Probabilistic Logic*

Nils J. Nilsson
*Computer Science Department, Stanford University,
Stanford, CA 94305, U.S.A.*

Recommended by Donald Loveland

ABSTRACT

Because many artificial intelligence applications require the ability to reason with uncertain knowledge, it is important to seek appropriate generalizations of logic for that case. We present here a semantical generalization of logic in which the truth values of sentences are probability values (between 0 *and* 1*). Our generalization applies to any logical system for which the consistency of a finite set of sentences can be established. The method described in the present paper combines logic with probability theory in such a way that probabilistic logical entailment reduces to ordinary logical entailment when the probabilities of all sentences are either* 0 *or* 1.

1. Introduction

Several artificial intelligence (AI) applications require the ability to reason with uncertain information. For example, in "expert systems," many of the rules obtained from experts as well as data provided by users arė not known with certainty. Since ordinary logic is so useful in those cases in which knowledge *is* certain, AI researchers have been interested in various generalizations of logic for dealing with uncertainties.

There is extensive mathematical literature on probabilistic and plausible inference, which we will not review here. (See for example [1–8].) One of the early expert systems in AI embodying a technique designed to handle uncertain knowledge was MYCIN [9]. The PROSPECTOR system [10] used a reasoning method based on Bayes' rule and is quite similar to MYCIN. Lowrance and Garvey [11, 12] have adapted the Shafer–Dempster theory to AI applications. AI researchers have also investigated methods based on finding maximum-entropy probability distributions [13–16]. Halpern and Rabin [17] propose a modal logic with a "likelihood operator." Although a number of reasoning methods have

* Much of the research on which this paper is based was carried out while the author was an employee of SRI International.

0004-3702/86/$3.50 © 1986, Elsevier Science Publishers B.V. (North-Holland)

been explored in AI, many expert systems still rely on ad hoc techniques that have little theoretical justification.

In this paper we present a semantical generalization of ordinary first-order logic in which the truth values of sentences can range between 0 and 1. The truth value of a sentence in *probabilistic logic* is taken to be the *probability* of that sentence in ordinary first-order logic. We make precise the notion of the probability of a sentence through a possible-worlds analysis. Our generalization applies to any logical system for which the consistency of a finite set of sentences can be established.

2. Possible Worlds and Probabilities

To define what we mean by the *probability of a sentence* we must start with a sample space over which to define probabilities (as is customary in probability theory). A sentence S can be either *true* or *false*. If we were concerned about just the one sentence S, we could imagine two sets of *possible worlds*—one, say $\mathcal{W}_1$, containing worlds in which S was *true* and one, say $\mathcal{W}_2$, containing worlds in which S was *false*. The *actual* world, the world we are actually in, must be in one of these two sets, but we might now know which one. We can model our uncertainty about the actual world by imagining that it is in $\mathcal{W}_1$ with probability p_1, and is in $\mathcal{W}_2$ with some probability $p_2 = 1 - p_1$. In this sense we can say that the probability of S (being true) is p_1.

If we have more sentences, we have more sets of possible worlds. Sentences may be *true* in some worlds and *false* in others—in different combinations. Each set contains worlds with a unique and consistent set of truth values for the sentences. If we have L sentences, we might have as many as 2^L sets of possible worlds. Typically though, we will have fewer than this maximum number because some combinations of *true* and *false* values for our L sentences will be logically inconsistent. We cannot, for example, imagine a world in which S_1 is *false*, S_2 is *true* and $S_1 \wedge S_2$ is *true*. That is, some sets of the 2^L worlds might contain only *impossible* worlds.

As an example, consider the sentences

$$\{P, P \supset Q, Q\}.$$

The consistent sets of truth values for these three sentences are given by the columns in the following table:

P	*true*	*true*	*false*	*false*
$P \supset Q$	*true*	*false*	*true*	*true*
Q	*true*	*false*	*true*	*false*

In this case, there are four sets of possible worlds each one corresponding to one of these four sets of truth values.

One method for determining the sets of consistent truth values, given a set $\mathscr{S}$ of sentences, is based on developing a binary *semantic tree.* At each node we branch left or right, depending on whether or not we assign one of the sentences in $\mathscr{S}$ a value of *true* or *false*, respectively. Just below the root we branch on the truth value of one of the sentences in $\mathscr{S}$, next on another sentence in $\mathscr{S}$, and so on. Each path in the tree corresponds to a unique assignment of truth values to the sentences of $\mathscr{S}$. We check the consistency of the truth-value assignments as we go, and we close off those paths corresponding to inconsistent valuations. A semantic tree for this example is shown in Fig. 1. Closed-off paths are indicated by an ×; consistent sets of valuations are indicated in columns at the tips of their corresponding paths.

The sets of possible worlds corresponding to the different sets of consistent truth values for the sentences in $\mathscr{S}$ comprise a sample space over which we can define a probability distribution. This probability distribution specifies for each set $\mathscr{W}_i$ of possible worlds what is the probability p_i that the actual world is in $\mathscr{W}_i$. (We sometimes say, loosely, that p_i is the probability of the set $\mathscr{W}_i$ of worlds.) The individual p_i sum to 1 because the sets of possible worlds are mutually exclusive and exhaustive. The probability of any sentence S in $\mathscr{S}$ is then reasonably taken to be just the sum of the probabilities of all the sets of worlds in which S is *true.* Since we typically do not know the ordinary (*true*/*false*) truth value of S in the actual world, it is convenient to imagine a logic that has truth values intermediate between *true* and *false* and, in this logic, define the truth value of S to be the probability of S. In the context of discussing uncertain beliefs, we use the phrases *the probability of S* and *the* (*probabilistic logic*) *truth value of S* interchangeably.

Because the sets of possible worlds are identified with sets of truth values for

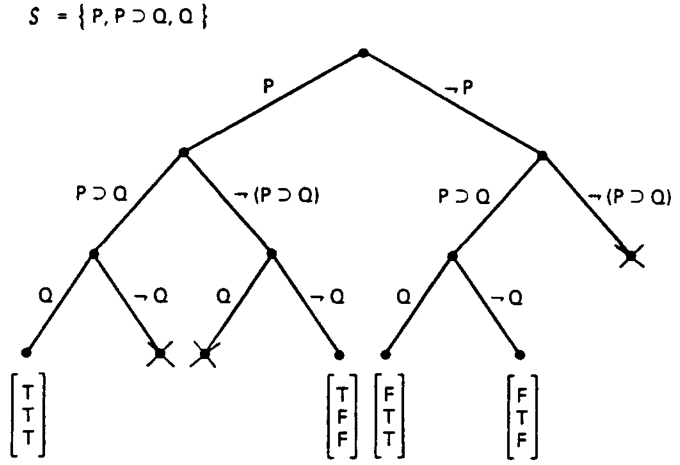

FIG. 1. A semantic tree.

sentences, these sets of possible worlds also correspond to equivalence classes of interpretations for these sentences. Each interpretation in the equivalence class associated with a set of possible worlds leads to the same set of truth values for the sentences in $\mathcal{S}$. We may sometimes refer to the possible worlds as interpretations.

It is convenient to introduce some vector notation to rephrase mathematically what we have just said. Suppose there are K sets of possible worlds for our L sentences in $\mathcal{S}$. These sets can be ordered in some arbitrary manner. Let the K-dimensional column vector $\boldsymbol{P}$ represent the probabilities of the sets of possible worlds. The ith component p_i is the probability of the ith set of possible worlds $\mathcal{W}_i$.

The sets of possible worlds themselves are characterized by the different consistent truth valuations that can be given to the sentences of $\mathcal{S}$. Let us arrange the sentences of $\mathcal{S}$ in arbitrary order and let the L-dimensional column vectors $\boldsymbol{V}_1, \boldsymbol{V}_2, \ldots, \boldsymbol{V}_K$ correspond to all of the consistent truth valuations of the sentences in $\mathcal{S}$. That is, in the ith set of worlds, $\mathcal{W}_i$, the sentences in $\mathcal{S}$ have truth valuations characterized by $\boldsymbol{V}_i$. We take each $\boldsymbol{V}_i$ to have components equal to either 0 or 1. The component $v_{ji} = 1$ if S_j has the value ***true*** in the worlds in $\mathcal{W}_i$; $v_{ji} = 0$ if S_j has the value ***false*** in the worlds in $\mathcal{W}_i$.

The K column vectors $\boldsymbol{V}_1, \ldots, \boldsymbol{V}_K$, can be grouped together, in the same order given to the sets of possible worlds, into an $L \times K$ matrix $\boldsymbol{V}$. Let us denote the probability of each sentence S_i in $\mathcal{S}$ by the components π_i of an L-dimensional column vector $\boldsymbol{\Pi}$. The probabilities of the sentences can then be related to the probabilities of the possible worlds by the following simple matrix equation:

$$\boldsymbol{\Pi} = \boldsymbol{V}\boldsymbol{P}.$$

This equation concisely expresses what we said in words earlier, namely that the probability of a sentence is the sum of the probabilities of the sets of possible worlds in which that sentence is *true*.

In using these ideas for reasoning with uncertain beliefs, we are typically not given the probabilities p_i for the different sets of possible worlds, but must instead induce them from what we are given. We consider two related types of reasoning problems. In the first, which we call ***probabilistic entailment***, we have a base set of sentences (called ***beliefs***) $\mathcal{B}$ with associated probabilities. From these, we deduce a new belief, S, and its associated probability. Using the notation we have just introduced, in this problem our set $\mathcal{S}$ of sentences consists of $\mathcal{B} \cup \{S\}$. We are given probabilities for the sentences in $\mathcal{B}$, we must solve the matrix equation for $\boldsymbol{P}$, and then use it again to compute the probability of S. There are several difficulties in carrying out these steps, and we shall discuss this problem in detail momentarily.

In the second type of problem, which is more closely related to the kind of reasoning used in expert systems, we are given a set of beliefs $\mathscr{B}$ and their associated probabilities. (We might presume that this information has been provided by an expert in the subject matter under consideration). In this problem, we might learn new information about the actual world. For example, we might learn that in the actual world, some sentence S_0 in $\mathscr{B}$ is *true* (or *false*). Or, more typically, we may learn information that gives us a new posterior probability for S_0. Given this information, we want to compute a posterior probability for some sentence of interest, S. The reasoning process in this case is an elaboration of that used in probabilistic entailment.

3. Probabilistic Entailment

In ordinary logic, *modus ponens* allows us to infer Q from P and $P \supset Q$. Also, Q is *logically entailed* by the set $\{P, P \supset Q\}$. (Modus ponens is a sound rule of inference.) In this section, we investigate the analogue of logical entailment for probabilistic logic. We shall be concerned with the question of determining the probability of an arbitrary sentence S given a set $\mathscr{B}$ of sentences and their probabilities. That is, we consider the *probabilistic entailment* of S from $\mathscr{B}$.

We begin our discussion by considering the three sentences P, $P \supset Q$, and Q. Just as we cannot consistently assign arbitrary (*true*/*false*) truth values to these three sentences, neither can we consistently assign arbitrary probability values to them. The consistent truth-value assignments are given by the columns in the matrix V, where *true* is represented by 1 and *false* is represented by 0.

$$V = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}.$$

The first row of the matrix gives truth values for P in the four sets of possible worlds. The second row gives truth values for $P \supset Q$, and the third row gives truth values for Q. Probability values for these sentences are constrained by the matrix equation

$$\boldsymbol{\Pi} = \boldsymbol{VP}$$

and by the rules of probability, $\Sigma_i p_i = 1$ and $0 \leqslant p_i \leqslant 1$ for all i.

These constraints can be given a simple geometric interpretation. The matrix equation maps a space of probability values over possible worlds into a space of probability values over sentences. The mapping is linear and therefore maps extreme values of $\boldsymbol{P}$ into extreme values of $\boldsymbol{\Pi}$. The extreme values of $\boldsymbol{P}$ are those for which individual values of p_i are equal to 1. But only one p_i in $\boldsymbol{P}$ can be equal to 1; the rest must be 0. Thus there are four extreme $\boldsymbol{P}$ vectors, namely [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], and [0, 0, 0, 1]. (These are all column vectors: we write them in row format in running text.) The extreme $\boldsymbol{\Pi}$ vectors corresponding to these extreme $\boldsymbol{P}$ vectors are simply the columns of the $\boldsymbol{V}$ matrix. This result is not surprising; when the sentences are given an interpretation corresponding to one of the sets of possible worlds, then the truth values of the sentences are the truth values assigned in that possible world. The principal benefit of this analysis comes from observing that, for arbitrary values of $\boldsymbol{P}$, $\boldsymbol{\Pi}$ must lie within the convex hull of the extreme values of $\boldsymbol{\Pi}$.

A picture of this mapping is shown in Fig. 2. The extreme values of $\boldsymbol{\Pi}$ are indicated by solid dots. Consistent values for the probabilities of the three sentences must lie in the convex hull of these points which is the solid region shown in the figure.

Now suppose we are given the probability values for the sentences P and $P \supset Q$. In terms of our notation, the probability of P, denoted by $p(P)$ is π_1; the probability of $P \supset Q$, denoted by $p(P \supset Q)$ is π_2. We can see from Fig. 2 that π_3 or $p(Q)$ must then lie between the two bounding planes shown in the figure. Calculating these bounds analytically results in the following inequality:

$$p(P \supset Q) + p(P) - 1 \leqslant p(Q) \leqslant p(P \supset Q) .$$

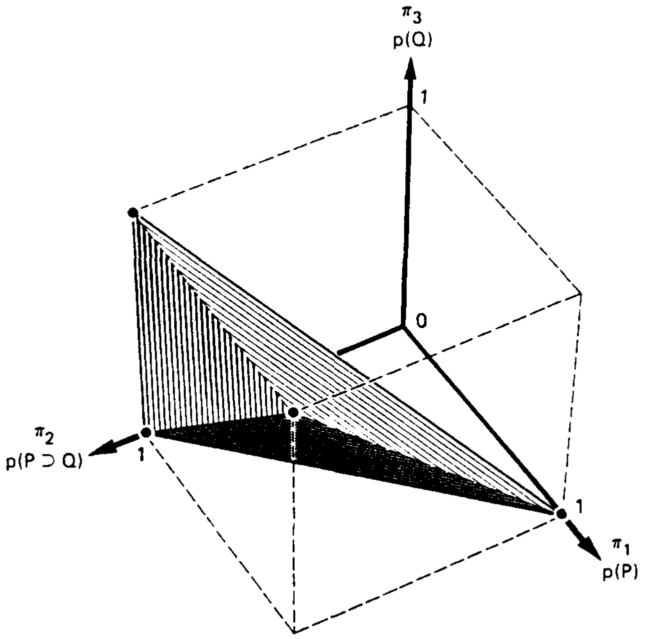

FIG. 2. The convex region of consistent probability values for P, $P \supset Q$, and Q.

(setting $P(Q)$ equal to its lower and upper bounds gives equations for the lower and upper bounding planes of Fig. 2.)

This example reveals some interesting points about probabilistic logic. First, just as it is possible to assign inconsistent *true-false* truth values to sentences, it is also possible to assign them inconsistent probabilities (that is, *probabilistic* truth values). For the sentences $\{P, P \supset Q, Q\}$ any assignment outside the convex region shown in Fig. 2 is inconsistent. (Assignment of consistent subjective probabilities to sentences is a well-known problem in designing expert systems. A solution suggested by our geometric view would be to move an inconsistent Π vector to a "nearby" point in the consistent region, perhaps preferring larger adjustments to the probabilities of some sentences than to the probabilities of others.) Second, even if consistent probabilities are assigned to P and to $P \supset Q$, the probability of Q is not, in general, determined uniquely, but is bounded by the expressions given above. Thus, we can expect that probabilistic entailment will, as a rule, merely bound (rather than precisely specify) the probability of the entailed sentence.

Solving probabilistic entailment problems can be done by adding the entailed sentence, S, to the base set of beliefs $\mathcal{B}$, computing the consistent sets of truth values for this expanded set (the columns of V), computing the convex hull of these points, and then entering this convex hull along coordinates given by the probabilities of the sentences in $\mathcal{B}$ to find the probability bounds on S. The three sentences of our example produced a simple, 3-dimensional probabilistic entailment problem. In general, when we have L sentences and K sets of possible worlds, we will have to find the bounding hyperplanes of a K-vertex solid in L dimensions.

Before continuing with our discussions about solution methods for probabilistic entailment problems, let us consider one more example small enough to permit three-dimensional geometric insight. This time we consider a simple problem in first-order logic.

Let $\mathcal{B}$ be the set $\{(\exists y)P(y), (\forall x)[P(x) \supset Q(x)]\}$, and let S be the sentence $(\exists z)Q(z)$. We are given probabilities for the sentences in $\mathcal{B}$ and want to compute bounds on the probability of $(\exists z)Q(z)$.

We first create $\mathcal{S}$ by adding S to $\mathcal{B}$ and then compute the consistent sets of truth values for the sentences in $\mathcal{S}$ by the semantic-tree method illustrated in Fig. 3. In that figure, we have represented sentences and their negations in Skolem form; A, B, and C are Skolem constants. Paths corresponding to inconsistent sets of truth values are closed off by ×. The consistent sets of truth values (in 0, 1 notation) are indicated in columns at the tips of their corresponding paths. These column vectors are shown as points in Fig. 4, and their convex hull is indicated. This region contains all consistent probabilities for the three sentences in $\mathcal{S}$. In terms of consistent probability values for $(\exists y)P(y)$ and $(\forall x)[P(x) \supset Q(x)]$, the bounds on $p[(\exists z)Q(z)]$ are given by:

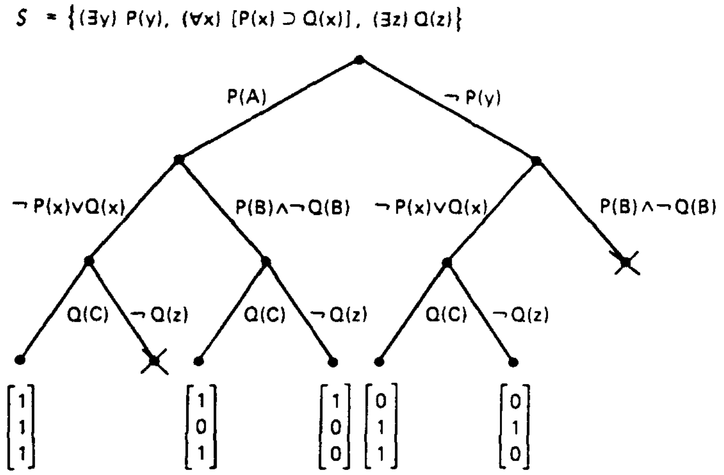

FIG. 3. A semantic tree for a problem in first-order logic.

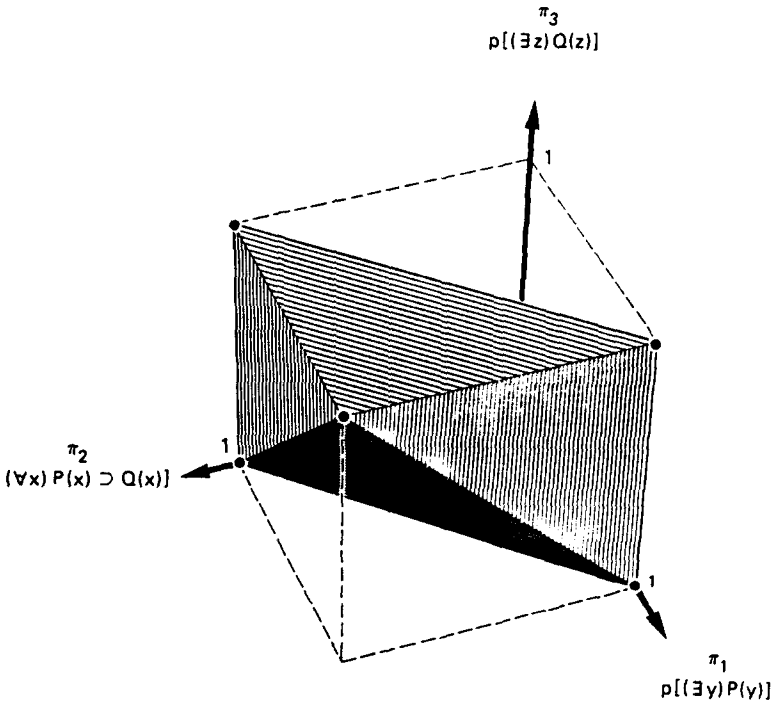

FIG. 4. The region of consistent probability values.

$$p[(\exists y)P(y)] + p[(\forall x)[P(x) \supset Q(x)]] - 1 \leqslant p[(\exists z)Q(z)] \leqslant 1\,.$$

As is apparent from Fig. 4, these bounds loosen markedly as we move away from

$$p[(\exists y)P(y)] = 1 \quad \text{and} \quad p[(\forall x)[P(x) \supset Q(x)]] = 1\,.$$

In principle, the probabilistic entailment problem can be solved by linear programming methods, but the size of problems encountered in probabilistic reasoning is usually much too large to permit a direct solution. Our focus will be to look for solution methods, sometimes approximate ones, that reduce the full problem to smaller problems of practical size. We first outline a canonical form for setting up probabilistic entailment problems. We have already mentioned that we arbitrarily order the sentences in $\mathcal{S}$ to permit specifying the consistent truth values as column vectors, V_i. We include the constraint that $\sum_i p_i = 1$ by adding a row vector of all ones as the top row of the matrix V. This row can be made to appear in V merely by including the tautology T as the first element of $\mathcal{S}$. (T has value *true* in all possible worlds.) By convention, we include the entailed sentence, S, as the last sentence in $\mathcal{S}$; thus the last row of V represents the consistent truth values of S in the various sets of possible worlds. The other rows of V (except the first and last) then represent the consistent truth values for the sentences in the base set of beliefs, $\mathcal{B}$.

We assume that we are given consistent probability values for all but the last sentence in $\mathcal{S}$. (The probability of the first sentence, namely T, is 1.) We compute the $L \times K$ matrix V (perhaps using the semantic tree method). Next we consider the matrix equation

$$\boldsymbol{\Pi} = \boldsymbol{VP}\,.$$

The K-dimensional column vector, $\boldsymbol{P}$, is unknown—as is the last element of $\boldsymbol{\Pi}$. To solve for $\boldsymbol{P}$ formally, we first construct the $(L-1) \times K$ matrix $\boldsymbol{V}'$ from $\boldsymbol{V}$ by eliminating the last row, call it $\boldsymbol{S}$, of $\boldsymbol{V}$. We construct the $(K-1)$-dimensional column vector $\boldsymbol{\Pi}'$ by eliminating the last element of $\boldsymbol{\Pi}$. Now we attempt to solve $\boldsymbol{\Pi}' = \boldsymbol{V}'\boldsymbol{P}$ for $\boldsymbol{P}$. Having done so, we can compute $\pi_L = p(S) = \boldsymbol{SP}$.

Usually the equation $\boldsymbol{\Pi}' = \boldsymbol{V}'\boldsymbol{P}$ is underdetermined and permits many solutions for $\boldsymbol{P}$. In these cases, assuming $\boldsymbol{V}$ is small enough to permit computations on it, we will be interested in those solutions that give bounds for $p(S)$. We will postpone until later a discussion of approaches toward solving problems with impractically large $\boldsymbol{V}$ matrices.

4. Computations Appropriate for Small Matrices

Using the notation of the last section, we denote the last row of $\boldsymbol{V}$ by the row vector $\boldsymbol{S}$. This vector gives the truth values for the entailed sentence S that are consistent with the truth values for the other sentences in $\mathcal{S}$. Then the probability, $p(S)$, of S is given by $\boldsymbol{SP}$ where $\boldsymbol{P}$ is a solution to $\boldsymbol{\Pi}' = \boldsymbol{V}'\boldsymbol{P}$. Analogously, we might denote the other rows in $\boldsymbol{V}$ by the row vectors $\boldsymbol{S}_i$. $\boldsymbol{S}_L = \boldsymbol{S}$ and recall that $\boldsymbol{S}_1 = [1, 1, \ldots, 1]$. (This notation is suggestive; the rows of $\boldsymbol{V}$ represent the sentences in $\mathcal{S}$ in terms of all possible truth values that are consistent with the truth values for the other sentences.)

In certain degenerate cases we can compute a unique $\boldsymbol{P}$ given $\boldsymbol{V}'$ and $\boldsymbol{\Pi}'$. For example, if $\boldsymbol{S}$ happens to be identical to the ith row of $\boldsymbol{V}'$, then $\boldsymbol{SP} = \pi_i$. More generally, if $\boldsymbol{S}$ can be written as a linear combination of rows of $\boldsymbol{V}'$, then $\boldsymbol{SP}$ can be simply written as the same linear combination of the π_i. For example, this method can be used to establish the following identities:

$$p(Q) = p(P) + p(P \supset Q) - p(Q \supset P),$$

$$p(Q) = p(P \supset Q) + p(\neg P \supset Q) - 1.$$

(To illustrate, we observe that in the first of these, after setting up the matrix $\boldsymbol{V}$, P is represented by the row vector $[1, 1, 0, 0]$, $P \supset Q$ by $[1, 0, 1, 1]$, $Q \supset P$ by $[1, 1, 0, 1]$, and Q by $[1, 0, 1, 0]$. The last vector is the sum of the first two minus the third.)

We might also imagine that if $\boldsymbol{S}$ can be approximated (in some sense) by a linear combination of the rows of $\boldsymbol{V}'$, then $\boldsymbol{SP}$ can be approximated by the same linear combination of the π_i. Such approximations may well be useful and worth looking for. An approximation that we might consider is $\boldsymbol{S}^*$, the projection of $\boldsymbol{S}$ onto the subspace defined by the row vectors of $\boldsymbol{V}'$. By assumption, $\boldsymbol{S}^*$ will be some linear combination of the row vectors in $\boldsymbol{V}'$, say:

$$\boldsymbol{S}^* = \sum_{i=1}^{L-1} c_i \boldsymbol{S}_i .$$

An approximation to the probability of S could then be taken to be $\boldsymbol{S}^*\boldsymbol{P}$, which is given by:

$$\boldsymbol{S}^*\boldsymbol{P} = \sum_{i=1}^{L-1} c_i \boldsymbol{S}_i \boldsymbol{P} = \sum_{i=1}^{L-1} c_i \pi_i .$$

Suppose we use this method to approximate the probability of Q given the sentences P, with probability $\pi_2 = p(P)$, and $P \supset Q$, with probability $\pi_3 = p(P \supset Q)$. (Recall that we include the sentence T, with probability $\pi_1 = 1$, in $\mathcal{S}$.) $\boldsymbol{V}'$ and $\boldsymbol{\Pi}'$ are then given by:

$$\boldsymbol{V}' = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix}, \qquad \boldsymbol{\Pi}' = \begin{bmatrix} 1 \\ \pi_2 \\ \pi_3 \end{bmatrix},$$

The row vector representation for Q (that is, the last row of V) is $\boldsymbol{Q} = [1, 0, 1, 0]$, and its projection onto the subspace defined by the row vectors of V' is $\boldsymbol{Q}^* = [1, 0, \frac{1}{2}, \frac{1}{2}]$. The coefficients c_i are given by $c_1 = -\frac{1}{2}$, $c_2 = \frac{1}{2}$, and $c_3 = 1$. Using these, the approximate value for $p(Q)$ is:

$$-\tfrac{1}{2} \times \pi_1 + \tfrac{1}{2} \times \pi_2 + 1 \times \pi_3 = -\tfrac{1}{2} + \tfrac{1}{2} p(P) + p(P \supset Q) .$$

It is interesting to note that this value happens to be midway between the two bounds on $p(Q)$ established in our earlier example.

Another technique that can be used when we are given underdetermined (but consistent) $\boldsymbol{V}'$ and $\boldsymbol{\Pi}'$ is to select from among the possible solutions for $\boldsymbol{P}$ that $\boldsymbol{P}$ with maximum entropy. This distribution assumes the minimum additional information about $\boldsymbol{P}$ given the sentences in $\mathcal{B}$ and their probabilities.

The *entropy* of a probability distribution, $\boldsymbol{P}$, is defined to be:

$$H = - \sum_i p_i \log p_i = - \boldsymbol{P}^{\text{t}} \log \boldsymbol{P},$$

where $\boldsymbol{P}^{\text{t}}$ is the *transpose* (i.e., the row vector form) of the column vector $\boldsymbol{P}$, and $\log \boldsymbol{P}$ is a (column) vector whose components are the logarithms of the corresponding components of $\boldsymbol{P}$.

To maximize H, by varying $\boldsymbol{P}$, subject to the constraint that $\boldsymbol{\Pi}' = \boldsymbol{V}'\boldsymbol{P}$, we use the method of Lagrange multipliers from the calculus of variations (following Cheeseman [16]). First we write H as follows:

$$\begin{aligned} H = -\boldsymbol{P}^{\text{t}} \log \boldsymbol{P} + l_1(\pi_1 - \boldsymbol{S}_1\boldsymbol{P}) + l_2(\pi_2 - \boldsymbol{S}_2\boldsymbol{P}) + \cdots \\ + l_{(L-1)}(\pi_{(L-1)} - \boldsymbol{S}_{(L-1)}\boldsymbol{P}), \end{aligned}$$

where $l_1, \ldots, l_{(L-1)}$ are Lagrange multipliers; $\pi_1, \ldots, \pi_{(L-1)}$ are the components of $\boldsymbol{\Pi}'$, and $\boldsymbol{S}_1, \ldots, \boldsymbol{S}_{(L-1)}$ are the row vectors of $\boldsymbol{V}'$.

Differentiating this expression with respect to p_i and setting the result to zero yields:

$$- \log p_i - 1 - l_1 v_{1i} - \cdots - l_{(L-1)} v_{(L-1)i} = 0,$$

where v_{ji} is the ith component of the jth row vector in $\boldsymbol{V}'$.

Thus, the distribution that maximizes the entropy has components

$$p_i = \mathrm{e}^{-1}\, \mathrm{e}^{-(l_1 v_{1i})} \cdots \mathrm{e}^{-(l_{(L-1)} v_{(L-1)i}}) .$$

Cheeseman [16] used the following definitions to simplify this expression:

$$a_1 = \mathrm{e}^{-1}\, \mathrm{e}^{-(l_1)}, \qquad a_j = \mathrm{e}^{-(l_j)}, \quad j = 2, \ldots, (L-1).$$

We then see that each p_i can be written as a product of some of the a_j, where a_j is included in p_i if v_{ji} is 1 and is not included otherwise. We note that a_1 is included in each of the p_i because $v_{1i} = 1$ for all i.

Now we can solve directly for the a_j by substituting these expressions for p_i as components of $\boldsymbol{P}$ and solving the equation $\boldsymbol{\Pi}' = \boldsymbol{V}'\boldsymbol{P}$ for the a_j.

Let us calculate the maximum-entropy distribution given the sentences P with probability π_2 and $P \supset Q$ with probability π_3. As before, $\boldsymbol{V}'$ and $\boldsymbol{\Pi}'$ are given by:

$$\boldsymbol{V}' = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix}, \qquad \boldsymbol{\Pi}' = \begin{bmatrix} 1 \\ \pi_2 \\ \pi_3 \end{bmatrix}.$$

We can read down the columns of $\boldsymbol{V}'$ to express each (entropy-maximizing) p_i in terms of products of the a_j:

$$p_1 = a_1a_2a_3, \qquad p_2 = a_1a_2, \qquad p_3 = a_1a_3, \qquad p_4 = a_1a_3.$$

Using these values in $\boldsymbol{\Pi}' = \boldsymbol{V}'\boldsymbol{P}$ yields the equations:

$$a_1a_2a_3 + a_1a_2 + 2a_1a_3 = 1,$$
$$a_1a_2a_3 + a_1a_2 = \pi_2, \qquad a_1a_2a_3 + 2a_1a_3 = \pi_3.$$

Solving yields:

$$a_1 = (1-\pi_2)(1-\pi_3)/2(\pi_2+\pi_3-1),$$
$$a_2 = 2(\pi_2+\pi_3-1)/(1-\pi_2), \qquad a_3 = (\pi_2+\pi_3-1)/(1-\pi_3).$$

Thus, the entropy-maximizing $\boldsymbol{P}$ is given by:

$$\boldsymbol{P} = \begin{bmatrix} \pi_2+\pi_3-1 \\ 1-\pi_3 \\ \frac{1}{2}(1-\pi_2) \\ \frac{1}{2}(1-\pi_2) \end{bmatrix}.$$

Using this probability distribution, we see that the probability of Q is

$$[1, 0, 1, 0]\boldsymbol{P} = \tfrac{1}{2}\pi_2 + \pi_3 - \tfrac{1}{2} = \tfrac{1}{2}p(P) + p(P \supset Q) - \tfrac{1}{2}.$$

(This happens to be the same value calculated by the "projection approximation" method!)

5. Dealing with Large Matrices

The techniques described in the last section all involved computing a possible-worlds probability vector, P, from V' and Π'. When V' is as large as it might be with even, say, a dozen or so sentences, these methods become impractical. Perhaps there are much simpler techniques for computing the approximate probability of a sentence S probabilistically entailed by $\mathcal{B}$.

Some approximation methods are based on subdividing $\mathcal{B}$ into smaller sets. Suppose for example that $\mathcal{B}$ could be partitioned into two parts, namely $\mathcal{B}_1$ and $\mathcal{B}_2$ with no atom that occurs in $\mathcal{B}_1$ occurring in $\mathcal{B}_2$ or in S. Clearly $\mathcal{B}_1$ could be eliminated from $\mathcal{B}$ without any effect on probabilistic entailment calculations for S. In this case, we say that the subset, $\mathcal{B}_2$, is a *sufficient* subset for S.

Or, suppose two sentences, S_1 and S_2, could be found such that a subset of $\mathcal{B}$, say $\mathcal{B}_1$, was sufficient for S_1 and another subset, say $\mathcal{B}_2$, was sufficient for S_2. Then we could split the probabilistic entailment of S from $\mathcal{B}$ into two smaller problems: first compute the probabilistic entailments of S_1 from $\mathcal{B}_1$ and of S_2 from $\mathcal{B}_2$. Next, compute the probabilistic entailment of S from $\{S_1, S_2\}$. The idea here is to find sentences, S_1 and S_2, such that, together, they "give as much information" about S as does $\mathcal{B}$. In this case, $\mathcal{B}_1$ and $\mathcal{B}_2$ are similar to what have been called *local event groups* [13]. This method, of course, is only approximate; its accuracy depends on how well the probabilities of S_1 and S_2 determine the probability of S.

We next suggest a process for finding an "approximate" (and smaller) matrix for V' given $\mathcal{B}$, Π', and S. This approximate matrix, which we denote by V'^*, can be made sufficiently small to permit practical computation of approximate probabilistic entailment. The approximation is exact in the non-probabilistic case when Π' consists of only ones and zeros. It can be made as precise as desired by making V'^* larger.

We follow the usual process for computing the matrix V'—except in computing V'^* we do not include *all* of the consistent sets of truth values. Instead, we construct a smaller set that includes only vectors "close to" the given Π'.

We first compute an approximate matrix, V^*, as follows:

(1) Construct a *true-false* vector, Π'_b, from Π' by changing to 1 the values of those components π_i whose values are greater than or equal to $\frac{1}{2}$. Change the values of the other components to 0.

(2) If S can have value *true* consistent with the truth values for the sentences in $\mathcal{B}$ given by Π'_b, then include in V^* the vector formed from Π'_b by appending to it a final component equal to 1. If S can have value *false* consistent with the valuations for the sentences in $\mathcal{B}$ given by Π'_b, then include in V^* the vector formed from Π'_b by appending to it a final component equal to 0.

(3) Reverse the values of the components of Π'_b one at a time, two at a time, and so on, starting with those components whose corresponding com-

ponents in Π' have values *closest* to $\frac{1}{2}$. For each of the altered *true-false* vectors thus obtained that represent consistent *true-false* truth values over $\mathscr{B}$, add new vector(s) to V^* according to the procedure described in step (2) immediately above. We use as many of these consistent, altered vectors as computational resources permit. The more vectors used, the better the approximation. (The ordering of the column vectors in V^* is arbitrary.)

We next construct the matrix V'^* by deleting the last row of V^*. (We take this last row to be an approximate vector representation S^* for the sentence S.)

It should be clear that as we include more and more vectors in V^*, it approaches V, and V'^* approaches V'. Also, if Π' is the vector with components all equal to 1, then $\Pi' = \Pi'_b$. In that case, if S logically follows from $\mathscr{B}$, V'^* need have only a single column (of 1's), $P = [1]$, $S^* = [1]$, and $p(S) = 1$. If $\neg S$ logically follows from $\mathscr{B}$, V'^* still need have only a single column (of all 1's), $P = [1]$, $S^* = [0]$, and $p(S) = 0$. If both S and $\neg S$ are consistent with $\mathscr{B}$, then V'^* would have two identical columns (of all ones), P could have permissible solutions $[1, 0]$ and $[0, 1]$, $S^* = [1, 0]$, and $p(S)$ could range consistently between 0 and 1.

Thus, our approximation behaves well at the limits of large V'^* and at the non-probabilistic extreme. Continuity arguments suggest that performance ought to degrade only gradually as we depart from these limits, although the method has not yet been tested on large examples. If we recall that the region of consistent probability vectors, Π, occupies the convex hull of the region defined by the extreme $(0, 1)$ probability vectors, we note that our approximation method constructs an approximate region, namely the convex hull of just those extreme vectors that are "close to" the given probability vector, Π'. We suspect that the more uncertain are the sentences in $\mathscr{B}$, the more vectors will have to be included in V^* to get accurate entailment.

6. Probabilities Conditioned on Additional Information

In typical applications of these ideas, experts in the subject matter of the application would provide us with a base set $\mathscr{B}$ of beliefs and their probabilities, Π. We would then like to use these uncertain beliefs to calculate the probability of some sentence, S, given information about some sentence, S_0. The information about S_0 might be that S_0 is *true*, or that it is *false*, or that it has some probability, $p(S_0)$. In general, neither S nor S_0 need be in $\mathscr{B}$—although either or both could be.

Suppose we are given that S_0 is *true*. Then we want to calculate the *conditional probability* $p(S \mid S_0)$. Using Bayes' rule this conditional probability is:

$$p(S \mid S_0) = \frac{p(S, S_0)}{p(S_0)} = \frac{p(S \wedge S_0)}{p(S_0)} .$$

The probabilities $p(S \wedge S_0)$ and $p(S_0)$ can be calculated using any of the methods described in this paper. (The probability of S given S_0 is just the sum of the probabilities of each of the possible worlds in which both S and S_0 are true normalized by dividing by the probability of S_0.) If the method gives unique values for $p(S \wedge S_0)$ and $p(S_0)$, then the conditional probability will also have a unique value. If the method gives bounds on the probabilities, then the conditional probability will also be bounded.

We can derive a similar expression if we are given that S_0 is *false*:

$$p(S \mid \neg S_0) = \frac{p(S, \neg S_0)}{p(\neg S_0)} = \frac{p(S \wedge \neg S_0)}{p(\neg S_0)}.$$

Often we do not know whether S_0 is *true* or *false* but might instead have only a *posterior* probability for S_0, say $p(S_0 \mid S_0')$. In this case, we associate the sentence S_0' with the event of having received some information about S_0 that permits us to assign the probability $p(S_0 \mid S_0')$ to S_0. (We must not confuse $p(S_0 \mid S_0')$ with $p(S_0)$. The former is a new or posterior probability after having learned specific information about a particular case; the latter is the prior probability based on general expert knowledge.)

Now we can compute an expression for $p(S \mid S_0')$ as a weighted average of $p(S \mid S_0)$ and $p(S \mid \neg S_0)$. Assuming that

$$p(S \mid S_0, S_0') = p(S \mid S_0) \quad \text{and} \quad p(S \mid \neg S_0, S_0') = p(S \mid \neg S_0),$$

the expression for the posterior probability for S (given S_0') becomes:

$$p(S \mid S_0') = p(S \mid S_0)p(S_0 \mid S_0') + p(S \mid \neg S_0)p(\neg S_0 \mid S_0').$$

Substituting the expressions we had derived earlier for $p(S \mid S_0)$ and $p(S \mid \neg S_0)$ we obtain:

$$p(S \mid S_0') = \frac{p(S \wedge S_0)}{p(S_0)} p(S_0 \mid S_0') + \frac{p(S \wedge \neg S_0)}{p(\neg S_0)} p(\neg S_0 \mid S_0').$$

Our methods usually justify only the calculation of bounds on probabilities. Indeed, we may only know bounds on the probabilities of the sentences in $\mathcal{B}$. If the probability of a sentence S is known only to lie between a lower bound, π_l, and an upper bound, π_u, then the difference $\pi_u - \pi_l$ expresses our *ignorance* about S. Using *upper* and *lower* probabilities gives us a method to distinguish between situations in which our beliefs can be described by a single probability number and those in which we have even less information. To have good reason to believe, for example, that a particular treatment method for a certain disease is effective in half the cases is to have arguably more information than to have no justifiable beliefs at all about its effects. In the latter case, the appropriate lower and upper probabilities would be 0 and 1, respectively.

All of the methods described in this paper can be easily modified to deal with sentences with upper and lower probabilities. In calculating bounds on the probability of some sentence S, one first uses those extreme values of probabilities that give one of the bounds and then the extremes that give the other. Grosof [18] has shown that an important special case of the Shafer–Dempster procedure for assigning *mass weights* to sentences is itself a special case of our procedure adapted to deal with upper and lower probabilities.

7. Conclusions

We have presented a straightforward generalization of the ordinary *true-false* semantics for logical sentences to a semantics that allows probabilistic values on sentences. Although implementation of the full procedure for probabilistic entailment would usually be computationally impractical, we also described a simple approximation method that might be appropriate for realistic applications in expert systems. Such applications would also require a technique for dealing with inconsistent probability values supplied by the expert and user. One possibility would be to move an inconsistent Π vector to a "nearby" point in the consistent region, perhaps preferring larger adjustments to "user probabilities" than to "expert probabilities." The technique can also be applied in an obvious way when the probabilities of sentences are merely bounded rather than having definite values.

Some have proposed that nonmonotonic reasoning be performed by probabilistic deductions of various kinds. In this connection, we point out that probabilistic entailment, as presented here, is actually *monotonic* in that constraints on the probability values of sentences (imposed by adding new uncertain "facts") can only *reduce* the region of consistent valuations. Adding such constraints never results in *adding* to the region of consistent valuations, and therefore no different valuations can result from such additional information.

ACKNOWLEDGMENT

This paper is based on a talk given by the author on December 8, 1983 at a symposium honoring Marvin Denicoff on the occasion of his retirement from the Office of Naval Research. Stan Rosenschein helped improve an early version by pointing out that one need only consider equivalence classes of interpretations (equivalent with respect to their consistent valuations over the finite set of sentences $\mathcal{S}$). I also profited from discussions with Michael Georgeff, Peter Cheeseman, Kurt Konolige, John Lowrance, and Benjamin Grosof. The referees made many helpful suggestions.

REFERENCES

1. Lukasiewicz, J., Logical foundations of probability theory, in: L. Berkowski, (Ed.), *Jan Lukasiewicz Selected Works*, (North-Holland, Amsterdam, 1970) 16–43.

2. Carnap, R., The two concepts of probability, in: *Logical Foundations of Probability* (University of Chicago Press, Chicago, IL, 1950) 19–51.
3. Hempel, C.G., Studies in the logic of confirmation, in: *Aspects of Scientific Explanation and other Essays in the Philosophy of Science* (The Free Press, New York, 1965) 3–51.
4. Suppes, P., Probabilistic inference and the concept of total evidence, in: Hintikka and P. Suppes (Eds.), *Aspects of Inductive Logic* (North-Holland, Amsterdam, 1966) 49–65.
5. Dempster, A.P., A generalization of Bayesian inference, *J. Roy. Statist. Soc. B* **30** (1968) 205–247.
6. Shafer, G.A., *Mathematical Theory of Evidence* (Princeton University Press, Princeton, NJ, 1979).
7. Adams, E.W. and Levine, H.F., On the uncertainties transmitted from premises to conclusions in deductive inferences, *Synthese* **30** (1975) 429–460.
8. Zadeh, L.A., Fuzzy logic and approximate reasoning, *Synthese* **30** (1975) 407–428.
9. Shortliffe, E.H., Computer-based medical consultations: MYCIN (Elsevier, New York, 1976).
10. Duda, R.O., Hart, P.E. and Nilsson, N.J., Subjective Bayesian methods for rule-base inference systems, in: *Proceedings 1976 National Computer Conference, AFIPS* **45** (1976) 1075–1082; reprinted in: B.W. Webber and N.J. Nilsson (Eds.), *Readings in Artificial Intelligence* (Tioga, Palo Alto, CA, 1981).
11. Lowrance, J.D. and Garvey, T.D., Evidential reasoning: a developing concept, in: *IEEE 1982 Proceedings International Conference on Cybernetics and Society* (October 1982) 6–9.
12. Lowrance, J.D. and Garvey, T.D., Evidential reasoning: an implementation for multisensor integration, SRI AI Center Technical Note 307, SRI International, Menlo Park, CA, 1983.
13. Lemmer, J.F. and Barth, S.W., Efficient minimum information updating for Bayesian inferencing in expert systems, in: *Proceedings Second National Conference on Artificial Intelligence*, Pittsburgh, PA (1982) 424–427.
14. Lemmer, J.F., Generalized Bayesian updating of incompletely specified distributions, Working Paper, Par Technology Corporation, New Hartford, NY, November 30, 1982.
15. Konolige, K.G., An information-theoretic approach to subjective Bayesian inference in rule-based systems, SRI Working Paper, 1982; a revision of Appendix D: Bayesian methods for updating probabilities, in: Duda, R.O. et al., A computer-based consultant for mineral exploration, Technical Rept., SRI International, Menlo Park, CA, 1978.
16. Cheeseman, P., A method of computing generalized Bayesian probability values for expert systems, in *Proceedings Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, Fed. Rep. Germany (William Kaufmann, Los Altos, CA, 1983).
17. Halpern, J.Y. and Rabin, M., A logic to reason about likelihood, IBM Research Rept. RJ 4136 (45774), December 19, 1983; also: *ACM Proceedings Fifteenth Annual ACM Symposium on Theory of Computing*, Boston, MA (ACM Order No. 508830) (1983) 310–319.
18. Grosof, B.N., An inequality paradigm for probabilistic knowledge, in: *Proceedings AAAI/IEEE Workshop on Uncertainty and Probability in Artificial Intelligence*, Los Angeles, CA, 1985.

Received November 1984; revised version received August 1985