Annu. Rev. Comput. Sci. 1990. 4:37–72

REASONING UNDER UNCERTAINTY

Judea Pearl

Department of Computer Science, University of California, Los Angeles, California 90024

CONTENTS

1. INTRODUCTION 37
1.1 *Overview* 37
1.2 *Why Bother With Uncertainty?* 39
1.3 *Why Is It Hard?* 39
1.4 *Three Approaches to Uncertainty* 40
1.5 *Extensional vs Intensional Approaches* 41
2. EXTENSIONAL SYSTEMS: MERITS, DEFICIENCIES, AND REMEDIES 42
2.1 *Computational Merits* 42
2.2 *Semantic Deficiencies* 43
3. INTENSIONAL SYSTEMS AND NETWORK REPRESENTATIONS 49
3.1 *Evidential Reasoning with Bayesian Networks* 50
3.2 *Dempster-Shafer Theory and Constraint Networks* 58
4. LESSONS AND OPEN ISSUES 65
4.1 *Relations to Nonmonotonic Logic* 65
4.2 *Graphoids and the Formalization of Relevance* 68

1. INTRODUCTION

1.1 *Overview*

One can hardly identify a field in artificial intelligence (AI) that doesn't use some sort of uncertain reasoning, namely, processes leading from evidence or clues to guesses and conclusions under conditions of partial information. Many powerful programs have been written that embody practical solutions to various aspects of reasoning with uncertainty. These include MYCIN (Shortliffe 1976), INTERNIST (Miller et al 1982), PROSPECTOR (Duda et al 1976), MEDAS (Ben-Bassat et al 1980), RUM (Bonissone et al 1987), MUM (Cohen et al 1987a), MDX (Chandrasakaran & Mittal 1983), and MUNIN (Andreassen et al 1987). This survey focuses

8756–7016/90/1115–0037$02.00

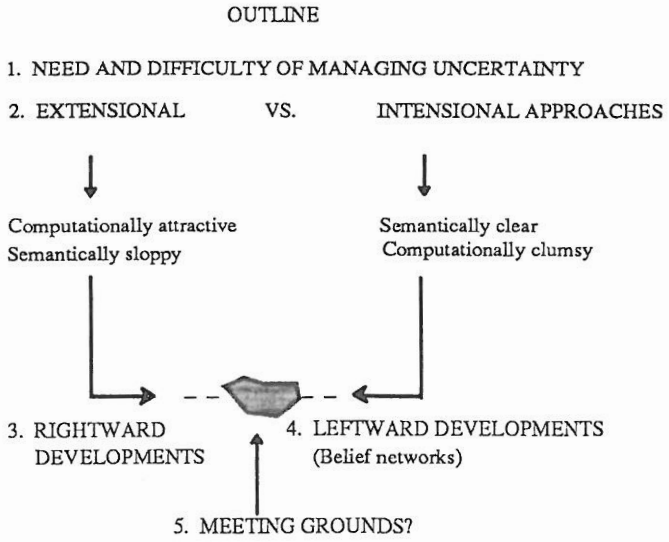

OUTLINE
1. NEED AND DIFFICULTY OF MANAGING UNCERTAINTY
2. EXTENSIONAL
VS.
INTENSIONAL APPROACHES
Computationally attractive
Semantically sloppy
Semantically clear
Computationally clumsy
3. RIGHTWARD DEVELOPMENTS
4. LEFTWARD DEVELOPMENTS (Belief networks)
5. MEETING GROUNDS?

Figure 1 Outline of survey and relationships between extensional and intensional approaches to uncertainty.

on a select set of issues, trends, and principles that have emerged from these past works. I hope to describe these in a unifying perspective and in greater depth than a more general survey would permit. For broader surveys, the reader is referred to Thompson (1985), Prade (1983), Stephanou & Sage (1987), and the works collected in Kanal & Lemmer (1986). Expanded technical treatments of the topics discussed in this review can be found in Pearl (1988a).

A summary of this paper is shown in Figure 1. I first discuss the general necessities and difficulties of managing uncertainty and then talk about two diametrically opposed approaches to the problem, one called *extensional*, the other *intensional*.[1] The extensional approach, also known as production systems, rule-based systems, or procedure-based systems, treats uncertainty as a generalized truth value attached to formulas and, following the tradition of classical logic, computes the uncertainty of any formula from the uncertainties of its subformulas. It is characterized by computationally attractive features, but is semantically sloppy. In the intensional approach, also known as the declarative or model-based

[1] This terminology is that of Perez & Jirousek (1985); the terms *syntactic* vs *semantic* are also adequate.

approach, uncertainty is attached to "states of affairs" or subsets of "possible worlds." It is semantically clear but computationally clumsy. The trade-off between semantic clarity and computational efficiency has been the main issue of concern in past research and has transcended notational boundaries.

Naturally, attempts have been made to rectify the deficiencies of both approaches. I briefly discuss (Section 2) efforts to improve the semantic clarity of extensional approaches. I then emphasize attempts to improve the computational efficiency of intensional approaches (Section 3). In this vein, I discuss the central role of *belief networks* representations, both the Bayesian type and the Dempster-Shafer type.

Finally, I speculate (Section 4) on the middle ground toward which the two approaches will hopefully converge in the next few years. This arena, I believe, will involve the issues of encoding context-dependent information, the formalization of relevance, and network decomposition techniques.

1.2 *Why Bother With Uncertainty?*

Reasoning about any realistic domain always requires that simplifications be made. By necessity, we leave many facts unknown, unsaid, or crudely summarized. For example, most rules used to encode knowledge and behavior have exceptions that one cannot afford to enumerate, and the situations in which the rules apply are usually ambiguously defined or hard to satisfy precisely in real life. Reasoning with exceptions is like navigating through a mine field; most steps are safe but some can be devastating. If we know its location, each mine can be avoided or diffused; but suppose that we must start our journey with a map the size of a postcard, lacking room to mark down the locations of the mines or to note how they are wired together. An alternative to the extremes of ignoring or enumerating exceptions is to *summarize* them—i.e. to indicate which areas of the minefield are more dangerous than others. Such summarization is essential if we wish to find a reasonable compromise between safety and speed of movement. Thus, the art of reasoning under uncertainty amounts to that of representing and processing summaries of exceptions.

1.3 *Why Is It Hard?*

One way of summarizing exceptions is to assign to propositions numerical measures that combine according to uniform syntactic principles the way truth values combine in logic. Adopted by first-generation expert systems, this approach often yielded unpredictable and counterintuitive results (see below). As a matter of fact, it is remarkable that this combination strategy went as far as it did, in view of the fact that uncertainty measures stand for something totally different from truth values. While truth values in logic characterize the formulas under discussion, uncertainty measures characterize exceptions—i.e. the facts *not* shown in the formulas. Accordingly, while the syntax of the formula is a perfect guide for combining the visibles, it is close to useless for combining the invisibles. For example, the machinery of Boolean algebra gives us no clue about how the exceptions to $A \rightarrow C$ interact with those of $B \rightarrow C$ to yield the exceptions to $(A \wedge B) \rightarrow C$. These invisible exceptions may interact in intricate and clandestine ways, as a result of which we lose most of the computationally attractive features of classical logic—e.g. modularity and monotonicity.

Although in logic, too, formulas interact in intricate ways, the interactions are visible. This enables us to calculate the impact of each new fact in stages by a process of derivation that resembles the propagation of a wave: We first compute the impact of the new fact on a set of syntactically related sentences, S_1, store the results, then propagate the impact from S_1 to another set of sentences, S_2, and so on, without having to come back and redo S_1. Unfortunately, this computational scheme, so common to logical deduction, cannot be justified under uncertainty unless one makes restrictive assumptions, which, in probabilistic terms, amount to *conditional independence*.

Another feature we lose in going from logic to shaded uncertainties is *incrementality*. We would like to account for the impact of each of several items of evidence individually: compute the effect of the first item, then attend to the next, absorb its added impact, and so on. This, too, can only be done after making restrictive assumptions of independence. Thus it appears that uncertainty reasoning represents a hopeless case of having to compute the impact of the entire set of past observations upon the entire set of sentences in one global step. This, of course, is an impossible task.

1.4 *Three Approaches to Uncertainty*

AI researchers tackling these problems can be classified into three schools, which I will call the logicist, neo-calculist, and neo-probabilist. The logicist school attempts to deal with uncertainty using nonnumerical techniques. The neo-calculist school uses numerical representations of uncertainty but, believing that probability calculus is inadequate for the task, invents entirely new calculi such as the Dempster-Shafer calculus, fuzzy logic, certainty factors, etc. Finally, the neo-probabilists remain within the traditional framework of probability theory while attempting to equip the theory with computational facilities needed to perform AI tasks. This taxonomy however, is superficial as it captures the notational rather than the semantical differences among the various approaches. A more fundamental taxonomy can be drawn along the dimensions shown in Figure 1, namely the extensional vs the intensional approaches. For example, it is possible to use probabilities either extensionally [e.g. in PROSPECTOR (Duda et al 1976)] or intensionally [e.g. in MUNIN (Andreassen et al 1987)]. Similarly, one can use the Dempster-Shafer notation either extensionally [as in Ginsberg (1984)] or intensionally [as in Lowrance et al (1986)].

1.5 *Extensional vs Intensional Approaches*

1.5.1 THE ROLE OF CONNECTIVES Extensional systems, a typical representative of which is the certainty-factors calculus used in MYCIN (Shortliffe 1976), treat uncertainty as a generalized truth value. The certainty of a formula is defined as a unique function of the certainties of its subformulas. Thus the connectives in the formula serve to select the appropriate weight-combining function. For example, the certainty of the conjunction $A \wedge B$ is given by some function (e.g. the minimum, or the product) of the certainty measures assigned to A and B individually. By contrast, in intensional systems, a typical representative of which is probability theory, certainty measures are assigned to sets of worlds and the connectives, too, combine sets of worlds by set-theoretical operations. For example, the probability of $P(A \wedge B)$ is given by the weight assigned to the intersection of two sets of worlds, those in which A is true and those in which B is true, but cannot be determined from the individual probabilities $P(A)$ and $P(B)$.

1.5.2 WHAT'S IN A RULE? Rules, too, have different roles in these two systems. The rules in extensional systems provide licenses for certain symbolic activities. For example, the rule $A \rightarrow B(m)$ may mean: If you see A, then you have the license to update the certainty of B by a certain amount that is a function of the rule strength m. The rules are interpreted as a summary of the past performance of the problem solver, describing the way an agent normally reacts to problem situations or to items of evidence. In intensional systems, the rules denote elastic constraints about the world. For example, in the Dempster-Shafer formalism the rule $A \rightarrow B(m)$ does not describe how an agent reacts to the finding of A but asserts that the set of worlds in which A and $\neg B$ hold simultaneously is rather unlikely and hence should be excluded with probability m. In the Bayesian formalism the rule $A \rightarrow B(m)$ is interpreted as a conditional probability statement $P(B|A) = m$, asserting that among all worlds satisfying A, those that also satisfy B constitute a proportion of size m. Although there exists a vast difference between these two interpretations (as is shown below in Sections 3.2.2 and 4.1.1), they both represent summaries of factual or empirical information rather than summaries of past decisions.

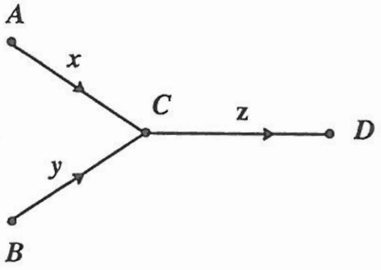
2. EXTENSIONAL SYSTEMS: MERITS, DEFICIENCIES, AND REMEDIES

2.1 *Computational Merits*

A good way to present the computational merits of extensional systems is to examine the way rules are handled in the certainty-factors formalism (Shortliffe 1976) and contrast it with that dictated by probability theory. Figure 2 depicts the combination functions that apply to series and parallel rules, from which one can form a rule network. The result is a modular procedure of determining the certainty factor of a conclusion, given the credibility of each rule, and the certainty factor of the premises (i.e. the roots of the network). To complete the calculus we must also define combining functions for conjunctions and negation. Setting mathematical details aside, the point to notice is that the same combination function applies uniformly to all rules in the system, regardless of what other rules might be in the neighborhood.

Rules:

- **If A then C (x)**
- **If B then C (y)**
- **If C then D (z)**



1. Parallel combination

$$CF(C) = \begin{cases} x + y - xy & x, y > 0 \\ (x + y) / (1 - \min(x, y)) & x, y \text{ different sign} \\ x + y + xy & x, y < 0 \end{cases}$$

2. Series combination

$$CF(D) = z \cdot \max(0, CF(C))$$

3. Conjunction, negation ...

Figure 2 Functions combining certainty factors in EMYCIN—an extensional system.

Computationally speaking, this uniformity mirrors the modularity of inference rules in classical logic. For example, the logical rule "if A then B" has the following procedural interpretation: "If you see A anywhere in the knowledge base, then, regardless of other things the knowledge base contains, and regardless of how A was derived, you have the license to assert B and add it to the database." This combination of *locality* ("regardless of other things") and *detachment* ("regardless of how it was derived") constitutes the principle of *modularity*. The numerical parameters that decorate the combination functions in Figure 2 do not alter this basic principle. The computational license provided by the rule $A \rightarrow B(m)$ reads: "If you see the certainty of A undergoing a change δ_A, then, regardless of other things the knowledge base contains, and regardless of how δ_A was triggered, you have an unqualified license to modify the current certainty of B by some amount, δ_B, which may depend on m, δ_A, and on the current certainty of B.[2]

To appreciate the power of this interpretation, let us compare it with that given by an intensional formalism such as probability theory. Interpreting rules as conditional probability statements, $P(B|A) = p$, does not provide us with a license to do anything. Even if we are fortunate to find A true in the database, we still cannot assert a thing about B or $P(B)$, because the meaning of the statement is: "If A is true, and A is the only thing that you know, then you can attach to B a probability p." As soon as we have other facts, K, in the database, the license to assert $P(B) = p$ is automatically revoked, and we need to look up $P(B|A, K)$ instead. Therefore, the conditional probability statement leaves one totally impotent, unable to initiate any computational activity, unless one can verify that all the other things in the knowledge base are irrelevant. It is for this reason that verification of irrelevancy is so crucial in intensional systems.

In truth, such verifications are also crucial in extensional systems, but the computational convenience of these systems and their striking resemblance to logical derivations tempt people to neglect the importance of verifying irrelevancy. I next describe the semantic penalties imposed when relevance considerations are ignored.

2.2 *Semantic Deficiencies*

The price tag attached to the computational advantages of extensional systems is that they often yield updating that is incoherent—i.e. subject to surprises and counterintuitive conclusions. These problems surface in several ways. The most notable are: 1. improper handling of bidirectional

[2] The observation that the rules refer to changes rather than absolute values was made by Horvitz & Heckerman (1986).

inferences, 2. difficulties in retracting conclusions, and 3. improper treatment of correlated sources of evidence.

2.2.1 THE ROLE OF BIDIRECTIONAL INFERENCES The ability to use both predictive and diagnostic information is an important component of plausible reasoning, and improper handling of such information leads to strange results. A common pattern of normal discourse is that of *abductive* reasoning: If A implies B, then finding that B is true makes A more credible (Polya 1954). This pattern involves reasoning both ways, from A to B, as well as from B to A. Moreover, it appears that people do not require two separate rules for performing these inferences; the first provides the license to invoke the second. Extensional systems, on the other hand, require that the second rule be stated explicitly and, what is more disturbing, that the first rule be removed. Otherwise, a cycle is created where any slight evidence in favor of A would be amplified via B and fed back to A, quickly turning into a stronger confirmation (of A and B), with no apparent factual justification. The prevailing practice in such systems (e.g. MYCIN) is to cut off cycles of that sort, permitting only diagnostic reasoning and no predictive inferences.

Cutting off its predictive component prevents the system from exhibiting another important pattern of plausible reasoning, one that we call "explaining away": If A implies B, and C implies B, and B is true, then finding that C is true makes A *less* credible. In other words finding a second explanation to an item of data makes the first explanation less credible. Such interaction among multiple causes appears in many applications. For example, when a physician discovers evidence in favor of one disease, this reduces the credibility of other diseases, although the patient may well be suffering from two or more disorders simultaneously. A suspect who provides an alternative explanation for being at the scene of the crime appears less likely to be guilty, even though the explanation furnished does not preclude his having committed the crime.

To exhibit this sort of reasoning, a system must use bidirectional inferences; from evidence to hypothesis (or explanation), as well as from hypothesis to evidence. While it is sometimes possible to use brute force (e.g. enumerating all exceptions) and restore "explaining away" without the dangers of circular reasoning, we shall see that any system that succeeds in doing this must sacrifice the principles of modularity—i.e. locality and detachment. More precisely, every system that updates beliefs modularly and uses natural rules is bound to behave in a manner contrary to prevailing patterns of plausible reasoning.

2.2.2 THE LIMITS OF MODULARITY The principle of locality attains its ultimate realization in the inference rules of classical logic. The rule "If P

then Q" means that if P is found true, we can assert Q with no further analysis, even if the database contains some other knowledge K. In plausible reasoning, the luxury of ignoring the rest of the database can no longer be maintained. For example, suppose we have a rule

R_1 = "If the ground is wet, then assume it rained (with certainty c_1)."

Validating the true of "The ground is wet" does not permit us to raise the certainty of "It rained" because the knowledge base might contain strange items such as K = "The sprinkler was on last night." These strange items, called *defeaters*, are sometimes easy to discover (as in the case of K' = "The neighbor's grass is dry," which directly opposes "It rained") but sometimes hide cleverly behind syntactical innocence. The neutral fact K = "The sprinkler was on" neither supports nor opposes the possibility of rain, yet K manages to undercut the rule R_1. This undercutting cannot be implemented in an extensional system; once R_1 is invoked, the increase in the certainty of "It rained" will never be retracted, because no rule would normally connect "The sprinkler was on" to "It rained." Imposing such a connection by proclaiming "The sprinkler was on" as an explicit exception to R_1 again defeats the spirit of modularity; it forces the rule-author to pack together items of information that are only remotely related to each other, and it burdens the rules with an unmanageably large number of exceptions.

Violation of detachment can also be demonstrated in this example. In deductive logic, if K implies P and P implies Q, then finding K true, permits us to deduce Q by simple chaining; a derived proposition (P) can trigger a rule with the same vigor as a directly observed proposition. However, chaining does not apply in plausible reasoning. The system may contain two innocent looking rules: "If wet-ground then rain," and "If sprinkler-on then wet-ground"; you find that the sprinkler is on and, obviously, you do not want to conclude that it rained. On the contrary, finding that the sprinkler is on only takes away support from "rain."

As another example, consider the relationships shown in Figure 3. Normally an alarm sound alerts us to the possibility of a burglary. If someone calls you at the office and tells you that your burglar alarm is ringing, you would surely rush home in a hurry, even though its ringing could have other causes. If you further hear a radio announcement that an earthquake occurred nearby, and if the last false alarm you recall was triggered by an earthquake, then your certainty of a burglary would diminish. Again, this requires going both ways, from effect to cause (radio → earthquake), cause to effect (earthquake → alarm), and then back from effect to cause (alarm → burglary). However, notice what pattern of reasoning results from such a chain: We have a rule "If A (alarm)

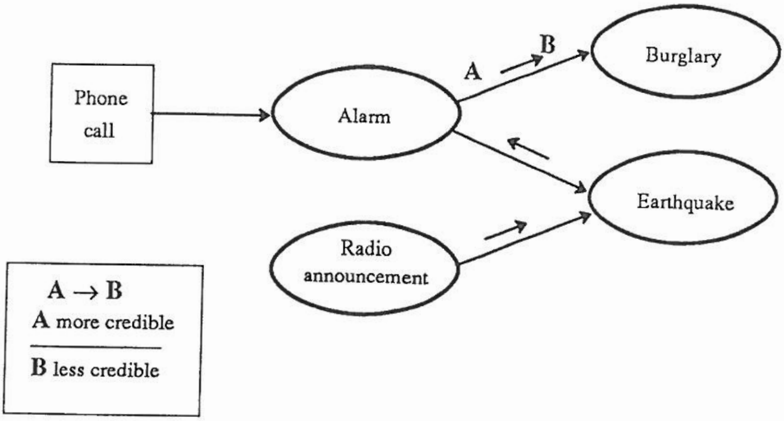


Figure 3 Making the antecedent of a rule more credible can cause the consequent to become less credible.

then *B* (burglary)." After you've listened to the radio, *A* becomes more credible and the conclusion *B* becomes less credible. Overall, we have: If *A* → *B* and *A* becomes more credible, then *B* becomes less credible. This behavior is clearly contrary to everything we expect from local belief updating.

In conclusion, the difficulties of summarizing exceptions do not stem from the nonnumeric, bi-value character of classical logic. Equally troublesome difficulties emerge when truth and certainty are measured on a grey scale, whether by a point estimate, by interval bounds, or by linguistic quantifiers such as "likely" or "credible." There seems to be a basic struggle between procedural modularity and semantic coherence, independent of the notational system used.

2.2.3 CORRELATED EVIDENCE Extensional systems, greedily exploiting the licenses provided by locality and detachment, respond only to the magnitudes of the weights and not to their origins. As a result they will produce the same conclusions regardless of whether the weights originate from identical or independent sources of information. An example from Henrion (1986b) about the Chernobyl disaster helps demonstrate the problems encountered by such a local strategy. Figure 4 shows how multiple, independent sources of evidence would normally increase the credibility of a hypothesis (e.g. "Thousands dead"), but the discovery that these sources have a common origin should reduce the credibility. Extensional systems are too local to recognize the common origin of the information,

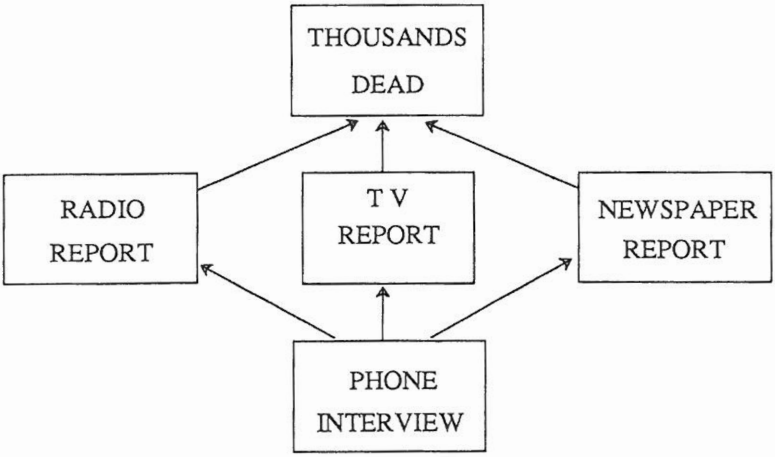


Figure 4 The Chernobyl disaster example (after Henrion) shows why rules cannot combine locally.

and they would update the credibility of the hypothesis as if it were supported by three independent sources.

2.2.4 ATTEMPTED REMEDIES AND THEIR LIMITATIONS The developers of extensional systems have proposed and implemented powerful techniques to remedy some of the semantic deficiencies we have discussed. The remedies, most of which focus on the issue of correlated evidence, take two approaches.

2.2.4.1 *Bounds propagation* Since most correlations are unknown, certainty measures are combined under two extreme assumptions: one, that the components are highly positively correlated, the other that they are negatively correlated. This gives rise to upper and lower bounds on the combined certainty, which are entered as inputs to subsequent computations, producing new bounds on the certainty of the conclusions. This approach has been implemented in INFERNO (Quinlan 1983) and represents a local approximation to Nilsson's probabilistic logic (Nilsson 1986).

2.2.4.2 *User-specified combination functions* A system named RUM (Bonissone et al 1987) permits the rule-author to specify the combination function that should apply to the rule's components. For example, if a, b, c stand for the weights assigned to propositions A, B, C, respectively, in the rule

$$A \wedge B \rightarrow C$$

the user can specify which of the following three combination functions should be used:

$$T_1(a, b) = \max(0, a+b-1)$$

$$T_2(a, b) = ab$$

$$T_3(a, b) = \min(a, b).$$

These functions (called *T norms*) represent the probabilistic combinations obtained under three extreme cases of correlation between A and B: highly negative, zero, and highly positive.

Cohen et al (1987b) have proposed a more refined scheme where, for any pair of values $P(A)$ and $P(B)$, the user is permitted to specify the value of the resulting probability, $P(C)$.

The difficulties with these correlation-handling remedies are several. First, the bounds produced by systems such as INFERNO are too wide. For example, if we are given $P(A) = p$ and $P(B|A) = q$ then the bounds we obtain for $P(B)$ are

$$pq \leq P(B) \leq 1-p(1-q)$$

which for small p approach the unit interval [0, 1]. Second, pair-wise correlations are generally not sufficient to handle the intricate dependencies that may occur among rules; higher-order dependencies are often necessary (Bundy 1985). Finally, even if one succeeds in specifying higher-order dependencies, a much more fundamental limitation exists: Dependencies are dynamic relationships that are created and destroyed as new evidence obtains. For example, the dependence between a child's shoe size and reading ability is destroyed once we find out the child's age. A dependency between the propositions "It rained last night" and "The sprinkler was on" is created once we find out that the ground is wet. Thus, correlations and combination functions specified at the knowledge-building phase may quickly become obsolete once the program is put into use.

Heckerman (1986a,b) delineated precisely the range of applicability of extensional systems of the MYCIN type. He proved that any system that updates certainty weights in a modular and consistent fashion can be given a probabilistic interpretation in which the certainty update of a proposition A is some function of the likelihood ratio

$$\lambda = \frac{P(Evidence|A)}{P(Evidence|\neg A)}.$$

In MYCIN, for example, the certainty update CF can be interpreted as

$$CF = \frac{\lambda - 1}{\lambda + 1}.$$

Once we have a probabilistic interpretation, it is easy to determine the set of structures within which the update procedure will be semantically valid. It turns out that a system of such rules will produce coherent update if and only if the rules form a directed tree—i.e. no two rules may diverge from the same premise. This limitation explains why strange results were obtained in the burglary example of Figure 3. There, the alarm event points to two possible explanations "burglary" and "earthquake," which amounts to two evidential rules diverging from the premise "alarm."

Hájek (1985) and Hájek & Valdes (1987) have developed an algebraic theory that characterizes an even wider range of the extensional systems and combining functions, including those based on Dempster-Shafer intervals. The unifying properties common to all such systems is that they form an ordered Abelian group. Again, the knowledge base must form a tree in order that no evidence is counted twice via alternative paths of reasoning.

3. INTENSIONAL SYSTEMS AND NETWORK REPRESENTATIONS

We have seen that handling uncertainties is not a trivial task but requires a fine balance between the requirements of modularity and coherence. In intensional systems, the syntax consists of declarative statements and, hence, mirrors world knowledge fairly nicely. For example, conditional probability statements, such as "If it rains the grass is likely to get wet," are both empirically testable and conceptually meaningful. Additionally, intensional systems have no problem handling bidirectional inferences and correlated evidence; these emerge as built-in features of one globally coherent model. However, since the syntax does not point to any useful procedures, we need to construct special mechanisms that convert the declarative input into query-answering routines.

A solution, or at least part of a solution, is offered by techniques based on *belief networks*. The idea is to make intensional systems operational by making relevance relationships explicit, thus curing the impotence of declarative statements such as $P(B|A) = p$. As mentioned earlier, the reason one cannot act on the basis of such declarations is that one must first make sure that other things contained in the knowledge base are irrelevant to B and hence can be ignored. The trick, therefore, is to encode knowledge in such a way that the ignorable is recognizable, or better yet that the unignorable is quickly identified and is readily accessible. Belief networks encode relevancies as neighboring nodes in a graph, thus ensuring that by consulting the neighborhood one gains a license to act; what you don't see locally doesn't matter. In effect, what network representations offer is a dynamically updated list of all currently valid licenses to ignore, and licenses to ignore constitute permissions to act.

Network representations are not foreign to AI systems. Most reasoning systems encode relevancies using intricate systems of pointers—i.e. networks of indexes that group facts into structures such as frames, scripts, causal chains, and inheritance hierarchies. These structures, while shunned by pure logicians, have proven indispensable in practice because they make the information required to perform an inference task reside "in the vicinity" of the propositions involved in the task. Indeed, many patterns of human reasoning can be explained only by people's tendency to follow the pathways laid out by such networks.

The special feature of the networks discussed in this review is that they have clear semantics. In other words, they are not auxiliary devices contrived to make reasoning more efficient but are an integral part of the semantics of the knowledge base, and most of their features can even be derived from the knowledge base (Pearl 1988a).

I first discuss the nature of these networks in two uncertainty formalisms: probability theory, where they are called *Bayesian networks*, *causal nets*, or *influence diagrams*, and the Dempster-Shafer theory, where they are referred to as *galleries* (Lowrance et al 1986), *qualitative Markov networks* (Shafer et al 1987), or *constraint networks* (Montanari 1974). In Section 4.1 I briefly discuss the theory of graphoids, which provides an axiomatic characterization of the notion of relevance and its relation to network representations.

3.1 *Evidential Reasoning with Bayesian Networks*

3.1.1 NETWORK CONSTRUCTION AND THE ROLE OF CAUSALITY

Formally, Bayesian networks are directed acyclic graphs in which each node represents a random variable, or uncertain quantity, that can take on two or more possible values. The arcs signify the existence of direct influences between the linked variables, and the strengths of these influences are quantified by forward conditional probabilities. Informally, the structure of a Bayesian network can be determined by a simple procedure: We assign a vertex to each variable in the domain and draw arrows toward each vertex X_i from a select set $\mathbf{S}_i$ of vertices perceived to be "direct causes" of X_i. The strength of these direct influences is then quantified by a link matrix $P(x_i|\mathbf{s}_i)$, which represents (judgmental estimates of) the conditional probabilities of the events $X_i = x_i$, given any value combination $\mathbf{s}_i$ of the parent set $\mathbf{S}_i$. The ensemble of these local estimates specifies a complete and consistent global model (i.e. a joint distribution function), on the basis of which all probabilistic queries can be answered. The overall joint distribution function on the variables $X_1, \ldots, X_n$, is given by the product:

$$P(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} P(x_i|\mathbf{s}_i).$$

So, for example, the joint distribution corresponding to the network of Figure 5 is given by:

$$P(h, e, r, s, d, w, g) = P(h)P(e)P(r|e)P(s|e, h)P(d|s)P(w|s)P(g|s)$$

where lower case symbols stand for any particular value (e.g. true or false) of their corresponding variables.

Conversely, the structure of the network can be determined by the joint distribution function, if such is ever available. Once we agree on a total order (e.g. temporal precedence) for the variables involved, the set of parents $\mathbf{S}_i$ of variable X_i is chosen from its predecessors by the criterion that

$$P(x_i|\mathbf{s}_i) = P(x_i|x_1, \ldots, x_{i-1}).$$

In other words, knowing the parents renders all other predecessors of X_i irrelevant relative to our belief in $X_i = x_i$. In principle, any choice $\mathbf{S}_i$ satisfying this criterion will define an adequate network but, of course, choosing minimal sets of parents will be more efficient, and ordering the

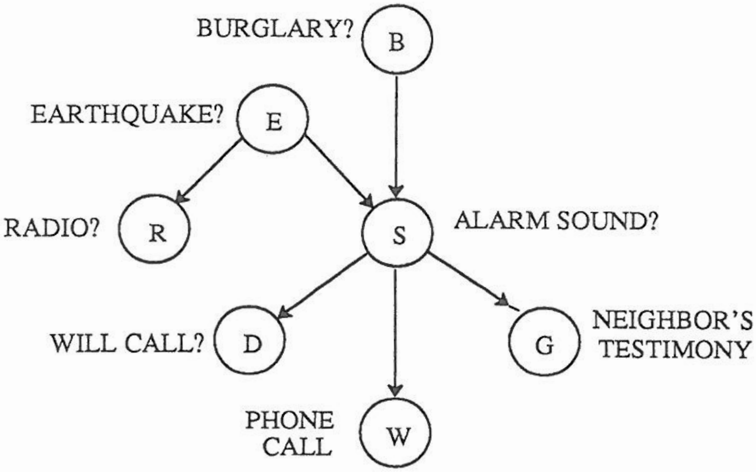


Figure 5 The Bayesian network associated with the burglary alarm story.

variable chronologically would normally result in sparser networks than otherwise.

Figure 5 depicts the burglary alarm story of Figure 3, with two added variables D and G. D describes the event that your daughter, having been surprised by the alarm, will try to reach you at the office. G stands for the testimony of another neighbor relative to the alarm sound S. The transition from Figure 3 to Figure 5 demonstrates the incremental nature of the process of constructing the knowledge base. Adding the facts about D only requires that one identifies the possible causes of D (in our case, S) and estimates two parameters:

$P(D|S)$ = How likely is it that your daughter will try to call, given that she hears the alarm sound, and

$P(D|\neg S)$ = How likely is it for her to call, assuming there is no alarm.

The addition of the link $S \rightarrow G$ requires similar parameters, except that if the testimony G is available [even if it is nonpropositional—say, a lengthy conversation (Pearl 1987b, 1988a)] it can be summarized by a single parameter; the likelihood ratio

$$\lambda = \frac{P(G|S)}{P(G|\neg S)}.$$

The advantage of a network representation is that it allows people to express directly the fundamental qualitative relationship of "direct dependency"; the network then displays a consistent set of many additional direct and indirect dependencies and preserves them as a stable part of the model, independent of the numerical estimates. For example, Figure 5 displays the fact that the radio report (R) would not change the prospects of the daughter's phone call (D), once we verify the actual state of the alarm system (S). This fact is conveyed via the network topology—showing S intercepting the path between R and D—despite the fact that it was not considered explicitly during the construction of the network. It can be inferred visually from the linkages used to put the network together and, moreover, will remain part of the model regardless of the numerical estimates that are assigned to the links.

The directionality of the arrows is essential for displaying nontransitive dependencies—e.g. S depends on both E and H yet E and H are independent; they become dependent only if S or any of its descendants is known. Had the arcs been stripped of their arrows, some of these relationships would be misrepresented. This role of identifying what information is or is not relevant in any given state of knowledge is the central feature of causal schemata. In this role, causality serves as a lubricant that modularizes our knowledge as it is cast from experience. By displaying the irrelevancies in the domain, causal schemata minimize the number of relationships that need to be considered while a model is constructed and, in effect, authorize many future local inferences. The prevailing practice in rule-based expert systems of encoding knowledge by evidential rules (i.e. if effect then cause) is deficient in this respect. It usually fails to account for induced dependencies between causes (e.g an earthquake explaining away the alarm sound); and if one ventures to encode these by direct rules, the number of rules becomes unmanageable (Shachter & Heckerman 1987).

There is a long and rich tradition in Bayesian belief networks, starting in 1921 with the work of geneticist Sewal Wright (1921). He developed a method called *path analysis* (Wright 1934) that later on became an established representation of causal models in economics (Wold 1964), sociology (Kenny 1979; Blalock 1971), and psychology (Duncan 1975). *Influence diagrams*, another component in this tradition (Howard & Matheson 1981; Shachter 1988), were developed for decision analysis and contain both event nodes and action nodes. Similar networks were called *recursive models* when used by statisticians seeking meaningful and effective decompositions of contingency tables (Lauritzen 1982; Wermuth & Lauritzen 1983; Kiiveri et al 1984).

The next subsection illustrates the role of networks as a representation capable of converting declarative knowledge to answer-producing procedures. The illustration focuses on Bayesian networks, but similar techniques have been developed for constraint networks in the Dempster-Shafer formalism (Shafer et al 1987; Kong 1986).

3.1.2 BELIEF PROPAGATION BY MESSAGE PASSING Since a fully specified Bayesian network constitutes a complete probabilistic model of all variables in the domain, it contains the information necessary to answer all probabilistic queries about these variables. Such queries include, for example, "What are the chances of a burglary, given that the radio announced an earthquake and the daughter did not call?" or "What is the most likely explanation for your daughter's not having called?" Additionally, owing to the relevance information conveyed by their links, belief networks can be used as inference engines—i.e. the nodes can be regarded as processors and the links as communication channels that provide the (storage locations of the) inputs and outputs as well as the timing information necessary for sequencing the computational steps. In other words, many of the computations can be conducted by a local and parallel message-passing process, with a minimum of external supervision, similar to the derivational steps taken by extensional systems (Pearl 1988a).

The advantages of this distributed, message-passing paradigm is that it provides a natural mechanism for exploiting the independencies embodied in sparsely constrained systems and translating them, by subtask decomposition, into a substantial reduction in complexity. Additionally, distributed propagation is inherently "transparent": The intermediate steps, by virtue of their reflecting interactions only among semantically related variables, are conceptually meaningful. This facilitates the use of natural, object-oriented programming tools and helps establish confidence in the final result.

Distributed schemes for belief *updating* and belief *revision* are described in Pearl (1986, 1987a). Belief updating aims at assigning each variable a posterior probability that correctly accounts for the evidence at hand. The aim of belief revision is to identify a composite set of propositions (one from each variable) that "best" explains the evidence at hand—i.e. attains the highest posterior probability. These involve the updating and transmittance of two types of messages: λ—the strength of evidential support that a variable obtains from its descendants; and π—the strength of causal support that a variable obtains from its nondescendants. This separation into causal and evidential components permits the execution of bidirectional inferences without the dangers of circular reasoning (see Section 2.2.1).

Figure 6 shows six successive stages of belief propagation through a

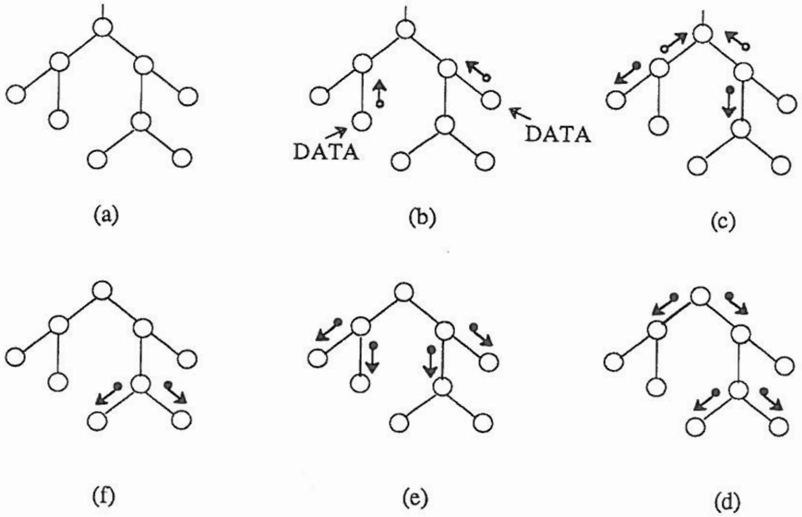


Figure 6 The impact of new data propagates through a tree by a message-passing process.

simple binary tree, assuming that all activities are triggered by changes in the parameters of neighboring processors. Initially (Figure 6*a*), the tree is in equilibrium, representing the state of belief due to all prior information. As soon as two nodes are activated by new information (Figure 6*b*), white tokens (representing λ) are placed on their links, directed towards their parents. Activated by these tokens, the parents compute their degree of belief and manufacture the appropriate number of tokens for their neighbors (Figure 6*c*): white tokens for their parents and black tokens (representing π) for the children. (The links through which the absorbed tokens have entered do not receive new tokens, thus reflecting the feature that a π-message is not affected by a λ-message crossing the same link.) The root node now receives two white tokens, one from each of its descendants. That triggers the production of two black tokens for top-down delivery (Figure 6*d*). The process continues in this fashion until, after six cycles, all tokens are absorbed, and the network reaches a new equilibrium, where each variable is assigned a probability measure reflecting the new information.

The updating scheme possesses the following properties:

1. New information diffuses through the network in a single pass—i.e. equilibrium is reached in time proportional to the diameter of the network.
2. The primitive processors are simple, repetitive, and they require no working memory except that used in matrix multiplication.
3. The local computations and the final belief distribution are entirely independent of the control mechanism that activates the individual operations. They can be activated by either data-driven or goal-driven (e.g. requests for evidence) control strategies, by a clock, or at random.

As soon as a node posts a token for its parent, it is ready to receive new data, and when this occurs, a new token is posted on the link, replacing the old one. In this fashion the network can track a changing environment and provide coherent interpretation of signals emanating simultaneously from multiple sources. Having an efficient mechanism of updating and/or revising beliefs also facilitates various control functions such as, for example, selecting the next best test in diagnosis. This can be done by the method of "hypothesizing"; we imagine what impact the outcome of various tests would have on some target hypothesis, and select the test with the highest impact.

The objective of updating beliefs coherently by purely local computations can be fully realized if the network is singly connected—i.e. if there is only one undirected path between any pair of nodes. These include trees, where each node has a single parent, as well as networks with multiparent nodes, representing events with several causal factors, as in Figure 5.

Here the π message transmitted from EARTHQUAKE to ALARM SOUND interacts with the λ message that ALARM receives from PHONE CALL to produce a reduction of the evidential support (λ) the ALARM SOUND lends to BURGLARY. This distinction between causal (π) and evidential (λ) supports identifies the origin of beliefs and permits the system to treat multiple causes differently from multiple symptoms; the former compete with each other, the latter support each other. It is due to this distinction that the system obtains coherent updating via modular computations, dispensing with the need to specify direct inhibitory connections from one cause to another (Pearl 1988b).

The profile of π and λ messages that load the network at any given time also provides the information needed for generating explanations, similar to the justification network in truth-maintenance systems (Doyle 1979). Tracing the most influential π and λ messages back to their origins yields a skeletal subgraph from which verbal explanations can be structured, clearly reflecting the distinction between causal and evidential supports.

3.1.3 COPING WITH LOOPS When loops are present, as in Figure 4, the network is no longer singly connected, and local propagation schemes invariably run into trouble. Several methods have been developed that extend the propagation method to networks containing loops while still maintaining global coherence relative to probability theory.[3] The most notable are conditioning, clustering, and stochastic simulation.

Before describing each of these methods, one should not overlook a simple but important approximation method called "ignore the loops"—i.e. propagate the π and λ messages according to the equations developed for a singly connected network. If loops are present, this strategy will cause the messages to circulate indefinitely until their magnitude becomes insignificantly small (this will always be the case because the conditional probabilities on the links tend to attenuate the messages). If the loops are long, ignoring them will not introduce a significant error because the degree of intermessage correlation, created by multiple paths, diminishes with the lengths of such paths. The results obtained after relaxation should be closer to the theoretical results than those obtained by extensional updating strategies, because the latter totally ignore the distinction between causal and evidential supports, while the former account for it in an approximate way.

The method of conditioning involves identifying a set of variables (called

[3] In general networks, the task of belief updating is NP-hard (Cooper 1987).

the *cycle cutset*) that, if known with certainty, would render the network singly connected; instantiating these variables to some values; conducting the propagation on the rest of the network; repeating the process for all possible instantiations; and then combining the results by taking their weighted average. In Figure 4, for example, we would run two propagation exercises, one under the assumption "Thousands dead" = true, the other under "Thousands dead" = false. The evidential supports obtained under these two assumptions would then be combined to yield the overall, unconditioned results.

The effectiveness of conditioning depends heavily on the topological properties of the network. In general, the number of instantiations required is 2^c, where c is the size of the cycle cutset chosen for conditioning. Since each propagation phase takes only time linear with the number of variables in the system (n), the overall complexity is exponential with the size of the cycle cutset that we can identify. If the network is sparse, topological considerations can be used to find a small cycle cutset and render the interpretation task tractable.

A second method of sidestepping the loop problem is that of stochastic simulation (Henrion 1986a). It amounts to generating a random population of scenarios agreeing with the evidence, then answering queries on the basis of this population. This is accomplished distributedly by having each processor inspect the current state of its neighbors, compute the belief distribution of its host variable, then randomly select one value from the computed distribution, to be inspected by its neighbors in their turn (Pearl 1987c). Probabilities are calculated by counting the frequency at which a proposition obtains the value *true*. The advantages of this method are that it is purely distributed and that the rate of convergence does not depend on the topology of the network. Unfortunately, the rate of convergence deteriorates when the links convey logical constraints—i.e. extreme probabilities (Chin & Cooper 1987).

The third technique, and currently the most promising, is that of *clustering*. It involves forming local groups of variables in such a way that the topology of the resulting network (treating each group as a single compound variable), is singly connected. For example, grouping the three intermediate nodes in Figure 4 into one compound variable will result in a three-node causal chain. Once a clustered configuration is found, the propagation methods described in the preceding subsection are applicable, with a processor assigned to each cluster. The complexity of this scheme is exponential with the size of the largest cluster found, because the processor assigned to manage that cluster must handle that many value combinations (e.g. eight in Figure 4).

A popular method of selecting clusters is to form *join trees*—i.e. trees made up of overlapping clusters in such a way that all links are contained within the clusters. The network of Figure 4, for example, will be decomposed into two overlapping clusters: one comprising the top four nodes, the other the bottom four. The merits of join-tree representations have been recognized by probabilists for over 25 years (e.g. Vorobev 1962; Goodman 1970; Haberman 1974). Their applications to databases are discussed in Beeri et al (1983) and Malvestuto (1986), and they have also been suggested for Bayes inferences (Lemmer 1983) and constraint processing (Dechter & Pearl 1988). A systematic method of finding such clusters and a thorough analysis of the updating scheme are described in Lauritzen & Spiegelhalter (1988). The method involves triangulating the network (Tarjan & Yannakakis 1984), identifying the maximal cliques of the triangulated (or chordal) graph, organizing the cliques in a tree structure, and assigning a processor to each clique. Beliefs can then propagate by the message-passing mechanism described in Section 3.1.2 (Pearl 1988a).

The attractive feature of clustering schemes is that, once the clusters are formed and their tree organization established, the resulting structure offers an effective database that can be amortized over many evidential reasoning tasks. A large variety of queries could be answered swiftly by unsupervised, local, and parallel processes. Therefore, if one takes seriously the paradigm that unsupervised parallelism is one capability that human learning aspires to achieve (Pearl 1986), then it is quite reasonable to speculate that the clusters found for join-tree representations form the nuclei around which higher cognitive concepts normally evolve.

It is important to note that the difficulties associated with the presence of loops are not unique to probabilistic formulations but are inherent to any problem where globally defined solutions are produced by local computations. Identical computational issues arise in Dempster-Shafer's formalism (Kong 1986), constraint-satisfaction problems (Dechter & Pearl 1987a), truth maintenance systems (Doyle 1979), diagnostic reasoning (Geffner & Pearl 1987a), relational databases (Beeri et al 1983), matrix inversion (Tarjan 1976), and network reliability (Arnborg et al 1987). The importance of network representation, though, is that it uncovers the core of these difficulties and provides a unifying abstraction that encourages the exchange of solution strategies across domains.

3.2 *Dempster-Shafer Theory and Constraint Networks*

Pure Bayesian theory requires the specification of a complete probabilistic model before reasoning can commence—i.e. determining for each variable X the conditional probabilities that govern the values of X, given the factors perceived as causes of those values. When a full specification is not available, Bayes practitioners have devised approximate methods of completing the model. For example, if we are given the impact of each individual cause but not the combined impact of several causes, we assume that they combine disjunctively, and that all exceptions are independent (Peng & Reggia 1986; Pearl 1987a).

An alternative method of handling partially specified models is provided by the Dempster-Shafer (D-S) theory (Shafer 1976). Rather than completing the model, the D-S theory sidesteps the missing specifications and resigns instead to less ambitious inference tasks: computing probabilities of *provability* (or *necessity*) rather than probabilities of *truths*. The partially specified model is idealized by qualitative relationships of compatibility constraints, and these qualitative relationships are then used as a logic for assembling proofs of various propositions. Items of evidence are modeled as probabilistic modifications of the available constraints, and the support they lend to a given hypothesis H is defined as the probability that a proof of H can be assembled.

The current popularity of the D-S theory stems from both its readiness to admit partial models and its compatibility with the classical, proof-based style of logical inference. As such, the approach matches the syntax of deductive databases and logic-programming languages but may inherit many of the problems associated with monotonic logic, some of which are discussed in Section 4.1.1.

3.2.1 BELIEF FUNCTIONS AS PROBABILITIES OF PROVABILITY I introduce the D-S theory from a rather unconventional perspective, one I hope will be more meaningful to AI researchers, especially those versed in constraint processing, truth maintenance systems, and logic programming. Our starting point is a static network of logical constraints that represents generic knowledge about the world. Each constraint is a declarative statement on a group of variables specifying what is and what is not permitted to hold in the domain. For example the rule $A \rightarrow B$ forbids the simultaneous assignment of *true* to A and *false* to B. A collection of such constraints yields a (possibly empty) set of *extensions* or *solutions*—i.e. assignments of values to all variables that simultaneously satisfy all constraints.

In addition to this static network, we also have items of evidence that provide direct but partial support to a select set of propositions in the system. Each such item of evidence is modeled as a randomly fluctuating constraint that, for a certain fraction of the time m, imposes the value *true* on the propositions supported by that item. The larger the m the stronger the support. To compute the overall support that several items of evidence impart to a given proposition, say A, we subject the static network to the corresponding set of externally imposed, randomly fluctuating constraints, assume that they act independently of each other, and ask for the proba-

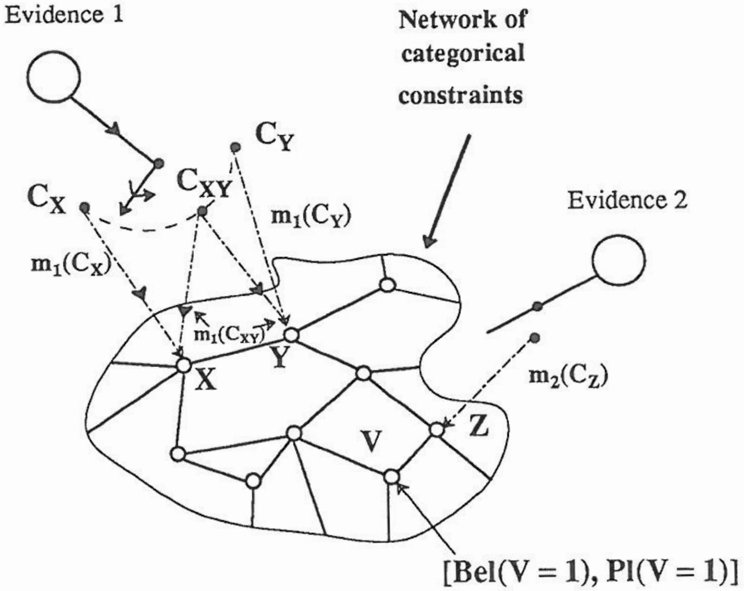


Figure 7 Multiple evidence modeled as random switches, imposing additional constraints on a static network of compatibility relations.

bility (or fraction of the time) that A can be proven true. This probability defines the belief function $Bel(A)$; similarly, a plausibility function $Pl(A) = 1 - Bel(\neg A)$ is defined by the probability that A is not proven false.

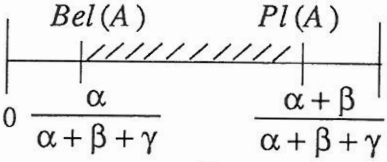
This scheme is illustrated metaphorically in Figure 7, which shows a static network of variables X, Y, Z, V . . . (the nodes) interacting via local constraints (the arcs), subject to the influence of two switches that impose additional time-varying constraints on various regions of the network. The switches represent two independent items of evidence, each characterized by the fraction of time spent in each position.

To illustrate the analysis of belief functions, let us assume that the static network represents the familiar graph-coloring problem: Each node may take on one of three possible colors, 1, 2, or 3, but no two adjacent nodes may take on identical colors. The position of the switches represents additional constraints—e.g. C_{XY}: either X or Y must contain the color 1, or, C_Z: Z cannot be assigned the color 2, etc. The relative time that a switch spends enforcing each of the constraints is indicated by the weight measures $m_1(C_X)$, $m_1(C_{XY})$, $m_2(C_Z)$, etc. Our objective is to compute $Bel(A)$ and $Pl(A)$, where A stands for the proposition $V = 1$, namely, variable V is assigned the color 1.

Figure 8 represents typical sets of solutions to the coloring problem

$$\begin{array}{lll}
 & \quad VXY\ \cdots & \\
\begin{array}{l}\text{Type-1 positions}\\ \text{Time} = \alpha\end{array} & \begin{bmatrix}1\,2\,3 & \cdots\\ 1\,1\,2 & \\ 1\,3\,2 & \end{bmatrix} & V = 1 \text{ in all solutions}\\
 & & \\
\begin{array}{l}\text{Type-2 positions}\\ \text{Time} = \beta\end{array} & \begin{array}{c}\begin{bmatrix}1\,2\,1 & \cdots\\ 2\,3\,1 & \\ 2\,2\,3 & \end{bmatrix}\\ \begin{bmatrix}3\,2\,1 & \cdots\\ 1\,2\,1 & \end{bmatrix}\end{array} & \begin{array}{l}V = 1 \text{ and } V \neq 1 \text{ are compatible}\\ \text{with each position}\end{array}\\
 & & \\
\begin{array}{l}\text{Type-3 positions}\\ \text{Time} = \gamma\end{array} & \begin{bmatrix}2\,1\,3 & \cdots\\ 2\,3\,1 & \\ 3\,3\,3 & \end{bmatrix} & V \neq 1 \text{ in all solutions}\\
 & & \\
\begin{array}{l}\text{Type-4 positions}\\ \text{Time} = \delta\end{array} & \begin{bmatrix}\text{Nil}\end{bmatrix} & \text{no solution}
\end{array}$$

(a)


Bel(A)
Pl(A)
0
$\frac{\alpha}{\alpha+\beta+\gamma}$
$\frac{\alpha+\beta}{\alpha+\beta+\gamma}$

(b)

Figure 8 (*a*) Four types of constraints in the graph-coloring problem and (*b*) the resulting belief interval for the proposition $A: V = 1$.

under different combinations of the switches (the actual values are fictitious). Each row represents one extension (or solution) where the entries indicate the value assigned to the variables (columns). The first set of solutions is characterized by having the value 1 assigned to V in each and every row. If the system spends a fraction α of the time in such combinations of switches, we say that $P(\mathbf{e} \models A) = \alpha$—i.e. the proposition A: "$V = 1$" can be proven true with probability α, given the evidence $\mathbf{e}$. A type-2 position is characterized by the column of V containing 1's as well as alternative values—e.g. 2 or 3. Each such position (or position combination) is compatible with both A and $\neg A$. Similarly, a type-3 position permits only extensions that exclude $V = 1$, while a type-4 position represents conflict situations; there exists no extension consistent with all the constraints. *Bel*(A) and *Pl*(A) are computed from the time spent in each type of constraint combination:

$$Bel(A) = \frac{\alpha}{\alpha+\beta+\gamma}$$

$$Pl(A) = 1 - Bel(V \neq 1) = 1 - \frac{\gamma}{\alpha+\beta+\gamma} = \frac{\alpha+\beta}{\alpha+\beta+\gamma}.$$

These are illustrated as a belief interval in Figure 8*b*.

The assumption of evidence independence, coupled with the normalization rule above, leads to an evidence-pooling procedure known as *Dempster's Rule of Combination*. For any combination of the evidential constraints, we need to decide whether the proposition A is entailed by that combination—i.e. if every extension contains A and none contain $\neg A$. The total time that a system spends under constraint combinations that compel A, divided by the total time spent in non-conflict combinations, yields $Bel(A)$.

The preceding analysis can be complex. The graph-coloring problem, even with only three colors, is known to be NP complete. Moreover, if each item of evidence is modeled by a 2-position switch, and if we have n such switches, then a brute force analysis of $Bel(A)$ would require solving 2^n graph-coloring problems. Analyzing the solutions obtained under every switch combination and identifying those combinations yielding $\mathbf{e} \models A$ seems hopeless. Fortunately, these difficulties can be alleviated by decomposing the network into a tree of clusters, where solutions can be obtained in linear time (Dechter & Pearl 1988). In trees, belief functions can be calculated by local computations because, as with probability calculations, the belief function associated with each variable can be computed from partial belief functions associated with its neighbors. The use of tree decomposition techniques for belief function computations are reported in Shafer et al (1987) and Kong (1986).

3.2.2 COMPARING BAYES AND DEMPSTER-SHAFER FORMALISMS

The D-S theory differs from probability theory in several aspects. First, it accepts an incomplete probabilistic model where some parameters (e.g. the prior or conditional probabilities) are missing. Second, the probabilistic information that is available, like the strength of evidence, is not interpreted as likelihood ratios but rather as random epiphenomena that impose truth values on various propositions for a certain fraction of the time. This model permits a proposition and its negation to be simultaneously compatible (with the evidence) for a certain portion of the time, and this may permit the sum of their beliefs to be smaller than unity. Finally, owing to the incompleteness of the model, the D-S theory does not pretend to provide full answers to probabilistic queries; rather, it resigns to providing partial answers. It estimates how close the evidence is to forcing the necessity of the hypothesis, instead of estimating how close the hypothesis is to being true.

Phrased another way, the D-S theory computes the probability that some set of conditions suggested by the evidence would materialize, from which the truth of A can be derived out of logical necessity. Thus, instead of the conditional probability $P(A|\mathbf{e})$, the D-S theory computes the probability of the logical entailment $\mathbf{e} \models A$. The entailment $\mathbf{e} \models A$ is not a proposition in the ordinary sense, but a meta-level relationship between $\mathbf{e}$ and A, requiring a logical, object-level theory by which proofs from $\mathbf{e}$ to A can be assembled. In the D-S scheme the object-level theory consists of categorical, *compatibility* constraints—e.g. that it is incompatible for an alarm system to turn off unless either a burglary or an earthquake occurred (see Figure 5). It is remarkable that, while the calculation of $P(A|\mathbf{e})$, and even the probability of the material conditional $P(\mathbf{e} \supset A)$, requires complete probabilistic models, $P(\mathbf{e} \models A)$ does not.

At this point, it is worthwhile reflecting on the significance of the interval $Pl(A) - Bel(A)$ in the D-S formalism. This interval is often interpreted as the degree of our ignorance about probabilities—i.e. the range where the "true" probability should fall if we had a complete probabilistic model. Such measures would have been a useful supplement to Bayes methods, which always provide point probabilities and thus can give a false sense of security in the model.

Unfortunately, the D-S intervals have little to do with ignorance, nor do they represent *bounds* on the probabilities that would ensue once ignorance is removed. For example, the interval $Pl(A) - Bel(A)$ often vanishes when the model is far from being complete. The equality $Bel(A) = Pl(A)$ simply means that, based on the categorical abstraction captured by the compatibility constraints, the available evidence could not simultaneously be compatible with A and its negation $\neg A$. It is curious to note that applying the same interpretation to noncategorical models yields an interval that *never* vanishes because, barring extreme probabilities, a body of evidence is always compatible with both a proposition and its negation. For example, if in the model of Figure 5 we assume that all rules have exceptions (e.g. there is a nonzero chance of false alarm, a nonzero chance of a prank phone call, etc) then all propositions will be assigned zero belief and unit plausibility, because none can actually be *proven* true. Thus, the choice of a categorical abstraction is a crucial one (Pearl 1988a).

3.2.3 RELATIONS TO TRUTH MAINTENANCE SYSTEMS AND INCIDENCE CALCULUS The readiness of the D-S formalism to accept knowledge in the form of logical constraints, rather than conditional probabilities, renders it close to uncertainty management techniques developed in the logicist camp of AI, most notably truth-maintenance systems (TMS) (Doyle 1979) and incidence calculus (Bundy 1985). These two approaches can be regarded as cousins to the Dempster-Shafer theory because, like the latter, they are based on *provability* as the basic relationship connecting evidence with a conclusion.

Truth-maintenance systems also use logical rules as their elementary units of knowledge, and (see Section 3.2.1) conclusions are drawn by piecing together rules to form proofs. Likewise, rules may have exceptions that may cause the expected conclusion of the proof to clash with observed facts or with other deductions. However, whereas the exceptions and/or assumptions in the D-S theory were summarized numerically, using the evidence weight m, the TMS approach maintains an explicit list of the main assumptions and exceptions involved in each rule.

In the assumption-based TMS approach (ATMS; De Kleer 1986) one further maintains for each conclusion c a list $L(c)$ of nonredundant sets of assumptions called *environments*, each of which is sufficient to support a proof of c. Thus $L(c)$ is a Boolean expression whose truth signifies the existence of a proof for c. If we are given probabilities on the assumptions that appear in $L(c)$ and if we further assume that they are independent, then we can obtain $Bel(c)$ by simply computing the conditional probability of $L(c)$, given that the assumptions are consistent:

$$Bel(c) = P[L(c) \mid \text{consistency}].$$

Moreover, the computation can be done symbolically, which might be more efficient than the computation method shown in Section 3.2.1. Thus, the ATMS can be used as a symbolic engine for computing the belief functions sought by the D-S theory. Steps in this direction have been taken by D'Ambrosio (1987) and Laskey & Lehner (1988).

Incidence calculus (Bundy 1985) suggests a method of computing belief functions by logical sampling, a technique similar in spirit to that of stochastic simulation (Henrion 1986a; Pearl 1987c). A probabilistic model is used to generate random samples of truth values (bit strings) for a select set of propositions representing uncertain facts. These values are presented as assumptions, or axioms, to a theorem prover. Different sets of assumptions give rise to different theorems, and $Bel(c)$ is given by that fraction of the time that c is proven from a consistent set of assumptions. This scheme is a physical embodiment of the random-switch model described in Figure 7. The random position of each switch is replaced by a random bit string assigned to the propositions (i.e. assumptions) impacted by the evidence.

The advantage of this scheme is that the theorem prover can be general purpose (e.g. first-order logic), not limited to propositional constraints. Moreover, the scheme can simulate dependencies among items of evidence, provided the bit strings are generated by a probabilistic model (e.g. a causal network) that embodies these dependencies.

4. LESSONS AND OPEN ISSUES

4.1 *Relations to Nonmonotonic Logic*

4.1.1 SOFTENED LOGIC VS HARDENED PROBABILITIES The ills of monotonic logic have often been attributed to its coarse and sharp, bi-valued character. Indeed, when one tries to figure out why logic would not predict the obvious fact that penguins do not fly even though they are birds, the first thing one tends to blame is the sharp, uncompromising stance toward exceptions of the rule "birds fly." It is natural, therefore, to assume that once we soften the constraints of Boolean logic and allow truth values to be measured on a grey scale, these problems will disappear. There have been several attempts along this line. Rich (1983) has proposed a likelihood-based interpretation of default rules, managed by certainty-factors calculus. Ginsberg (1984) and Baldwin (1987) have pursued similar aspirations using the Dempster-Shafer notion of belief functions. While these attempts produce valuable results (revealing, for instance, how sensitive a conclusion is to the uncertainty of its premises), the fundamental problem of monotonicity remains unresolved. For example, regardless of the certainty calculus used, these analyses always yield an increase in the belief that penguins can fly, if one adds the superfluous information that penguins are birds and birds normally fly. Identical problems surface in the use of incidence calculus and softened versions of truth-maintenance systems (D'Ambrosio 1987; Laskey & Lehner 1988).

Evidently, it is not enough to add a soft probabilistic veneer on top of a system that is basically structured after hard monotonic logic. The problem with monotonic logic lies not in the hardness of its truth values, but rather in its inability to process context-dependent information. Logic does not have a device equivalent to the conditional probability statement "$P(B|A)$ is high," whose main function is to identify the context A where the proposition B can be believed, and to make sure that only legitimate changes in that context (e.g. going from A = penguins to A' = bird-penguins or A'' = white penguins) will be permitted without significant changes in the belief of B.

Lacking an appropriate logical device for conditionalization, the natural tendency is to interpret the English sentence "if A then B" as a softened version of the material implication constraint $A \supset B$. A useful consequence of such softening is allaying the fears of outright contradictions. For example, while the classical interpretation of the three rules: "penguins do not fly," "penguins are birds," and "birds fly," yields an unforgivable contradiction, the uncertainties attached to these rules now render them manageable. However, they are still managed in the wrong way, because the material-implication interpretation of if-then type rules is so fundamentally wrong that its maladies cannot be rectified by simply allowing exceptions in the form of shaded truth values. The source of the problem lies in the property of transitivity,

$$(a \rightarrow b, b \rightarrow c) \Rightarrow a \rightarrow c,$$

which is inherent in the material-implication interpretation. There are occasions where rule transitivity must be totally suppressed, not merely weakened, or else strange results will surface. One such occasion occurs in property inheritance, where subclass specificity should override superclass properties. Another occurs in causal reasoning, where predictions should not trigger explanations (e.g. "sprinkler-on" predicts "wet-ground," "wet-ground" suggests "rain," yet "sprinkler-on" should not suggest "rain"). In such cases, softening the rules only weakens the flow of inference through the rule chain but does not bring it to a dead halt, as it should.

Apparently, what it needed is a new interpretation of "if-then" statements, one that does not destroy the context-sensitive character of probabilistic conditionalization. McCarthy (1986) remarks that circumscription indeed provides such an interpretation. In his words:

> Since circumscription doesn't provide numerical probabilities, its probabilistic interpretation involves probabilities that are either infinitesimal, within an infinitesimal of one, or intermediate—without any discrimination among the intermediate values. The circumscriptions give conditional probabilities. Thus we may treat the probability that a bird can't fly as an infinitesimal. However, if the rare event occurs that the bird is a penguin, then the conditional probability that it can fly is infinitesimal, but we may hear of some rare condition that would allow it to fly after all.

Rather than contrive new logics and hope that they match the capabilities of probability theory, an alternative approach would be to start with probability theory and, if we can't get the numbers or find their use inconvenient, we can extract qualitative approximations as idealized abstractions of the latter, while preserving its context-dependent properties. In this way, nonmonotonic logics should crystallize that are guaranteed to capture the context-dependent features of natural defaults (Pearl 1988a).

4.1.2 THE LOGIC OF "ALMOST TRUE" This program had in fact been initiated over 20 years ago by the philosopher Ernest Adams (1966), who developed a logic of conditionals based on probabilistic semantics. The sentence "if *A* then *B*" is interpreted to mean that the conditional probability of *B* given *A* is very close to 1, short of actually being 1. An adaptation of Adams's logic to default schema of the form $Bird(x) \rightarrow Fly(x)$, where x is a variable, is reported in Geffner & Pearl (1987b). The resulting logic is nonmonotonic relative to learning new facts, in accordance with McCarthy's desiderata. For example, learning that Tweety is a bird would yield the conclusion that Tweety can fly; subsequently learning that Tweety is also a penguin would yield the opposite conclusion: Tweety can't fly. Further, learning that Tweety is white will not alter this belief, because white is a typical color for penguins. However, and this is where it falls short of expectations, learning that Tweety is clever would cause Adams's logic to retract all previously held beliefs about Tweety's flying and answer: "I don't know." The logic is so conservative that it never jumps to conclusions that some new rule schema might invalidate (e.g. that clever penguins can fly). In other words, the logic does not capture the usual convention that, unless we are told otherwise, properties are presumed to be *irrelevant* to each other.[4]

Attempts to enrich Adams's logic with relevance-based features are described in Pearl (1987d), Geffner & Pearl (1987b) and Geffner (1988). The idea is to follow a default strategy similar to that of belief networks (Section 3.1); dependencies exist only if they are mentioned explicitly or if they logically follow from other explicit dependencies. However, whereas the stratified method of constructing belief networks ensures that all relevant dependencies were already encoded in the network, this can no longer be assumed when knowledge is presented in the form of isolated default rules and logical constraints. A new logic is needed to tell us when one relevancy follows from others. This issue is further discussed in Section 4.2.

4.1.3 THE ISSUE OF CONSISTENCY There is another dimension along which probabilistic analysis can assist current research in nonmonotonic logics. The latter do not provide any criterion for testing whether a database comprising default rules is internally consistent. The prevailing attitude is that once we tolerate exceptions we might as well tolerate anything (Brachman 1985). However, there is a sharp qualitative difference between exceptions and outright contradictions. For example, the statement "red penguins can fly" can be accepted as a description of a world in which redness defines an abnormal type of penguin. However, the statements "typically birds fly" and "typically birds do not fly" stand in outright contradiction to each other; since there is no world in which the two can hold simultaneously, they will invariably lead to strange, inconsistent

[4] Grosof (1986) discusses this convention in terms of a principle of maximizing conditional independencies, similar in spirit to the maximum entropy principle (Cheeseman 1983).

conclusions. While such obvious contradictions can easily be removed from the database (e.g. Touretzky 1986), more subtle ones might escape detection—e.g. "birds fly," "birds are feathered animals," "feathered animals are birds," and "feathered animals do not fly."

Adams's logic provides a criterion for detecting such inconsistencies, in the form of three axioms that should never be violated. In inheritance hierarchies this criterion yields a simple graphical test (Pearl 1987e), which is a generalization of Touretzky's: A network N is consistent iff for every pair of conflicting rules $p_1 \rightarrow q$ and $p_2 \rightarrow \neg q$, p_1 and p_2 are distinct and there is no cycle of rules that embraces both p_1 and p_2. For more intricate structures of default rules the test becomes more involved.

4.2 *Graphoids and the Formalization of Relevance*

A central requirement in several topics of this survey has been to articulate the conditions under which one item of information is considered relevant to another, given what we already know, and to encode knowledge in structures that vividly display these conditions as the knowledge undergoes changes. Different formalisms give rise to different definitions of relevance. For example, in probability theory, relevance is identified with dependence. In constraint-based formalisms (and in relational databases) relevance is associated with induced constraints; two variables are said to be relevant to each other if we can restrict the range of values permitted for one by constraining the other.

The essence of relevance can be identified with a structure common to all these formalisms. It consists of four axioms that convey the simple idea that when we learn an irrelevant fact, the relevance relationships of all other propositions remain unaltered; any information that was irrelevant remains irrelevant and that which was relevant remains relevant. Structures that conform to these axioms are called *graphoids* (Pearl & Paz 1987). Interestingly, both undirected graphs and directed acyclic graphs conform to the graphoids axioms (hence the name) if we associate the sentence "variable x is irrelevant to variable y once we know z" with the graphical condition "every path from x to y is intercepted by the set of nodes corresponding to z." [A special definition of "intercept" is required for directed graphs (Pearl 1988a).]

With this perspective in mind, graphs, networks, and diagrams can be viewed as inference engines devised for efficiently representing and manipulating relevance relationships: The topology of the network is assembled from a list of local relevance statements (e.g. direct dependencies), this input list entails (using the graphoid axioms) a host of additional statements, and the function of the graph is to ensure that a substantial portion of the latter can be read off by simple graphical criteria. Such a mapping will enable one to determine, at any state of knowledge z, which information is relevant to the task at hand and which can be ignored. Permissions to ignore, as we saw in Section 3.1, are the fuel that gives intensional systems the power to act.

An important result from the theory of graphoids states that Bayesian networks constitute a sound and complete inference mechanism relative to probabilistic dependencies—i.e. it identifies, in polynomial time, each and every conditional-independence relationship that logically follows from those used in the construction of the network (Pearl & Verma 1987; Geiger & Pearl 1988). Similar results hold for other types of relevance relationships—e.g. partial correlations and constraint-based dependencies. However, the essential requirement for soundness and completeness is that the network be constructed *causally*—i.e. that we specify, recursively, the relationship of each variable to its predecessors in some total order. (Once the network is constructed, the original order can be forgotten; only the partial order displayed in the network matters.)

One can speculate whether it is this soundness-completeness feature that renders causal schemata so important in knowledge organization. More generally, the precise relationship between causality as a representation of irrelevancies and causality as a commitment to a particular inference strategy [e.g. chronological ignorance (Shoham 1986)] is yet to be fully investigated. A different notion of relevance has been explored by Subramanian & Genesereth (1987), based on logical derivability. The latter takes propositions, rather than variables, as the atomic entities in the relevance relationships; again, the connection to graphoid structures is not fully understood.

ACKNOWLEDGMENTS

This paper is a revision of a survey presented at the AAAI-87 Conference, Seattle, Washington, August 1987, later printed under the title "Evidential Reasoning Under Uncertainty," in *Exploring Artificial Intelligence*, ed. H. Shrobe, Los Altos: Morgan-Kaufmann, 1988. It also contains sections from my book, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan-Kaufmann, 1988. Reprinted by permission from the publisher.

This work was partially supported by NSF under Grants No. IRI-8718920 and IRI-8610155, and NRL under Grant No. N00014-87-K-2029.

I thank Norm Dalkey, Rina Dechter, Hector Geffner, and Azaria Paz for many helpful discussions.

Literature Cited

Adams, E. 1966. Probability and the logic of conditionals. In *Aspects of Inductive Logic*, ed. J. Hintikka, P. Suppes. Amsterdam: North-Holland

Andreassen, S., Woldbye, M., Falck, B., Andersen, S. K. 1987. MUNIN—A causal probabilistic network for interpretation of electromyographic findings. *Proc. Int. Joint Conf. Artif. Intell. 10th, Milan, Italy*, pp. 366–72

Arnborg, S., Corneil, D. G., Proskurowski, A. 1987. Complexity of finding embeddings in a k-tree. *SIAM J. Algebraic Discrete Methods* 8(2): 277–84

Baldwin, J. F. 1987. Evidential support logic programming. *Fuzzy Sets and Systems* 24: 1–26

Beeri, C., Fagin, R., Maier, D., Yannakakis, M. 1983. On the desirability of acyclic database schemes. *J. Assoc. Comput.* 30: 479–513

Ben-Bassat, M., Carlson, R. W., Puri, V. K., Lipnick, E., Portigal, L. D., Weil, M. H. 1980. Pattern-based interactive diagnosis of multiple disorders: the MEDAS system. *IEEE Trans. Pattern Anal. & Machine Intell.* PAMI-2(2): 148–60

Blalock, H. M. 1971. *Causal Models in the Social Sciences*. London: Macmillan

Bonissone, P. P., Gans, S. S., Decker, K. S. 1987. RUM: a layered architecture for reasoning with uncertainty. *Proc. Int. Joint Conf. Artif. Intell., 10th, Milan, Italy*, pp. 891–98

Brachman, R. J. 1985. I lied about the trees, or, defaults and definitions in knowledge representation. *AI Mag.* 6(3): 80–93

Bundy, A. 1985. Incidence calculus: a mechanism for probabilistic reasoning. *J. Automated Reasoning* 1: 263–83

Chandrasakaran, B., Mittal, S. 1983. Conceptual representation of medical knowledge for diagnosis by computer: MDX and related systems. *Adv. Comput.* 22: 217–93

Cheeseman, P. 1983. A method of computing generalized Bayesian probability values for expert systems. *Proc. Int. Joint Conf. Artif. Intell., 6th, Karlsruhe, W. Germany*, pp. 198–202

Chin, H. L., Cooper, G. F. 1987. Stochastic simulation of Bayesian belief networks. *Proc. Uncertainty in AI Workshop, Seattle*, pp. 106–13

Clancey, W. J. 1985. Heuristic classification. *Artif. Intell.* 27(3): 289–350

Cohen, P. R. 1985. *Heuristic Reasoning about Uncertainty: an Artificial Intelligence Approach*. Boston: Pitmans

Cohen, P., Day, D., Delisio, J., Greenberg, M., Kjeldsen, R., Suthers, D., Berman, P. 1987a. Management of uncertainty in medicine. *Int. J. Approx. Reasoning* 1(1): 103–16

Cohen, P. R., Shafer, G., Shenoy, P. P. 1987b. Modifiable combining functions. *Proc. Uncertainty in AI Workshop, Seattle*, pp. 10–21

Cooper, G. F. 1987. Probabilistic inference using belief networks is NP-hard. *Rep. KSL-87-27*. Medical Computer Science Group, Stanford University

D'Ambrosio, B. 1987. Truth maintenance with numeric certainty estimates. *Proc. Conf. on AI Applications, 3rd, Orlando*, pp. 244–49

De Kleer, J. 1986. An assumption-based truth maintenance system. *Artif. Intell.* 29: 241–88

Dechter, R., Pearl, J. 1987a. Network-based heuristics for constraint-satisfaction problems. *Artif. Intell.* 34(1): 1–38

Dechter, R., Pearl, J. 1988. Tree-clustering schemes for constraint-processing. *Proc. AAAI-88, St. Paul, Minn.*, pp. 150–54. *Artif. Intell.* 38(3): 353–66

Doyle, J. 1979. A truth maintenance system. *Artif. Intell.* 12(3): 231–72

Duda, R. O., Hart, P. E., Nilsson, N. J. 1976. Subjective Bayesian methods for rule-based inference systems. *Proc. Natl. Comput. Conf. (AFIPS)* 45: 1075–82

Duncan, O. D. 1975. *Introduction to Structural Equation Models*. New York: Academic

Geffner, H., Pearl, J. 1987a. An improved constraint-propagation algorithm for diagnosis. *Proc. Int. Joint Conf. Artif. Intell., 10th, Milan, Italy*, pp. 1105–11

Geffner, H., Pearl, J. 1987b. A sound framework for reasoning with defaults. *Proc. Conf. Soc. Exact Philosophy, Rochester, New York*, May 1988

Geffner, H. 1988. On the logic of defaults. *Proc. AAAI-88, St. Paul, Minn.*, pp. 449–54

Geiger, D., Pearl, J. 1988. On the logic of causal models. *Proc. AAAI Workshop on Uncertainty in AI, Minneapolis*, pp. 136–47

Ginsberg, M. L. 1984. Non-monotonic reasoning using Dempster's rule. *Proc. Natl. Conf. Artif. Intell., 3rd, Austin, Texas*, pp. 126–29

Goodman, R. M. 1970. The multivariate analysis of qualitative data: interaction among multiple classifications. *J. Am. Stat. Assoc.* 65: 226–56

Grosof, B. N. 1986. Non-monotonicity in probabilistic reasoning. *Proc. AAAI Workshop on Uncertainty in AI, Philadelphia*, pp. 91–98

Haberman, S. J. 1974. *The general log-linear model.* PhD thesis, Dept. Statistics, Univ. Chicago

Hájek, P. 1985. Combining functions for certainty degrees in consulting systems. *Int. J. Man-Machine Stud.* 22: 59–65

Hájek, P., Valdes, J. J. 1987. Algebraic foundations of uncertainty processing in rule-based expert systems. *Ceskoslovenka Akademie Ved, Matematicky Ustav* (*Tech. Rep.*)

Heckerman, D. 1986a. A probabilistic interpretation for MYCIN's certainty factors. In *Uncertainty in Artificial Intelligence*, ed. L. N. Kanal, J. F. Lemmer. Amsterdam: North-Holland

Heckerman, D. 1986a. A rational measure of confirmation. *Tech. Rep. Memo-KSL-86-25*. Medical Computer Science Group, Stanford University

Henrion, M. 1986a. Propagation of uncertainty by logic sampling in Bayes' networks. *Dept. Eng. Public Policy, Tech. Rep.* Carnegie-Mellon Univ.

Henrion, M. 1986b. Should we use probability in uncertain inference systems? *Proc. Cognit. Sci. Soc. Meet., Amherst*, pp. 320–30

Horvitz, E. J., Heckerman, D. E. 1986. The inconsistent use of measures of certainty in artificial intelligence research. In *Uncertainty in Artificial Intelligence*, ed. L. Kanal, J. Lemmer, pp. 137–51. Amsterdam: North-Holland

Howard, R. A., Matheson, J. E. 1981. Influence diagrams. In *Principles and Applications of Decision Analysis.* Menlo Park, Calif: Strategic Decisions Group

Kanal, L. N., Lemmer, J. F., ed. 1986. *Uncertainty in Artificial Intelligence*. Amsterdam: North-Holland

Kenny, D. A. 1979. *Correlation and Causality*. New York: Wiley

Kiiveri, H., Speed, T. P., Carlin, J. B. 1984. Recursive causal models. *J. Australian Math. Soc.* 36: 30–52

Kong, A. 1986. *Multivariate belief functions and graphical models.* PhD thesis, Dept. Statistics, Harvard Univ.

Laskey, K. B., Lehner, P. E. 1988. Belief maintenance: an integrated approach to uncertainty management. *Proc. Natl. Conf. Artif. Intell., 7th, St. Paul, Minn.*, pp. 210–14

Lauritzen, S. L. 1982. *Lectures on Contingency Tables*. Aalborg, Denmark: Univ. Aalborg Press. 2nd ed.

Lauritzen, S. L., Spiegelhalter, D. J. 1988. Local computations with probabilities on graphical structures and their applications to expert systems. *J. Roy. Statist. Soc. Ser. B* 50(2): 154–227

Lemmer, J. 1983. Generalized Bayesian updating of incompletely specified distributions. *Large Scale Syst.* 5: 51–68

Lowrance, J. D., Garvey, T. D., Strat, T. M. 1986. A framework for evidential-reasoning systems. *Proc. Natl. Conf. Artif. Intell., 5th, Philadelphia*, pp. 896–901

Malvestuto, F. M. 1986. Decomposing complex contingency tables to reduce storage requirements. In *International Workshop on Scientific and Statistical Database Management*, ed. R. Cubitt et al, pp. 66–71. Luxembourg

McCarthy, J. 1986. Applications of circumscription to formalizing common-sense knowledge. *Artif. Intell.* 28(1): 89–116

Miller, R. A., Poole, H. E., Myers, J. P. 1982. INTERNIS-1, an experimental computer-based diagnostic consultant for general internal medicine. *New Engl. J. Med.* 307(8): 468–70

Montanari, U. 1974. Networks of constraints, fundamental properties and applications to picture processing. *Inf. Sci.* 7: 95–132

Nilsson, N. 1986. Probabilistic logic. *Artif. Intell.* 28(1): 71–87

Pearl, J. 1986. Fusion, propagation and structuring in belief networks. *Artif. Intell.* 29(3): 241–88

Pearl, J. 1987a. Distributed revision of composite beliefs. *Artif. Intell.* 33(2): 173–215

Pearl, J. 1987b. Bayes decision methods. In *Encyclopedia of Artificial Intelligence*, pp. 48–56. New York: Wiley Interscience

Pearl, J. 1987c. Evidential reasoning using stochastic simulation of causal models. *Artif. Intell.* 32(2): 245–58

Pearl, J. 1987d. Probabilistic semantics for inheritance hierarchies with exceptions. UCLA Cognit. Syst. Lab., *Tech. Rep. 870052 (R-93)*. Also in Pearl 1988a

Pearl, J. 1987e. Deciding consistency in inheritance networks. UCLA Cognit. Syst. Lab., *Tech. Rep. 870053 (R-96)*

Pearl, J. 1988a. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Palo Alto, Calif: Morgan-Kaufmann. 552 pp.

Pearl, J. 1988b. Embracing causality in formal reasoning. *Artif. Intell.* 35(2): 259–71

Pearl, J., Paz, A. 1987. Graphoids: a graph-based logic for reasoning about relevance relations. In *Advances in Artificial Intelligence-II*, ed. B. Du Boulay et al, pp. 357–63. Amsterdam: North-Holland

Pearl, J., Verma, T. 1987. The logic of representing dependencies by directed graphs. *Proc. AAAI-87, Seattle*, pp. 374–79

Peng, Y., Reggia, J. 1986. Plausibility of diagnostic hypotheses. *Proc. Natl. Conf. Artif. Intell., 5th, Philadelphia*, pp. 140–45

Perez, A., Jirousek, R. 1985. Constructing an intensional expert system (INES). In

Medical Decision Making, pp. 307–15. Amsterdam: Elsevier

Polya, G. 1954. *Patterns of Plausible Inference*. Princeton, NJ: Princeton Univ. Press

Prade, H. 1983. A synthetic view of approximate reasoning techniques. *Proc. Int. Joint Conf. Artif. Intell., 8th, Karlsruhe, W. Germany*, pp. 130–36

Quinlan, J. R. 1983. Inferno: a cautious approach to uncertain inference. *Comput. J.* 26: 255–69

Rich, E. 1983. Default reasoning as likelihood reasoning. *Proc. Int. Joint Conf. Artif. Intell., 8th, Karlsruhe, W. Germany*, pp. 348–51

Shachter, R. D. 1988. Probabilistic inference and influence diagrams. *Operations Res.* 36: 589–604

Shachter, R. D., Heckerman, D. V. 1987. A backward view for assessment. *AI Mag.* 8(8): 55–62

Shafer, G. 1976. *Mathematical Theory of Evidence*. Princeton, NJ: Princeton Univ. Press

Shafer, G., Shenoy, P. P., Mellouli, K. 1987. Propagating belief functions in qualitative Markov trees. *Intl. J. Approx. Reasoning* 1(4): 349–400

Shoham, Y. 1986. Chronological ignorance: time, nonmonotonicity, necessity and causal theories. *Proc. AAAI-86, Philadelphia*, pp. 389–93

Shortliffe, E. H. 1976. *Computer-Based Medical Consultation: MYCIN*. Amsterdam: Elsevier

Stephanou, H., Sage, A. 1987. Perspectives on imperfect information processing. *IEEE Trans. Syst., Man, Cybernet.* SMC-17(5): 780–98

Subramanian, D., Genesereth, M. 1987. The relevance of irrelevance. *Proc. Int. Joint Conf. Artif. Intell., 10th, Milan, Italy*, pp. 416–22

Tarjan, R. E. 1976. Graph theory and Gaussian elimination. In *Sparse Matrix Computations*, ed. D. J. Rose, pp. 3–22. New York: Academic

Tarjan, R. E., Yannakakis, M. 1984. Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM J. Comput.* 13: 566–79

Thompson, T. R. 1985. Parallel formulation of evidential reasoning theories. *Proc. Int. Joint Conf. Artif. Intell., 8th, Los Angeles*, pp. 321–27

Touretzky, D. S. 1986. *The Mathematics of Inheritance Systems*. San Mateo, Calif: Morgan Kaufmann

Vorobev, N. N. 1962. Consistent families of measures and their extensions. *Theory Probab. Appl.* 7: 147–63

Wermuth, N., Lauritzen, S. L. 1983. Graphical and recursive models for contingency tables. *Biometrika* 70: 537–52

Wold, H. 1964. *Econometric Model Building*. Amsterdam: North-Holland

Wright, S. 1921. Correlation and causation. *J. Agric. Res.* 20: 557–85

Wright, S. 1934. The method of path coefficients. *Ann. Math. Statist.* 5: 161–215