used. A set of nodes which, when instantiated, renders a multiply-connected network singly-connected corresponds to a cutset of the underlying graph. Conditioning computes a weighted average of the joint conditional probability distributions for all possible instantiations of such a cutset (for further details, see (Pearl 1988)).

In the L-S algorithm, a set of subsidiary arcs are added to the graph to *triangulate* the graph, and the joint distribution can be efficiently computed in terms of the cliques of the original graph. A graph is triangulated if there are no cycles of length 4 or more without a chord or shortcut. L-S represents the network in terms of a clique tree, a tree of the cliques in the graph. Computing in terms of the tree is very efficient, as with the Kim-Pearl algorithm; the L-S algorithm alters the connectivity of a probability network much like conditioning does, rendering a multiply-connected network singly-connected so that an efficient Kim-Pearl type propagation algorithm may be employed. An important characteristic of the L-S algorithm is that after a relatively time consuming initialization step for a given network topology, it is capable of performing Bayesian updating in the network very efficiently.

Since conditioning and L-S compute exact results, they are directly affected by the NP-hard results; both exhibit exponential behavior in the worst case[2]. Conditioning involves the computation of the minimum cutset which is an NP-hard problem, and L-S is exponential in the size of the largest clique in the graph.

In light of the worst-case exponential behavior of the exact algorithms, it is worthwhile to consider how the inference task may be relaxed in order to gain efficiency. Both bounding algorithms and approximation algorithms relax the exactness criterion, trading precise answers for speed. By a bounding algorithm, we mean an algorithm that exploits the structure of a network and the constraints to assign some rough bounds on each node. By an approximation algorithm, we mean a Monte Carlo simulation algorithm. The distinction between the two types of algorithms is somewhat arbitrary, in that it is possible to have an approximation algorithm which in a sense bounds its answer by an error margin. Both types of algorithms are able to supply a rough answer in a very short time, and the reliability of the answers increases with the amount of time spent computing the answer. The main difference of the two types of algorithms is that a bounding algorithm always makes sound, albeit initially very weak, inferences, whereas a approximation algorithm may take some time before the numbers it generates make some sense. Although neither type of algorithm supplies a good answer right away, they can provide some rough answers fairly quickly. Because of this, using these algorithms can have significant advantage in situations where the time spent in computing is important (Cooper 1984; Dean and Boddy 1988; Horvitz 1988).

Bounding algorithms look at the constraints and bound the distribution, typically by supplying an upper and lower bound (Horvitz 1988; Henrion 1988b). The bounds are

[2]Although they both have exponential worst-case behavior, experiments conducted by other researchers (Chavez 1988) indicate that in practice, L-S is much faster; in some examples of medical networks, it is reported to offer about a ten-fold speedup.

successively refined such that they approach the correct distribution in the limit. These algorithms appear promising, but more research is needed to determine their properties.

Approximation algorithms simulate the states that a probability network is likely to go through, given a set of constraints. By a state of the network, we mean an assignment of a value to each of the variables in the network. For example, given a prior probability constraint that $p(A = 1) = 0.8$ for some variable A, we would, with the use of an oracle, assign $A = 1$ in 80% of the simulated states of the network. The aim in approximation is to develop an algorithm which comes ϵ close to the correct answer almost all the time (i.e. with high probability). As noted earlier, most often, the algorithms perform iterative refinement of the approximations, such that the error is some inversely monotonic function of the time spent in computing the answer. Monte Carlo algorithms have a solid basis in the physical sciences (Metropolis *et al.*1953). Unfortunately, the Monte Carlo algorithms developed to date for probability networks (Henrion 1988a; Pearl 1987) have been somewhat less than ideal; there are no proofs with regard to convergence (we cannot say anything about when the computed distribution is within ϵ of the correct distribution, or how confident we can be that it is within that range), and in practice convergence is often slow. Furthermore, they exhibit pathological behavior in the case of certain network topologies (Chin and Cooper 1987).

Recent research (Chavez 1988) indicates that it may be possible to use results in the theory of compututation to the problem of computing with probability networks. Jerrum and Sinclair (1988) describe an approximation method for a well-known combinatorially hard problem, that of computing the permanent of a matrix. Their method hinges on the use of a *fully-polynomial random approximation scheme* or FPRAS for short, which they define as follows. Let f be a function from input strings to natural numbers. An FPRAS for f is a probabilistic algorithm which, when presented with a string x and a real number $\epsilon > 0$, runs in polynomial time in $|x|$ and $1/\epsilon$ and outputs a number which with high probability estimates $f(x)$ to within a factor of $(1 + \epsilon)$. The FPRAS for computing the permanent involves a Markov chain whose states map onto approximations of the permanent. The transition probabilities are determined by the particular matrix that is being considered. A matrix with sufficient density has the property that it is *rapidly mixing*; that is, the chain converges rapidly to a uniform distribution no matter what the starting state is. Therefore, simulating the Markov chain from some random starting state for a short period of time results in the chain being in a state which with high probability closely approximates the permanent. Researchers are currently investigating the application of the same technique to the probabilistic inference task.

The algorithms considered to this point have assumed the existence of a complete causal model. When only a partial causal model is available, we have only a subset of the conditional probabilities that are required for the probability network algorithms. Therefore, in order to perform the joint conditional probability computations, it is necessary, explicitly or implicitly, to fill out the missing parts of the model. Essentially, we are required to guess what the rest of the causal model looks like. Clearly, we must avoid violating the constraints that already exist, but in general, there will be many distributions that satisfy the constraints. We would like to choose the distribution that makes the fewest assumptions about the underlying space as possible. In other words, we need some method that makes the least presumptive inferences given a set of constraints.

It has been argued that a probability distribution with the maximum entropy meets such a requirement (Jaynes 1979). Roughly speaking, entropy is an inverse measure of information; therefore, maximization of entropy is equivalent to minimization of information. Since it contains the least information out of all the possible distributions, the maximum entropy distribution must be the one which assumes the least; precisely what we desire. The maximum entropy distribution is equivalent to the Gibbs and Boltzmann distributions that have been so successfully applied in the context of statistical physics. It has very clean theoretical properties, and it is in many respects the optimal Bayesian estimate (Hinton and Sejnowski 1983). In (Dean and Kanazawa 1988), we describe a method based on *stochastic relaxation* (Geman and Geman 1984; Lippman 1986) for computing the maximum entropy distribution for a probability model. Unfortunately, only loose bounds have been proved for convergence of stochastic relaxation, that are far too slow in practice.

Alternative methods for computing with an incomplete model actually include a subclass of the bounding algorithms that were previously mentioned. Although in the previous discussion, we were mostly concerned with their complexity, it turns out that we can define the bounding heuristics to apply to partially specified models. Cooper (1984) describes such a method in his thesis.

6 Fundamental Problems in Temporal Reasoning

Given that our model addresses many of the same problems that concern logicians working on temporal logic, we will briefly mention how our model deals with certain classic problems in temporal reasoning: the frame, ramification, and qualification problems. We will begin by considering the frame problem stated in probabilistic terms: "Does our model accurately capture our expectations regarding fluents that are considered *not* likely to change as a consequence of a particular event occurring?" The answer is yes insofar as frame axioms can be said to solve the frame problem in temporal logic; persistence constraints are the probabilistic equivalent of frame axioms. In considering the ramification problem, we will consider two possible interpretations. First, "Does our model enable us to compute appropriate expectations regarding the value of a particular fluent at a particular point in time without bothering with a myriad of seemingly unimportant consequences?" The answer to this is a resounding no; our model commits us to predicting every possible consequence of every possible action no matter how implausible. A second interpretation (or perhaps facet is a better word) of the ramification problem is "Does our model enable us to handle additional consequences that follow from a set of causal predictions?" For instance, if A is in box B and I move B to a new location, I should be able to predict that A will be in the new location along with B. Our model provides no provision at all for this sort of reasoning. The basic idea of Bayesian inference can be extended to handle this sort of reasoning, but we have not investigated this to date. The last problem we consider concerns reasoning about exceptions involving the rules governing cause-and-effect relationships. Does our model solve the qualification problem? That is to say, "Does our model accurately capture our expectations regarding the possible exceptions to knowledge about cause-and-effect relationships?" The answer is yes; conditional probabilities would seem to be exactly suited for this sort of reasoning. It should be noted, however, that our model imposes a considerable burden on the person setting up the model. The model described in this paper requires specifying all possible causes for each possible effect and the probability of each effect for every possible combination of possible causes. It is not clear, however, that one can get away with less. Given the problems inherent in eliciting such information from experts, it would appear that we will have to automate the process of setting up our probabilistic models.

7 Conclusions

This paper presents a model of temporal reasoning suited for situations involving incomplete knowledge. By expressing knowledge of cause-and-effect relations in terms of conditional probabilities, we are able to make appropriate judgements concerning the persistence of propositions. By directly encoding expectations regarding how long certain propositions are likely to persist, we are able to reason about phenomena for which we do not possess an accurate cause-and-effect model as well as phenomena for which we do possess an accurate model but about which it is either difficult or impossible to acquire evidence due to sensing and real-time processing constraints. Our model can be described in terms of a network representation for probabilistic theories that has recently received a great deal of attention in the literature, and for which there now exist a number of computational techniques that appear to be quite promising. We expect that the development of approximate methods for computing the expectations engendered by a given probability network will enable us to apply our methods to solve a number of problems in robotic control and decision support that require reasoning about time and change.

References

R. MARTIN CHAVEZ. 1988. Personal communication.

HOWARD CHIN AND GREGORY F. COOPER. 1987. Stochastic simulation of bayesian belief networks. In Proceedings of the 1987 Workshop on Uncertainty in Artificial Intelligence, Seattle, Washington.

C. CHOW AND C. LIU. 1968. Approximating discrete probability distributions with dependence trees. IEEE Transactions on Information Theory.

GREGORY F. COOPER. 1984. NESTOR: A computer-based medical diagnostic aid that integrates causal and probabilistic knowledge. PhD thesis, Stanford University.

GREGORY F. COOPER. 1987. Probabilistic inference using belief networks is np-hard. Stanford Knowledge Systems Laboratory, Memo KSL-87-27.

THOMAS DEAN AND MARK BODDY. 1988. An analysis of time-dependent planning. In Proceedings AAAI-88, St. Paul, Minnesota, pp. 49–54.

THOMAS DEAN AND KEIJI KANAZAWA. 1987. Persistence and probabilistic inference. Brown University Department of Computer Science, Technical Report CS-87-23.

THOMAS DEAN AND KEIJI KANAZAWA. 1988. Probabilistic temporal reasoning. In Proceedings AAAI-88, St. Paul, Minnesota, pp. 524–528.

R.O. DUDA, P.E. HART, AND NILSSON N.J. 1981. Subjective bayesian methods for rule-based inference systems. In B.W. Webber and N.J. Nilsson, editors, Readings in Artificial Intelligence. Tioga, Palo Alto, CA.

STUART GEMAN AND DONALD GEMAN. 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, **6**, pp. 721–741.

MAX HENRION. 1988. Propagating uncertainty by logic sampling in bayes' networks. In John F. Lemmer and Laveen F. Kanal, editors, Uncertainty in Artificial Intelligence 2. North-Holland, pp. 149–163.

MAX HENRION. 1988. Towards efficient probabilistic diagnosis in multiply connected belief networks. In Proceedings of the Conference on Influence Diagrams.

G.E. HINTON AND T.J. SEJNOWSKI. 1983. Optimal perceptual inference. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 448–453.

ERIC J. HORVITZ. 1988. Reasoning under varying and uncertain resource constraints. In Proceedings AAAI-88, St. Paul, Minnesota, pp. 111–116.

EDWIN T. JAYNES. 1979. Where do we stand on maximum entropy. In Levin and Tribus, editors, The Maximum Entropy Formalism. MIT Press, pp. 15–118.

MARK JERRUM AND ALISTAIR SINCLAIR. 1988. Conductance and the rapid mixing property for markov chains: the approximation of the permanent resolved (preliminary version). In ACM Symposium on Theoretical Computing, pp. 235–244.

JIM H. KIM AND JUDEA PEARL. 1983. A computational model for causal and diagnostic reasoning in inference systems. In Proceedings of IJCAI 8, Karlsruhe, West Germany.

STEPHEN L. LAURITZEN AND DAVID J. SPIEGELHALTER. 1988. Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society Series B, **50**, pp. 157–194.

ALAN F. LIPPMAN. 1986. A Maximum Entropy Method for Expert System Construction. PhD thesis, Brown University.

JOHN MCCARTHY AND PATRICK J. HAYES. 1969. Some philosophical problems from the standpoint of artificial intelligence. Machine Intelligence, 4.

JOHN MCCARTHY. 1986. Applications of circumscription to formalizing commonsense knowledge. Artificial Intelligence, **28**, pp. 89–116.

DREW V. MCDERMOTT. 1982. A temporal logic for reasoning about processes and plans. Cognitive Science, **6**, pp. 101–155.

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AUGUSTA H. TELLER, AND EDWARD TELLER. 1953. Equation of state calculations by fast computing machines. The Journal of Chemical Physics, **21**, pp. 1087–1092.

JUDEA PEARL. 1987. Evidential reasoning using stochastic simulation of causal models. Artificial Intelligence, **32**, pp. 245–257.

JUDEA PEARL. 1988. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan-Kaufman, Los Altos, California.

ROSS D. SHACHTER. 1986. Evaluating influence diagrams. Operations Research, **34**, pp. 871–882.

RYSZARD SYSKI. 1979. Random Processes, Marcel Dekker, New York.

List of Figures