Anytime Deduction for Probabilistic Logic

Alan M. Frisch* **and Peter Haddawy†**
Department of Computer Science and Beckman Institute
University of Illinois
405 North Mathews Ave.
Urbana, IL 61801
U.S.A.

October 21, 1992

Abstract

This paper proposes and investigates an approach to deduction in probabilistic logic, using as its medium a language that generalizes the propositional version of Nilsson's probabilistic logic by incorporating conditional probabilities. Unlike many other approaches to deduction in probabilistic logic, this approach is based on inference rules and therefore can produce proofs to explain how conclusions are drawn. We show how these rules can be incorporated into an *anytime deduction procedure* that proceeds by computing increasingly narrow probability intervals that contain the tightest entailed probability interval. Since the procedure can be stopped at any time to yield partial information concerning the probability range of any entailed sentence, one can make a tradeoff between precision and computation time. The deduction method presented here contrasts with other methods whose ability to perform logical reasoning is either limited or requires finding all truth assignments consistent with the given sentences.

To appear in *Artificial Intelligence*.

*Current address: Department of Computer Science, University of York, York YO1 5DD, United Kingdom. Phone: +44 (904) 432745. Email: frisch@minster.york.ac.uk.

†Current address: Department of Electrical Engineering and Computer Science, University of Wisconsin-Milwaukee, Milwaukee, WI 53201, U.S.A. Phone: +1 (414) 229-4955. Email: haddawy@cs.uwm.edu.

1 Introduction

For a long time a rift has existed within AI between logic-based and probability-based approaches to knowledge representation. Recently AI researchers have attempted to bridge this rift by developing formal languages integrating probability with first-order logic [27; 3; 12; 17; 18]. Interest in this problem as a research topic for AI was sparked by Nilsson's [27] paper on probabilistic logic.[1]

Probabilistic logic is appealing in many ways. Its well-defined model theory clarifies the meaning of the probability attached to a sentence. The model theory also provides clear definitions of consistency, validity, and entailment independent of any system of inference. Thus, what uncertainty researchers have called combination and propagation rules become logical rules of inference and the model theory decisively settles any questions about whether one inference rule or another is justified.

Accompanying his presentation of probabilistic logic, Nilsson presents a procedure for computing *probabilistic entailment*. Given a set of propositional or first-order sentences and their associated probabilities, this procedure computes a range of probabilities within which the probability of some given target sentence must lie.[2] The procedure operates by first finding all consistent assignments of truth values to the given sentences and then using these assignments to set up a system of linear equations that must be solved.

Nilsson's proposal for computing probabilistic entailment has two drawbacks. The foremost shortcoming with his approach is its dependence on determining all consistent truth value assignments for a set of sentences, a problem that is NP-complete for propositional logic and undecidable for first-order logic. Hence, in the first-order case the method may not proceed even as far as setting up the system of linear equations, in which case the method yields no information about the probability of the target sentence.

The second drawback of Nilsson's procedure is that it does not produce a proof in the usual sense. A traditional deductive system based on inference rules yields a proof that explains the line of reasoning used and justifies the conclusion. Such proofs are useful as explanations to humans and as input to explanation-based learning algorithms. These drawbacks are inherited by subsequently developed elaborations of Nilsson's basic approach [35; 32].

A related approach to computing probabilities is Bundy's [4] incidence calculus. This approach was proposed prior to the publication of Nilsson's probabilistic logic and so was not described in terms of his semantics. But if viewed in terms of the semantics of probabilistic logic, it can now be seen as a method of computing probabilistic entailment. Bundy's approach, like Nilsson's, requires performing an initial impractical computation.

Another approach to computing probabilistic entailment is Quinlan's network propagation approach [31]. This method has its own limitations. Quinlan's method works with a static network representing a fixed set of formulas and relationships among them. Because it is unable to form new formulas as is done in proof systems, it cannot pursue any line of reasoning involving formulas other than those built into the network. The network can only propagate probability constraints among its existing formulas. Amarger, Dubois and Prade

[1] Nilsson's [26] retrospective on the paper mentions some of the subsequent research on this topic.

[2] For an earlier investigation of probabilistic entailment for both propositional and first-order logic see the article by Adams and Levine [1].

[2] extend Quinlan's approach with rules for the derivation of conditional probabilities, as well as means of introducing new nodes. However, the new nodes are limited to conjunctions and disjunctions of other nodes.

The main contribution of this paper is to describe an alternative approach to deduction in probabilistic logic and to investigate some systems based on this approach. We report on progress we have made towards developing a proof system based on inference rules for a generalization of propositional probabilistic logic. We show how these rules can be incorporated into an *anytime deduction procedure*. The anytime property means that the inference process proceeds by computing increasingly narrow probability intervals that contain the tightest entailed probability interval. Thus the procedure can be stopped at any time to yield partial information concerning the probability range of the target sentence. Consequently, with an anytime deduction system one can tradeoff between precision and computation time. Even if an anytime system is not complete it may still be preferable to Nilsson's inference method. In the first-order case, it may provide at least partial information in situations where Nilsson's approach provides no information at all. In the propositional case it can provide useful partial information without performing the exponential computation required by Nilsson's method. Our approach is superior to the network propagation approach since it allows sentences to be freely combined to produce new sentences, whereas all network methods either limit or prohibit the production of new formulas.

Building upon our previous work [16], this paper 1) defines a language, called $\mathcal{L}_{PL}$, that generalizes the propositional version of Nilsson's probabilistic logic by incorporating conditional probabilities, 2) presents sound, quasi-tight[3] inference rules for $\mathcal{L}_{PL}$, and 3) identifies a subset of $\mathcal{L}_{PL}$ for which a subset of the inference rules is complete. Several examples illustrating the roles of the various inference rules as well as their anytime nature are provided. We relate our results to other work on computing probabilistic entailment and compare our framework for anytime deduction with work on resource-bounded computation and anytime algorithms.

2 Probabilistic Logic

The language of Nilsson's probabilistic logic is that of ordinary first-order logic. However, in probabilistic logic, the semantic value of a sentence is a probability. When the semantic value of every sentence is either 0 (False) or 1 (True), entailment in probabilistic logic is equivalent to ordinary logical entailment, although the semantics is closer to that of alethic logic than to that of first-order logic.[4]

The meaning of the probability assigned to a sentence is defined in terms of a probability distribution over possible worlds. In propositional probabilistic logic a model is defined to contain a non-empty finite set of possible worlds and a discrete probability distribution over the worlds.[5] The semantic value of a propositional sentence is simply the sum of the probabilities of the worlds that satisfy the sentence.

[3]A rule is quasi-tight if, in a sense to be made clear later, it derives the most informative answer.

[4]Elsewhere [17], we discuss in detail the treatment of probability as a modal operator and the relation between probabilistic logic and alethic logic.

[5]Nilsson also shows how one can restrict attention to a finite set of worlds in the first-order case. This is done by defining a probability distribution over the finite set of equivalence classes of possible worlds

The probabilistic language we work with, $\mathcal{L}_{PL}$, generalizes the propositional version of Nilsson's probabilistic logic in two ways. First, $\mathcal{L}_{PL}$ allows assertions concerning both conditional and unconditional probability, whereas Nilsson's language only admits unconditional probability. Secondly, sentences in $\mathcal{L}_{PL}$ explicitly state an interval within which a conditional probability lies. Nilsson's language does not explicitly state probability values—they are stated in the meta-language—and his paper only considers problems that involve point probabilities.

Sentences of $\mathcal{L}_{PL}$ are of the form $\mathbf{P}(\phi \,|\, \xi) \in I$, where ϕ and ξ are arbitrary sentences of propositional logic, and I names a closed subinterval of the closed unit interval. We view unconditional probability sentences as special cases of conditional sentences. If T is an arbitrary tautology, then $\mathbf{P}(\phi \,|\, T) \in I$ is used to state that the probability of ϕ is in the interval I. For shorthand we sometimes write probabilities conditioned on T simply as unconditional probabilities: $\mathbf{P}(\phi) \in I$. The advantage of this uniform treatment is that we need only have inference rules dealing with conditional probability.

Throughout, this paper uses A, B, C, D, and words starting with a capital letter as the atomic propositions of $\mathcal{L}_{PL}$. The metalinguistic symbols ϕ, ξ, α, β, γ, and δ denote sentences of propositional logic, I and J denote possibly empty, closed probability intervals, ψ denotes a (conditional or unconditional) sentence of $\mathcal{L}_{PL}$, and Ψ denotes a set of sentences of $\mathcal{L}_{PL}$. Furthermore, l, u, x and y denote probability values and an expression of the form $[x\ y]$ denotes the *non-empty* closed interval between x and y.

The interval I is called the *probabilistic component* of $\mathbf{P}(\phi \,|\, \xi) \in I$ and ϕ is called the *logical component* of the unconditional sentence $\mathbf{P}(\phi) \in I$. If ψ is an unconditional sentence, then its logical component is denoted by $\mathsf{LC}(\psi)$, and if Ψ is a set of unconditional sentences then $\mathsf{LC}(\Psi)$ is defined as $\{\mathsf{LC}(\psi) : \psi \in \Psi\}$.

Now let us consider the semantics of $\mathcal{L}_{PL}$. Following Nilsson, a model consists of a non-empty finite set of possible worlds and a probability distribution over the set. The denotation of the expression $\mathbf{P}(\phi)$ in model M, written $[\![\mathbf{P}(\phi)]\!]^M$, is simply the sum of the probabilities of the worlds in M that satisfy ϕ. Now consider the truth conditions for an arbitrary sentence, $\mathbf{P}(\phi \,|\, \xi) \in I$. The traditional definition of conditional probability leaves the conditional probability undefined when the probability of the conditioning formula is zero. Such a treatment in the present framework would require the use of a three-valued logic with its inherent complications. To avoid this we opt to define $\mathbf{P}(\phi \,|\, \xi) \in I$ to be satisfied by models that assign zero to $\mathbf{P}(\xi)$.[6] So the semantic definition for conditional probability sentences is:

$$[\![\mathbf{P}(\phi \,|\, \xi) \in I]\!]^M = \begin{cases} \text{True} & \text{if } [\![\mathbf{P}(\xi)]\!]^M = 0 \\ \text{True} & \text{if } [\![\mathbf{P}(\xi)]\!]^M > 0 \text{ and } \frac{[\![\mathbf{P}(\phi \wedge \xi)]\!]^M}{[\![\mathbf{P}(\xi)]\!]^M} \in I \\ \text{False} & \text{otherwise} \end{cases}$$

As a consequence of this definition, two sentences of the form $\mathbf{P}(\phi \,|\, \xi) \in I$ and $\mathbf{P}(\phi \,|\, \xi) \in J$ may be jointly satisfiable even if I and J are disjoint; both sentences are satisfied by any

representing consistent sets of truth assignments to a finite set of first-order sentences.

[6]Alternatively, one could define the semantic value of a conditional probability sentence to be false in models where the probability of the conditioning formula is zero. But this would lead to problems with tight entailment as we point out in Section 2.1.

model that assigns zero to $\mathbf{P}(\xi)$. Also notice that $\mathbf{P}(\phi \,|\, \xi) \in \emptyset$ may be satisfiable; it is satisfied by precisely those models that assign zero to $\mathbf{P}(\xi)$. Thus, $\mathbf{P}(\phi \,|\, \xi) \in \emptyset$ and $\mathbf{P}(\xi) \in [0\ 0]$ are logically equivalent.

Rather than defining the unconditional probabilities in terms of conditionals, we could equivalently define the semantic values of unconditional sentences as:

$$[\![\mathbf{P}(\phi) \in I]\!]^M = \begin{cases} \text{True} & \text{if } [\![\mathbf{P}(\phi)]\!]^M \in I \\ \text{False} & \text{otherwise} \end{cases}$$

$\mathcal{L}_{PL}$ is two-valued logic that explicitly talks about probabilities. Though an expression such as $\mathbf{P}(\phi)$ can be thought of as having a probability as a semantic value, the sentences of $\mathcal{L}_{PL}$ only take on True and False as semantic values. Consequently, notions such as validity, consistency, entailment, soundness, and completeness are defined as in classical two-valued logic.

Unlike classical logic, $\mathcal{L}_{PL}$ is not compact: an infinite set Ψ may entail a sentence even though no finite subset of Ψ entails the sentence. For example, $\mathbf{P}(A) \in [.5\ .5]$ is entailed by $\{\mathbf{P}(A) \in [.5 - 1/n\ .5 + 1/n] : n \geq 2\}$, but not by any of its finite subsets. Thus there is no sound proof system in $\mathcal{L}_{PL}$ that generates a finite proof of every logical consequence of an arbitrary infinite set of sentences.

2.1 Tight Entailment

Given a set Ψ of $\mathcal{L}_{PL}$ sentences that we take to be true, we may want to compute the possible values of a specified conditional probability, $\mathbf{P}(\phi \,|\, \xi)$. We characterize such a problem by the pair $\langle \Psi, (\phi \,|\, \xi) \rangle$, and, following Nilsson, we call it a *probabilistic entailment problem.* If we want to know the possible values of an unconditional probability, $\mathbf{P}(\phi)$, we have the problem $\langle \Psi, (\phi \,|\, T) \rangle$, and we simply write $\langle \Psi, \phi \rangle$. In either case, the second element of the pair is called the *target* of the probabilistic entailment problem.

To precisely formulate the probabilistic entailment problem, momentarily suppose that for every set $S \subseteq [0\ 1]$ we can write sentences of the form $\mathbf{P}(\phi \,|\, \xi) \in S$ with the obvious truth conditions. Given $\langle \Psi, (\phi \,|\, \xi) \rangle$, ($\Psi$ is as before, a set of $\mathcal{L}_{PL}$ sentences whose probabilistic components are closed intervals) the probabilistic entailment problem is that of finding the smallest set S such that $\mathbf{P}(\phi \,|\, \xi) \in S$ is entailed by Ψ. We call S the *tight entailment of* $(\phi \,|\, \xi)$ *from* Ψ. Observe that if Ψ is unsatisfiable or if Ψ contains the sentence $\mathbf{P}(\xi) \in [0\ 0]$, then the tight entailment of $(\phi \,|\, \xi)$ from Ψ is the empty set.

The following theorem guarantees that a tight entailment in $\mathcal{L}_{PL}$ is always a closed interval and thus we refer to it as the *tightest entailed interval.*

Theorem 1 *For any set of $\mathcal{L}_{PL}$ sentences Ψ and any two propositional sentences ϕ and ξ the tight entailment of $(\phi \,|\, \xi)$ from Ψ is a closed interval.*[7]

Proof: Every sentence in Ψ is of the form $\mathbf{P}(\alpha \,|\, \beta) \in I$. Since $\mathbf{P}(\alpha \,|\, \beta) \in \emptyset$ is logically equivalent to $\mathbf{P}(\beta) \in [0\ 0]$, we shall assume, without loss of generality, that Ψ does not contain any sentences whose probabilistic component is the empty set. We use

[7] This result carries over to the first-order case as well.

a generalization of Nilsson's approach for converting a probabilistic entailment problem to an optimization problem. Consider the set of all logically possible truth assignments to $\{\gamma : \text{either } \mathbf{P}(\gamma \mid \delta) \in I \text{ or } \mathbf{P}(\delta \mid \gamma) \in I \text{ is in } \Psi\} \cup \{\phi, \xi\}$, which is the set of all propositional sentences that occur in this probabilistic entailment problem. Associate a variable with each assignment, and for every propositional formula γ, let W_γ be the expression which is the sum of the variables corresponding to the assignments satisfying ψ. Since $\mathbf{P}(\alpha \mid \beta) \in [l\ u]$ is satisfied by a model M if and only if

$$l \cdot [\![\mathbf{P}(\beta)]\!]^M \leq [\![\mathbf{P}(\alpha \wedge \beta)]\!]^M \leq u \cdot [\![\mathbf{P}(\beta)]\!]^M,$$

this sentence is converted into two linear constraints:

$$l \cdot W_\beta \leq W_{\alpha \wedge \beta}, \text{ and}$$

$$W_{\alpha \wedge \beta} \leq u \cdot W_\beta.$$

All other constraints that result from the conversion process—such as the constraint that the probabilities of the worlds must sum to one—are also linear and define closed regions. Therefore the feasible region for this optimization problem is a connected, closed region. The target conditional probability, $\mathbf{P}(\phi \mid \xi)$, gives rise to the objective function $W_{\phi \wedge \xi}/W_\xi$.

If for some point in the feasible region $W_\xi = 0$, then by the semantic definition of conditional probability the tight entailment is $\emptyset$, a closed interval. Otherwise (and this includes the case in which the feasible region is empty), $W_{\phi \wedge \xi}/W_\xi$ is continuous over the feasible region. Since the feasible region is connected, it follows from the Intermediate Value Theorem that the range of values that $W_{\phi \wedge \xi}/W_\xi$ can take on over the feasible region is an interval. Furthermore, this interval is closed since the feasible region is closed. □

This theorem states an interesting closure property for the logic: given closed probability intervals on sentences, the tight entailment is a closed interval. This property does not hold for point probabilities; though the sentences in Ψ may all have point probabilities, the tight entailment of $(\phi \mid \xi)$ may not be a point. Grosof [14] also observed this closure property in relation to his Type-1-*ui* theories. This property is important for the formulation of inference rules since the rules must be applicable to the sentences they derive in order to allow multi-step derivations. Notice that if we had defined conditional probability sentences to be false in models in which the probability of the conditioning proposition is zero, we would not have this closure property: The tight entailment of α from $\mathbf{P}(\beta \mid \alpha) \in [0\ 1]$ would be the open interval $(0\ 1]$ since any model M in which $[\![\mathbf{P}(\alpha)]\!]^M = 0$ would not satisfy $\mathbf{P}(\beta \mid \alpha) \in [0\ 1]$.

A *proof system* is a set of axioms and inference rules. In addition to the usual notions of soundness and completeness, we define a weaker notion of completeness specific to tight entailment. A proof system is *complete for tight entailment* if and only if for every probabilistic entailment problem, $\langle \Psi, (\phi \mid \xi) \rangle$, there is some subset, I, of the tightest entailed interval such that $\mathbf{P}(\phi \mid \xi) \in I$ is derivable from Ψ. This definition is similar to Grosof's [14] definition of completeness for probabilistic inference. If a proof system is complete then it is complete for tight entailment since the tightest entailed interval is just one of the entailed intervals. Any proof system that is complete for tight entailment can be made complete simply by adding the *interval expansion inference rule*: from $\mathbf{P}(\alpha \mid \beta) \in I$ derive $\mathbf{P}(\alpha \mid \beta) \in J$ provided that I is a subset of J. In practice, we are more interested in computing tight entailment than in computing general entailment.

Nilsson proposes a method for computing tight entailment for problems $\langle \Psi, \phi \rangle$ involving only unconditional probabilities and in which the probability intervals in Ψ are point values. The method converts the probabilistic entailment problem to a linear programming problem in which the objective is to minimize and then to maximize the probability of ϕ. The conversion requires finding every possible assignment of truth and falsity that a world can give to $\mathsf{LC}(\Psi) \cup \{\phi\}$. These assignments are then used to set up a system of linear equations that expresses the constraints among the probability distribution over these assignments and the probabilities of $\mathsf{LC}(\Psi)$. Paass [28] generalizes this method to allow Ψ to contain conditional probability sentences with arbitrary probability intervals. The target sentence must still be an unconditional probability. This more general probabilistic entailment problem can still be converted to a linear programming problem.

In propositional probabilistic logic, this method provides a decision procedure for finding the tight entailment of ϕ from Ψ. However, for a first-order probabilistic logic this method does not provide even a semi-decision procedure. The difficulty is that the possible truth assignments to $\mathsf{LC}(\Psi) \cup \{\phi\}$ may not be enumerable. Therefore, Nilsson's method may not even proceed as far as setting up the system of linear equations. In this case, the method yields no information about the probability of ϕ.

Nilsson's procedure is all or none; it either computes the tight entailment of ϕ from Ψ or it yields no information about it. As such, it fails to exploit the capacity of probabilistic logic to express intermediate results. Before the ultimate value, or set of values, of ϕ is computed, it is possible to have an intermediate result stating that some truth values have been eliminated. In general, when using a multi-valued logic one may be able to rule out the possibility that a sentence has certain truth values before determining a single truth value or the smallest set of possible truth values. Probabilistic logic has this capacity because embedded within every two-valued expression $P(\alpha) \in I$ is a multi-valued expression, $P(\alpha)$. Notice that a classical two-valued logic does not have this capacity; once one truth value is eliminated, the value of the target sentence is fully determined.

3 Anytime Deduction

So the question arises as to whether one can formulate an inference procedure that provides partial information about a probabilistic entailment problem before a complete answer is computed and that provides increasingly informative answers as a computation progresses. We call such a procedure an *anytime deduction procedure*.[8]

A procedure for deducing the truth value of a sentence in a multi-valued logic (i.e. more than two truth values) is an anytime deduction procedure if it has two properties:

Partiality: The informativeness of derived partial answers increases monotonically for all executions, and for some executions there is a point during the execution at which the procedure has partial information strictly between no information and total information.

Correctness: All derived partial answers are correct.

[8]This is the same concept as the convergent deduction procedure that we previously introduced [16].

For probabilistic logic a derived partial answer is the interval of the target sentence that has been derived, and one derived partial answer is more informed than any derived partial answer of which it is a subset. So we interpret monotonically increasing informativeness as meaning that derived probability intervals only ever change to subsets of their values.

Our method for anytime deduction for probabilistic logic is based on a set of sound inference rules. Though the rules are not presented and discussed until the next section, the following example illustrates their use.

Example 1 Consider the sentences

$$\mathbf{P}(B \rightarrow A) \in [1\ 1] \tag{1}$$

$$\mathbf{P}(A \rightarrow C) \in [1\ 1] \tag{2}$$

$$\mathbf{P}(B) \in [.2\ .2] \tag{3}$$

$$\mathbf{P}(C) \in [.6\ .6] \tag{4}$$

Two separate proofs for $\mathbf{P}(A)$ may be constructed. Sentences (1) and (3) entail $\mathbf{P}(A) \in [.2\ 1]$ and sentences (2) and (4) entail $\mathbf{P}(A) \in [0\ .6]$. Since both of these intervals are derived by sound inference, the probability of A must lie within *both* intervals. Accordingly, we intersect the two intervals to yield $\mathbf{P}(A) \in [.2\ .6]$. The interval $[.2\ .6]$ is, in fact, the tight entailment of A from (1)–(4) and is the interval that Nilsson's method would obtain. The rule for intersecting intervals, called the *multiple derivation rule*, is the key operation that gives our deduction procedure its anytime character.

Given a set of sound inference rules for $\mathcal{L}_{PL}$ it is simple to construct an anytime deduction procedure for computing the tight entailment of $(\phi \,|\, \xi)$ from Ψ. Throughout its execution the procedure maintains an estimate of the tightest entailed interval, referred to as the *current derived interval*. The procedure begins with $[0\ 1]$ as its current derived interval and proceeds by enumerating all possible proofs from Ψ that can be constructed with the given inference rules. If the procedure has a current derived interval of I_1 and it generates a proof of $\mathbf{P}(\phi \,|\, \xi) \in I_2$, the current derived interval is updated to $I_1 \cap I_2$ using the multiple derivation rule. A procedure that operates in this manner can, at any time, be asked to report its current derived interval and thus can provide information about the tightest entailed interval without computing exactly what that interval is. The current derived interval decreases monotonically throughout an execution and it is always correct, that is, it always contains the tightest entailed interval.

Consider Example 1 as an illustration of how such a procedure might operate. The procedure starts by setting the current derived interval for A to $[0\ 1]$. Using its inference rules, the procedure might then derive $\mathbf{P}(A) \in [.2\ 1]$ from (1) and (3). Intersecting $[.2\ 1]$ with the current derived interval yields $[.2\ 1]$, which is then taken to be the current derived interval. The procedure then might derive $\mathbf{P}(A) \in [0\ .6]$ from (2) and (4). A new current derived interval of $[.2\ .6]$ is obtained from the old by intersecting it with $[0\ .6]$. This is the tightest entailed interval, so no further computation with sound inference rules could modify it.

An anytime deduction procedure is called *convergent* if during all executions the current derived interval converges in the limit. That is, there is some interval I (which is not

(i)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \quad \mathbf{P}(\alpha \vee \beta \mid \delta) \in [u\ v] \quad \mathbf{P}(\alpha \wedge \beta \mid \delta) \in [w\ z]}{\mathbf{P}(\beta \mid \delta) \in [\max(w, u-y+w)\ \ \min(v, v-x+z))]}$$

provided $w \leq y,\ x \leq v,\ w \leq v$

(ii)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \quad \mathbf{P}(\beta \mid \delta) \in [u\ v] \quad \mathbf{P}(\alpha \wedge \beta \mid \delta) \in [w\ z]}{\mathbf{P}(\alpha \vee \beta \mid \delta) \in [\max(x, u, w, x+u-z)\ \ \lfloor y+v-w \rfloor_1]}$$

provided $w \leq y,\ w \leq v,\ z \geq x+u-1$

(iii)

$$\frac{\mathbf{P}(\beta \mid \delta) \in [x\ y] \quad \mathbf{P}(\alpha \mid \delta) \in [u\ v] \quad \mathbf{P}(\alpha \vee \beta \mid \delta) \in [w\ z]}{\mathbf{P}(\alpha \wedge \beta \mid \delta) \in [\lceil x+u-z \rceil^0\ \ \min(y, v, z, y+v-w)]}$$

provided $z \geq x,\ z \geq u$

(iv)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y]}{\mathbf{P}(\neg\alpha \mid \delta) \in [1-y\ 1-x]}$$

Figure 1: Inference rules with a fixed conditioning sentence.

necessarily the tightest entailed interval) such that for each interval $J \supset I$ there is some point in time after which the current derived interval is always a subset of J. The procedure is *finitely convergent* if there is some point in time after which the current derived interval never changes.[9] A deduction *procedure* that always finitely converges to the tightest entailed interval can be obtained by augmenting a sound and complete proof *system* with a method of enumerating all proofs. Any decision procedure for probabilistic entailment must be finitely convergent. However, a finitely convergent procedure is not necessarily a decision procedure because the procedure may not recognize when it has converged.

4 The Inference Rules

This section examines the set of inference rules for $\mathcal{L}_{PL}$ shown in Figures 1, 2, 3, and 4.

We do not claim this set of inference rules to be complete, although we have tried to be comprehensive in covering the rules contained in the literature. In these rules the metalinguistic variables α, β, γ, and δ are taken to represent arbitrary propositional formulas. Recall that T is used to denote an arbitrary tautology. An expression of the form $\lceil x \rceil^0$ denotes $\max(0, x)$ and $\lfloor x \rfloor_1$ denotes $\min(1, x)$. To simplify the presentation of the rules, some conventions are assumed. In those cases where computating a bound of the probabilistic component of the conclusion requires division by zero, that bound shall be taken to be zero if it is a lower bound and one if it is an upper bound. In those cases where an inference rule derives a sentence whose probabilistic component is an expression $[x\ y]$ where $x > y$, such

[9] Observe that an execution converges to $\emptyset$ only if it finitely converges to $\emptyset$.

(v)

$$\frac{\begin{array}{l}\mathbf{P}(\alpha \wedge \beta \,|\, \delta) \in [x\ y] \\ \mathbf{P}(\beta \,|\, \delta) \in [u\ v]\end{array}}{\mathbf{P}(\alpha \,|\, \beta \wedge \delta) \in [x/v\ z]}$$

where $z = \begin{cases} 1 \text{ if } y > u \\ 0 \text{ if } y = u = 0 \\ y/u \text{ otherwise} \end{cases}$

provided $x \leq v$, $v > 0$.

(vi)

$$\frac{\begin{array}{l}\mathbf{P}(\beta \,|\, \alpha \wedge \delta) \in [x\ y] \\ \mathbf{P}(\alpha \wedge \beta \,|\, \delta) \in [u\ v]\end{array}}{\mathbf{P}(\alpha \,|\, \delta) \in [u/y\ \lfloor v/x \rfloor_1]}$$

provided $y \geq u,\ x > 0,\ y > 0$

(vii)

$$\frac{\begin{array}{l}\mathbf{P}(\beta \,|\, \delta) \in [x\ y] \\ \mathbf{P}(\alpha \,|\, \beta \wedge \delta) \in [u\ v]\end{array}}{\mathbf{P}(\alpha \wedge \beta \,|\, \delta) \in [x \cdot u\ y \cdot v]}$$

(viii)

$$\frac{\begin{array}{l}\mathbf{P}(\alpha_k \,|\, \alpha_1 \wedge \delta) \in [w\ z] \\ \mathbf{P}(\alpha_i \,|\, \alpha_{i+1} \wedge \delta) \in [x_i\ y_i],\ 1 \leq i \leq k-1 \\ \mathbf{P}(\alpha_{i+1} \,|\, \alpha_i \wedge \delta) \in [u_i\ v_i],\ 1 \leq i \leq k-1\end{array}}{\mathbf{P}(\alpha_1 | \alpha_k \wedge \delta) \in [(w \cdot \prod_{i=1}^{k-1} \frac{x_i}{v_i})\ (z \cdot \prod_{i=1}^{k-1} \frac{y_i}{u_i})]}$$

(ix)

$$\frac{\begin{array}{l}\mathbf{P}(\alpha \,|\, \beta \wedge \delta) \in [x_1\ y_1] \\ \mathbf{P}(\beta \,|\, \alpha \wedge \delta) \in [x_2\ y_2] \\ \mathbf{P}(\gamma \,|\, \beta \wedge \delta) \in [u_1\ v_1] \\ \mathbf{P}(\beta \,|\, \gamma \wedge \delta) \in [u_2\ v_2]\end{array}}{\mathbf{P}(\gamma \,|\, \alpha \wedge \delta) \in [x_2 \cdot \lceil \frac{1-u_1}{x_1} \rceil^0\ \ \min(1,\ 1 - x_2 + \frac{x_2 \cdot v_1}{x_1}, z)]}$$

where $z = \begin{cases} 1 \text{ if } u_2 = 0 \\ \frac{y_2}{x_1} \cdot (\frac{v_1}{u_2} + \min(0, x_1 - v_1)) \text{ otherwise} \end{cases}$

Figure 2: Inference rules with non-fixed conditioning sentence.

(x)

$$\frac{\mathbf{P}(\beta \,|\, \delta) \in [0\ 0]}{\mathbf{P}(\alpha \,|\, \beta \wedge \delta) \in \emptyset}$$

(xi)

$$\frac{\mathbf{P}(\alpha \,|\, \beta) \in \emptyset}{\mathbf{P}(\beta \,|\, T) \in [0\ 0]}$$

(xii)

$$\frac{\mathbf{P}(\alpha \,|\, \beta) \in [x\ y]}{\mathbf{P}(\alpha \,|\, \delta) \in [x\ y]}$$
provided β logically equivalent to δ

(xiii)

$$\frac{}{\mathbf{P}(\alpha \,|\, \delta) \in [1\ 1]}$$
provided δ entails α

(xiv)

$$\frac{\mathbf{P}(\alpha \,|\, \delta) \in [x\ y]}{\mathbf{P}(\beta \,|\, \delta) \in [x\ 1]}$$
provided α entails β

(xv)

$$\frac{\mathbf{P}(\beta \,|\, \delta) \in [x\ y]}{\mathbf{P}(\alpha \,|\, \delta) \in [0\ y]}$$
provided α entails β

(xvi)

$$\frac{}{\mathbf{P}(\alpha \,|\, \delta) \in [0\ 1]}$$

(xvii)

$$\frac{\mathbf{P}(\alpha \,|\, \delta) \in [x\ y] \quad \mathbf{P}(\alpha \,|\, \delta) \in [u\ v]}{\mathbf{P}(\alpha \,|\, \delta) \in [\max(x,u)\ \min(y,v)]}$$

(xviii)

$$\frac{\mathbf{P}(\alpha \,|\, T) \in \emptyset}{\mathbf{P}(\beta \,|\, \gamma) \in \emptyset}$$

(xix)

$$\frac{\mathbf{P}(\alpha \,|\, \delta) \in I}{\mathbf{P}(\alpha \,|\, \delta) \in J}$$
provided $I \subset J$

Figure 3: Inference rules not based on logical connectives.

(xx)

$$\frac{\mathbf{P}(\beta \mid \delta) \in [x\ y] \qquad \mathbf{P}(\beta \rightarrow \alpha \mid \delta) \in [u\ v]}{\mathbf{P}(\alpha \mid \delta) \in [\lceil x+u-1\rceil^0\ v]}$$

provided $y \geq 1-v$

(xxi)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \qquad \mathbf{P}(\beta \rightarrow \alpha \mid \delta) \in [u\ v]}{\mathbf{P}(\beta \mid \delta) \in [1-v\ \lfloor y+1-u\rfloor_1]}$$

provided $x \leq v$

(xxii)

$$\frac{\mathbf{P}(\beta \wedge \gamma \mid \delta) \in [x\ y] \qquad \mathbf{P}(\beta \rightarrow \alpha \mid \delta) \in [u\ v]}{\mathbf{P}(\alpha \wedge \beta \wedge \gamma \mid \delta) \in [\lceil x+u-1\rceil^0\ \min(y,v)]}$$

(xxiii)

$$\frac{\mathbf{P}(\alpha \rightarrow \beta \mid \delta) \in [x\ y] \qquad \mathbf{P}(\alpha \rightarrow \gamma \mid \delta) \in [u\ v]}{\mathbf{P}(\alpha \rightarrow \beta \wedge \gamma \mid \delta) \in [\lceil x+u-1\rceil^0\ \min(y,v)]}$$

(xxiv)

$$\frac{\mathbf{P}(\beta \rightarrow \alpha \mid \delta) \in [x\ y] \qquad \mathbf{P}(\gamma \rightarrow \alpha \mid \delta) \in [u\ v]}{\mathbf{P}(\beta \wedge \gamma \rightarrow \alpha \mid \delta) \in [\max(x,u)\ \lfloor y+v\rfloor_1]}$$

(xxv)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \qquad \mathbf{P}(\beta \mid \delta) \in [u\ v]}{\mathbf{P}(\alpha \wedge \beta \mid \delta) \in [\lceil x+u-1\rceil^0\ \min(y,v)]}$$

(xxvi)

$$\frac{\mathbf{P}(\alpha \wedge \beta \mid \delta) \in [x\ y]}{\mathbf{P}(\alpha \mid \delta) \in [x\ 1]}$$

(xxvii)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \qquad \mathbf{P}(\beta \mid \delta) \in [u\ v]}{\mathbf{P}(\alpha \vee \beta \mid \delta) \in [\max(x,u)\ \lfloor y+v\rfloor_1]}$$

(xxviii)

$$\frac{\mathbf{P}(\alpha \vee \beta \mid \delta) \in [x\ y]}{\mathbf{P}(\alpha \mid \delta) \in [0\ y]}$$

(xxix)

$$\frac{\mathbf{P}(\alpha \wedge \beta \mid \delta) \in [x\ y] \qquad \mathbf{P}(\alpha \wedge \neg\beta \mid \delta) \in [u\ v]}{\mathbf{P}(\alpha \mid \delta) \in [x+u\ \lfloor y+v\rfloor_1]}$$

provided $v \leq 1-x,\ y \leq 1-u$

(xxx)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y]}{\mathbf{P}(\beta \mid \delta) \in [x\ y]}$$

provided β logically equivalent to α

(xxxi)

$$\frac{}{\mathbf{P}(\alpha \mid \delta) \in [0\ 0]}$$

provided α is unsatisfiable

(xxxii)

$$\frac{}{\mathbf{P}(T \mid \delta) \in [1\ 1]}$$

Figure 4: Derived inference rules.

an expression should be read as the empty set. The conclusion of an inference rule is the only place where such an expression occurs in this paper. We continue to insist that every other occurrence of $[x\ y]$ in this paper, including occurrences in the premises of the inference rules, denotes a non-empty interval and hence that $x \leq y$.

The rules in Figure 1 are essentially rules for unconditional probability sentences that have been adapted to the conditional setting by considering a fixed conditioning sentence. That is, the same conditioning sentence is used in each premise and in the conclusion. All of these rules reduce to non-conditional probability rules by taking the conditioning sentences to be T. Rules (i)–(iii) are all derived from the relation $\mathbf{P}(\alpha \vee \beta) = \mathbf{P}(\alpha) + \mathbf{P}(\beta) - \mathbf{P}(\alpha \wedge \beta)$. The provided conditions ensure that the probabilities of the premises are consistent. They are obtained from two relations. First, the probability of a conjunction cannot be greater than the probability of either of the conjuncts or of the disjunction of the two conjuncts. Second, the probability of a disjunction cannot be less than the probability of either of the disjuncts. The expressions for the derived bounds also take into account these two relations. For example, in the lower bound of rule (i), the expression $u - y + w$ comes from the above equation for $\mathbf{P}(\alpha \vee \beta)$ but if u and y are weak bounds then a tighter bound might be obtained from the fact that $\mathbf{P}(\alpha \wedge \beta) \leq \mathbf{P}(\beta)$. Hence we have $\max(w,\ u - y + w)$.

The rules of Figure 2 operate on sentences in which the conditioning sentence is not fixed. These rules do not reduce to non-conditional probability rules.

Unlike the rules in Figures 1 and 2, the rules in Figure 3 mention no logical connectives. These rules are based on semantical relations between the propositional components of the sentences, the definition of conditioning on a sentence of probability zero, and so on. Rules (x) and (xi) are based on the logical equivalence of $\mathbf{P}(\alpha \mid \beta) \in \emptyset$ and $\mathbf{P}(\beta) \in [0\ 0]$. Rules (xiv) and (xv) are based on the property that if α entails β then the probability of β is at least as great as the probability of α. Rule (xvii) is the *multiple derivation rule* which gives our proof system its anytime character. Rule (xviii) allows any sentence to be derived from an inconsistent set of sentences. It implements the semantic property that no model satisfies both the sentence $\mathbf{P}(\alpha \mid \delta) \in \emptyset$ and a sentence saying that the probability of δ is not zero. Rule (xix) allows us to expand probability intervals. In particular, from $\mathbf{P}(\alpha \mid \gamma) \in \emptyset$ we can derive $\mathbf{P}(\alpha \mid \gamma) \in I$ for any I.

The rules in Figure 4 are all derivable from the previous rules in the sense that any inference performed by the derived rule could be performed by a sequence of previous rules. For example rule (xxx) is derivable from rules (xiv), (xv), and (xvii). Rule (xx) is derivable from rules (iv), (i), (xvi), and (xxx) as follows.

$$
\begin{array}{rll}
\mathbf{P}(B \mid D) \in [x\ y] & \text{given} & (5) \\
\mathbf{P}(B \rightarrow A \mid D) \in [u\ v] & \text{given} & (6) \\
\mathbf{P}(\neg B \mid D) \in [1-y\ 1-x] & \text{rule (iv) applied to (5)} & (7) \\
\mathbf{P}(\neg B \vee A \mid D) \in [u\ v] & \text{rule (xxx) applied to (6)} & (8) \\
\mathbf{P}(A \wedge B \mid D) \in [0\ 1] & \text{rule (xvi)} & (9) \\
\mathbf{P}(A \mid D) \in [\lceil x+u-1 \rceil^0\ v] \text{ provided } y \geq 1-v & \text{rule (i) applied to (7)–(9)} & (10)
\end{array}
$$

Several of our rules have appeared elsewhere in the literature or are slight generalizations of rules that appear in the literature in the sense that we have added a fixed conditioning proposition δ. Rules (xx) and (xxi) are presented by Garvey, Lowrance, and Fischler [13].

A derivation of rule (xx) can be found in a paper by Dubois [11]. Rule (xxv) for introducing conjunction is also presented by Ursic [34]. Rules (xxiii) and (xxiv) accomplish a transformation similar to Pearl's [29] clustering technique for removing loops in Bayes nets. Rules (viii) and (ix) are respectively the the generalized Bayes' rule [10] and the rule of quantified syllogism, presented by Amarger, et.al. [2].

Our inference rules contain all the rules present in Quinlan's [31] INFERNO system. Since all of INFERNO's rules represent bidirectional inference between a pair of formulas, most of INFERNO's rules correspond to two of our rules. Rules (i) and (ii) corespond to the rule labeled "A disjoins-exclusive $\{s_1, s_2, ..., s_n\}$." Rules (vi) and (vii) taken together generalize the INFERNO rule labeled "A enables S with strength X." Rule (iv) corresponds to the rule labeled "A negates S." Rules (xxv) and (xxvi) correspond to the rule labeled "A conjoins $\{s_1, s_2, ..., s_n\}$." Rules (xxvii) and (xxviii) correspond to the rule labeled "A disjoins $\{s_1, s_2, ..., s_n\}$." The rules corresponding to INFERNO's rules for conditional independence are presented in Section 6. The relationship between our system and INFERNO is discussed further in Section 7.

4.1 Examples

This section presents three examples illustrating the use of the inference rules and their roles in performing probabilistic deduction. The first example involves inference with conditional probabilities and the second two involve inference with unconditional probabilities. In all three examples, we derive the tightest entailed interval of the target sentence.

4.1.1 Example 2

Consider the following set of sentences concerning the likelihood of a person having the flu, a cold, a fever, and having a fever given each of the two other conditions.

$$\mathbf{P}(Flu) \in [.01\ .01] \tag{11}$$

$$\mathbf{P}(Cold) \in [.05\ .05] \tag{12}$$

$$\mathbf{P}(Fever) \in [.05\ .1] \tag{13}$$

$$\mathbf{P}(Fever \,|\, Flu) \in [.8\ .9] \tag{14}$$

$$\mathbf{P}(Fever \,|\, Cold) \in [.3\ .4] \tag{15}$$

Suppose we want to know the chance that a person has a cold or the flu given that he has a fever. Our inference rules can be used to derive this conditional probability as follows:

$\mathbf{P}(Fever \wedge Flu) \in [.008\ .009]$	rule (vii) applied to (11) and (14)	(16)
$\mathbf{P}(Flu \,|\, Fever) \in [.08\ .18]$	rule (v) applied to (13) and (16)	(17)
$\mathbf{P}(Fever \wedge Cold) \in [.015\ .02]$	rule (vii) applied to (12) and (15)	(18)
$\mathbf{P}(Cold \,|\, Fever) \in [.15\ .4]$	rule (v) applied to (13) and (18)	(19)
$\mathbf{P}(Cold \vee Flu \,|\, Fever) \in [.15\ .58]$	rule (xxvii) applied to (17) and (19)	(20)

[10] Bayes' rule can be derived from this rule and rule (xxxii) by taking k to be 3, and δ and α_2 to be T.

We now present two examples demonstrating the use of the inference rules dealing with unconditional probability, as well as illustrating the anytime character of the rules. In both examples the rules derive increasingly narrow intervals that finitely converge to the tightest entailed interval of the target sentence.

4.1.2 Example 3

This example illustrates how rules (xxiii) and (xxiv) can be used to derive the probability of a conjunction when a logical dependency exists between the conjuncts. Suppose we wish to derive the tight entailment of $B \wedge C$ from the following set of sentences:

$$\mathbf{P}(A) \in [.6\ 1] \tag{21}$$

$$\mathbf{P}(A \rightarrow B) \in [.8\ .9] \tag{22}$$

$$\mathbf{P}(A \rightarrow C) \in [.9\ 1] \tag{23}$$

$$\mathbf{P}(B \rightarrow D) \in [.5\ .8] \tag{24}$$

$$\mathbf{P}(C \rightarrow D) \in [.8\ .9] \tag{25}$$

$$\mathbf{P}(D) \in [0\ .2] \tag{26}$$

A graphical representation of these sentences is shown in Figure 5. Simply using rule (xx) to propagate along implications (22) and (23) and then combining these with rule (xxv) yields $\mathbf{P}(B \wedge C) \in [0\ .8]$. A tighter bound can be derived by exploiting the fact that B and C both depend on A. The derivation goes as follows:

$\mathbf{P}(A \rightarrow B \wedge C) \in [.7\ .9]$	rule (xxiii) applied to sentences (22) and (23)	(27)
$\mathbf{P}(B \wedge C) \in [.3\ .9]$	rule (xx) applied to (21) and (27)	(28)
$\mathbf{P}(B \wedge C \rightarrow D) \in [.8\ 1]$	rule (xxiv) applied to (24) and (25)	(29)
$\mathbf{P}(B \wedge C) \in [0\ .4]$	rule (xxi) applied to (26) and (29)	(30)
$\mathbf{P}(B \wedge C) \in [.3\ .4]$	rule (xvii) applied to (28) and (30)	

The derived interval is the tightest entailed interval. This derivation could have been stopped after step (28) to yield non-trivial partial information about the tightest entailed interval of $B \wedge C$.

4.1.3 Example 4

This example illustrates the use of rules (xxi), (xxii), (xxv), (xxvi), and (xvii). Consider the problem of deriving the tight entailment of $A \wedge D$ from the following set of sentences:

$$\mathbf{P}(B \rightarrow A) \in [.9\ 1] \tag{31}$$

$$\mathbf{P}(D \rightarrow B) \in [.8\ .9] \tag{32}$$

$$\mathbf{P}(A \rightarrow C) \in [.6\ .8] \tag{33}$$

$$\mathbf{P}(D) \in [.8\ 1] \tag{34}$$

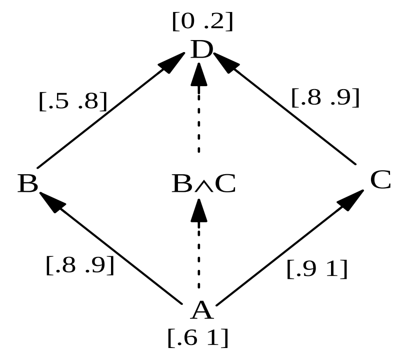

Figure 5: Graphical representation of Example 3.

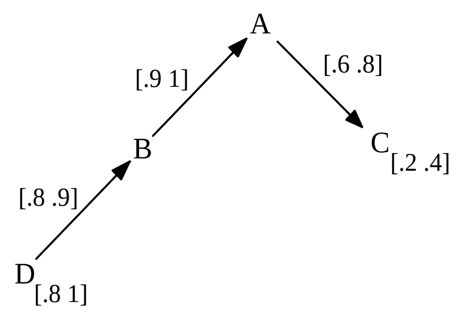

Figure 6: Graphical representation of Example 4.

$$\mathbf{P}(C) \in [.2\ .4] \tag{35}$$

A graphical representation of the sentences is shown in Figure 6. The derivation proceeds as follows:

$\mathbf{P}(A) \in [.2\ .8]$	rule (xxi) applied to sentences (33) and (35)	(36)
$\mathbf{P}(A \wedge D) \in [0\ .8]$	rule (xxv) applied to (36) and (34)	(37)
$\mathbf{P}(B \wedge D) \in [.6\ .9]$	rule (xxii) applied to (32) and (34)	(38)
$\mathbf{P}(A \wedge B \wedge D) \in [.5\ .9]$	rule (xxii) applied to (38) and (31)	(39)
$\mathbf{P}(A \wedge D) \in [.5\ 1]$	rule (xxvi) applied to (39)	(40)
$\mathbf{P}(A \wedge D) \in [.5\ .8]$	rule (xvii) applied to (37) and (40)	

Again in this derivation the derived interval is the tightest entailed interval, and non-trivial partial information about the tightest entailed interval is available already after the second step of the derivation.

4.2 Properties of the Inference Rules

We are interested in proving two properties of our inference rules: soundness and quasi-tightness. We define quasi-tightness as follows. Consider an arbitrary inference rule

$$\frac{\begin{array}{l}\mathbf{P}(\alpha_1 \,|\, \beta_1) \in I_1 \\ \vdots \\ \mathbf{P}(\alpha_n \,|\, \beta_n) \in I_n\end{array}}{\mathbf{P}(\alpha \,|\, \beta) \in I}$$

where $n \geq 0$. Partition the set of instances of the inference rule such that two instances are in the same class if, and only if, each instantiates $I, I_1, \ldots, I_n$ identically. Each class C typically contains many different propositional instances of the rule. In any class C, each rule instance $r \in C$ derives the same interval, I, even though the probabilistic entailment problem associated with r, $\langle\{\mathbf{P}(\alpha_i \,|\, \beta_i) \in I_i\}, (\alpha \,|\, \beta)\rangle$ with the variables appropriately instantiated, might have a different tightest entailed interval, E_r. Let E_C be the smallest interval that contains E_r for every $r \in C$. A rule is *quasi-tight* if and only if $I \subseteq E_C$ for every class C.

Let us illustrate this definition by observing that rule (xix) is not quasi-tight. According to the definition, one of the classes—call it C—of this inference rule consists of the set of instances of the form

$$\frac{\mathbf{P}(\alpha) \in [.5 \; .6]}{\mathbf{P}(\alpha) \in [.4 \; .7]}$$

For any way of instantiating α, the tight entailment of α from $\mathbf{P}(\alpha) \in [.5 \; .6]$ is $[.5 \; .6]$, so E_C is $[.5 \; .6]$. Since $[.4 \; .7]$ is not a subset of $[.5 \; .6]$, this rule is not quasi-tight.

The quasi-tightness of an inference rule does not imply that the rule derives the tightest entailed interval for all its instances.[11] For example, rule (xx) is quasi-tight but derives $\mathbf{P}(A) \in [.5 \; 1]$ from $\mathbf{P}(A) \in [.5 \; .6]$ and $\mathbf{P}(A \rightarrow A) \in [1 \; 1]$, even though $[.5 \; 1]$ is clearly not the tightest entailed interval. This loss of information arises because the rule does not account for the equality of α and β in this instance.

Theorem 2 *Inference rules (i)–(xviii) and (xx)–(xxxii), which are shown in Figures 1, 2, 3, and 4 are both sound and quasi-tight, and rule (xix) is sound.*

Proof: As an example, we prove the quasi-tightness and soundness of rule (v). The proofs for the other rules are similar.

To prove the soundness of rule (v), suppose that M is a model that satisfies both $\mathbf{P}(\alpha \wedge \beta \,|\, \delta) \in [x \; y]$ and $\mathbf{P}(\beta \,|\, \delta) \in [u \; v]$, and that $v > 0$. The existence of a model satisfying the premises is guaranteed by the provision $x \leq v$ which ensures that the probabilities of the premises are consistent. If $[\![\mathbf{P}(\delta)]\!]^M = 0$ then M satisfies the conclusion of the rule. Otherwise,

$$\frac{[\![\mathbf{P}(\alpha \wedge \beta \wedge \delta)]\!]^M}{[\![\mathbf{P}(\delta)]\!]^M} \geq x \quad \text{and} \quad \frac{[\![\mathbf{P}(\beta \wedge \delta)]\!]^M}{[\![\mathbf{P}(\delta)]\!]^M} \leq v,$$

[11] For this reason, we eschew the use of the possibly misleading term "local completeness" [2] for the quasi-tightness property.

so

$$\frac{[\![\mathbf{P}(\alpha \wedge \beta \wedge \delta)]\!]^M}{[\![\mathbf{P}(\beta \wedge \delta)]\!]^M} \geq \frac{x}{v}.$$

Then by the semantic definition of conditional probability, a lower bound on $\mathbf{P}(\alpha \,|\, \beta \wedge \delta)$ is x/v. For the upper bound, we must consider the three cases for z. In the first case, $y > u$, the derived upper bound of 1 is trivially sound. In the second case, $y = u = 0$, we have also that $x = 0$. Consider the models in which the probability of δ is nonzero. In these models $\alpha \wedge \beta$ has probability zero. We have two classes of such models. In one class, M_1, β has nonzero probability, so $[\![\mathbf{P}(\alpha \,|\, \beta \wedge \delta) \in [0\ 0]]\!]^{M_1} = \text{True}$. In the other class, M_2, β has probability zero, so $\mathbf{P}(\alpha \,|\, \beta \wedge \delta)$ can be in any interval, including [0 0]. So in the case where $y = u = 0$ the tightest entailed interval is [0 0]. In the last case we have

$$\frac{[\![\mathbf{P}(\alpha \wedge \beta \wedge \delta)]\!]^M}{[\![\mathbf{P}(\delta)]\!]^M} \leq y \quad \text{and} \quad \frac{[\![\mathbf{P}(\beta \wedge \delta)]\!]^M}{[\![\mathbf{P}(\delta)]\!]^M} \geq u,$$

so

$$\frac{[\![\mathbf{P}(\alpha \wedge \beta \wedge \delta)]\!]^M}{[\![\mathbf{P}(\beta \wedge \delta)]\!]^M} \leq \frac{y}{u}.$$

Hence an upper bound on $\mathbf{P}(\alpha \,|\, \beta \wedge \delta)$ is 1 if $y > u$, 0 if $y = u = 0$, and y/u otherwise.

To prove the quasi-tightness of rule (v), consider an arbitrary equivalence class of all instances that instantiate x, y, u, v identically and where $v > 0$ and $x \leq v$. We must show that the computed lower bound is greater than or equal to the lower bound of the tightest entailed interval of some instance in the class, and that the computed upper bound is less than or equal to the upper bound of the tightest entailed interval of some instance in the class. This is accomplished by showing that for the single instance in which α, β and δ are instantiated with A, B and T, respectively, there exist models, M_1, M_2, M_3, and M_4 of the premises such that

$[\![\mathbf{P}(A \,|\, B \wedge T) \in [x/v\ x/v]]\!]^{M_1} = \text{True}$,
$[\![\mathbf{P}(A \,|\, B \wedge T) \in [y/u\ y/u]]\!]^{M_2} = \text{True}$,
$[\![\mathbf{P}(A \,|\, B \wedge T) \in [1\ 1]]\!]^{M_3} = \text{True}$, and
$[\![\mathbf{P}(A \,|\, B \wedge T) \in [0\ 0]]\!]^{M_4} = \text{True}$.

Model M_1 shows the tightness of the lower bound and models M_2, M_3, and M_4 show the tightness of the upper bound, covering the cases in which $y \leq u$, $y > u$, and $y = u = 0$, respectively. The existence of a model satisfying the premises is guaranteed by the provision on the rule which ensures that the probabilities of the premises are consistent. In the soundness proof we showed the existence of model M_4. We now construct the remaining models.

First we construct M_1. Let $[\![\mathbf{P}(B)]\!]^{M_1} = v$, $[\![\mathbf{P}(A)]\!]^{M_1} = x/v$, and A and B be conditionally independent in M. Then M satisfies the rule premises and

$$\frac{[\![\mathbf{P}(A \wedge B \wedge T)]\!]^{M_1}}{[\![\mathbf{P}(B \wedge T)]\!]^{M_1}} = \frac{x}{v}.$$

Next we construct M_2. Let $[\![\mathbf{P}(B)]\!]^{M_2} = u$, $[\![\mathbf{P}(A)]\!]^{M_2} = y/u$, and A and B be conditionally independent in M_2. Then M_2 satisfies the premises and

$$\frac{[\![\mathbf{P}(A \wedge B \wedge T)]\!]^{M_2}}{[\![\mathbf{P}(B \wedge T)]\!]^{M_2}} = \frac{y}{u}.$$

Finally we construct M_3. Since $y > u$ and $x \leq v$, there must exist some value z in the intersection of $[x\ y]$ and $[u\ v]$. Let $[\![\mathbf{P}(B)]\!]^{M_3} = z$ and let A be satisfied by precisely the same worlds in M_3 that satisfy B. Then M_3 satisfies the premises and

$$\frac{[\![\mathbf{P}(A \wedge B \wedge T)]\!]^{M_3}}{[\![\mathbf{P}(B \wedge T)]\!]^{M_3}} = \frac{z}{z} = 1.$$

□

5 Restricted Entailment Problems

This section identifies two types of restricted probabilistic entailment problems involving only unconditional sentences: type-A and type-B. We prove that a particular subset of the inference rules from Figures 3 and 4 are complete for type-A problems, and we conjecture the completeness of a larger rule set for type-B problems.

In a type-A or type-B probabilistic entailment problem, $\langle \Psi, \phi \rangle$, Ψ is a finite set of sentences, each of which is either a *fact* or an *implication*. A *fact* is of the form $\mathbf{P}(\alpha) \in I$, and an *implication* is of the form $\mathbf{P}(\alpha \rightarrow \beta) \in I$, where α and β are atomic propositional sentences. Furthermore, the probabilistic component of an implication in a type-A problem must be [1 1]. In type-A problems ϕ must be an atomic proposition, and in type-B problems it must be a conjunction of atomic propositions. Thus every type-A problem is a type-B problem. The form of type-A and type-B problems is summarized in Table 1.

Table 1: Form of type-A and type-B problems.

Problem Type	Implications	Facts	Queries
type-A	$\mathbf{P}(\alpha \rightarrow \beta) \in [1\ 1]$	$\mathbf{P}(\alpha) \in I$	γ
type-B	$\mathbf{P}(\alpha \rightarrow \beta) \in I$	$\mathbf{P}(\alpha) \in I$	$\gamma_1 \wedge \cdots \wedge \gamma_n$

where $\alpha, \beta, \gamma, \gamma_i$ are all atomic propositions.

5.1 Type-A Problems

This section shows that inference rules (xx), (xxi), (xvi), (xvii), and (xviii) form a complete proof system for type-A probabilistic entailment problems. We call these five rules the type-A rules.

When the type-A rules are applied to type-A problems, α and β range over proposition letters. Since sentences in type-A problems are unconditional and all implications have a probabilistic component of [1 1], inference rules (xx) and (xxi) can be simplified to

(xx)

$$\frac{\mathbf{P}(\beta) \in [x\ y] \qquad \mathbf{P}(\beta \rightarrow \alpha) \in [1\ 1]}{\mathbf{P}(\alpha) \in [x\ 1]}$$

(xxi)

$$\frac{\mathbf{P}(\alpha) \in [x\ y] \qquad \mathbf{P}(\beta \rightarrow \alpha) \in [1\ 1]}{\mathbf{P}(\beta) \in [0\ y]}$$

We illustrate the type-A inference rules by using them to perform the derivation from Example 1 of Section 3. Suppose we wish to derive the tight entailment of A from sentences (1)–(4). A successful derivation is as follows:

$\mathbf{P}(A) \in [.2\ 1]$	rule (xx) applied to sentences (1) and (3)	(41)
$\mathbf{P}(A) \in [0\ .6]$	rule (xxi) applied to sentences (2) and (4)	(42)
$\mathbf{P}(A) \in [.2\ .6]$	rule (xvii) applied to sentences (41) and (42)	(43)

In working on a type-A probabilistic entailment problem, $\langle \Psi, \phi \rangle$, the type-A rules need only be applied a bounded number of times. Thus, because the rules are sound and complete, they can form the basis of an algorithm for solving type-A probabilistic entailment problems. To see that the number of rule applications can be bounded, first notice that rules (xx), (xxi), and (xvii) can only derive from Φ a sentence whose logical component is an atom that occurs in Φ. Furthermore, notice that the ability to infer a sentence whose logical component is ϕ is not affected by restricting the application of rules (xvi) and (xviii) to only those cases where they infer a sentence whose logical component is an atomic proposition that occurs in $\Psi \cup \{\phi\}$. We call this the *relevance restriction* and refer to the type-A rules with this restriction imposed as the *restricted type-A rules.* The restricted type-A rules can never generate new atomic propositions or new endpoints for the probability bounds. Since $\Psi \cup \{\phi\}$ is finite, it contains only a finite number of propositional sentences and a finite number of probability bounds. Therefore, only a finite number of sentences can be inferred using the restricted type-A rules.

We now show that the restricted type-A rules are complete for type-A probabilistic entailment problems. As previously mentioned, the addition of rule (xix), the interval expansion rule, makes the system complete in the usual sense.

Theorem 3 *Taken together, inference rules (xx), (xxi), (xvi), (xvii), and (xviii) with the relevance restriction are complete for type-A probabilistic entailment problems.*

Proof: Consider solving the type A probabilistic entailment problem $\langle \Psi, \phi \rangle$ by using the restricted type-A rules. If $\mathbf{P}(\alpha) \in \emptyset$ can be inferred for some α, then $\mathbf{P}(\phi) \in \emptyset$ can be inferred with rule (xviii). In such cases, the tightest entailed interval, $\emptyset$, is inferred. Now assume that no formula of the form $P(\alpha) \in \emptyset$ is inferable. If I is the tight entailment of ϕ from Ψ we need to show that for some $[x\ y] \subset I$ there is a derivation of $\mathbf{P}(\phi) \in [x\ y]$ from Ψ using the restricted type-A rules. Since $[x\ y] \subset I$ if and only if $x \in I$ and $y \in I$, it suffices to show that that some model of Ψ assigns ϕ probability x and some model of Ψ assigns ϕ probability y.

From the above discussion about the finiteness of inference with the restricted rules, it follows that for any atomic proposition α that occurs in $\Psi \cup \{\phi\}$ there is a smallest interval I such that $\mathbf{P}(\alpha) \in I$ can be inferred from Ψ using the restricted type-A rules. Call this interval $[l_\alpha\ u_\alpha]$. We construct a model, M, of Ψ such that $[\![\mathbf{P}(\phi)]\!]^M = l_\phi$. In a similar way, one can construct a model of Ψ such that $[\![\mathbf{P}(\phi)]\!]^M = u_\phi$.

Let $A_1, \ldots, A_n$ be the propositional letters that occur in $\Psi \cup \{\phi\}$ ordered so that $l_{A_i} \leq l_{A_j}$ if $i \leq j$. Let $\{w_1, \ldots, w_n\}$ be a set of worlds such that the only atoms satisfied by w_i are $A_i, \ldots, A_n$. Let M be the model that contains the worlds $\{w_1, \ldots, w_n\}$ and assigns probability l_{A_1} to w_1 and probability $l_{A_i} - l_{A_{i-1}}$ to each w_i, $2 \leq i \leq n$. So $[\![\mathbf{P}(A_i)]\!]^M = l_{A_i}$ for every $1 \leq i \leq n$. All that remains is to show that M satisfies Ψ. Let ψ be an arbitrary sentence in Ψ. Consider two cases:

ψ is $\mathbf{P}(A_i) \in [x\ y]$: Then $l_{A_i} \geq x$. Furthermore, $l_{A_i} \leq y$, or else $\mathbf{P}(A_i) \in \emptyset$ would be inferable, contrary to assumption. So, M satisfies ψ.

ψ is $\mathbf{P}(A_i \rightarrow A_j) \in [1\ 1]$: Inference rule (xx) guarantees that $l_{A_j} \geq l_{A_i}$ and thus, by the construction of M, every world in M that satisfies A_i also satisfies A_j. So M satisfies ψ. □

5.2 Type-B Problems

Consider now type-B probabilistic entailment problems. The proof system for this problem type consists of inference rules (xx)–(xxvi), (xvi), (xvii), and (xviii). All the propositional meta-variables in the inference rules are now taken to represent either atomic propositions or conjunctions of an arbitrary number of atomic propositions. Examples 3 and 4 from Section 4.1 are type-B entailment problems. In both those examples, the inference rules derive the tightest entailed interval for the target sentence. We conjecture these inference rules to be complete for type-B problems, though we do not have a proof.

6 Reasoning about Probabilistic Independence

The inference rules shown in Figures 1, 2, 3, and 4 do not cover all of the inference rules present in Quinlan's [31] INFERNO system. In this section we show that we can easily add generalizations of the rules that INFERNO has for reasoning about probabilistic independence. The ability to readily add new inference rules, such as those that capture the inferences performed in network-based systems, demonstrates the flexibility of our approach.

We extend the current language by introducing an independence operator. We express the fact that ϕ is independent of ξ given δ with an independence sentence of the form

(xxxiii)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \quad \mathbf{P}(\beta \mid \delta) \in [u\ v] \quad \mathsf{Indep}(\alpha, \delta, \beta)}{\mathbf{P}(\alpha \wedge \beta \mid \delta) \in [x \cdot u\ y \cdot v]}$$

(xxxiv)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \quad \mathbf{P}(\alpha \wedge \beta \mid \delta) \in [u\ v] \quad \mathsf{Indep}(\alpha, \delta, \beta)}{\mathbf{P}(\beta \mid \delta) \in [u/y\ \lfloor v/x \rfloor_1]}$$

(xxxv)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \quad \mathbf{P}(\beta \mid \delta) \in [u\ v] \quad \mathsf{Indep}(\alpha, \delta, \beta)}{\mathbf{P}(\alpha \wedge \beta \mid \delta) \in [x + u - x \cdot u \quad y + v - y \cdot v]}$$

(xxxvi)

$$\frac{\mathbf{P}(\alpha \mid \delta) \in [x\ y] \quad \mathbf{P}(\alpha \vee \beta \mid \delta) \in [u\ v] \quad \mathsf{Indep}(\alpha, \delta, \beta)}{\mathbf{P}(\beta \mid \delta) \in [\lceil (u - y)/(1 - y) \rceil^0 \quad (v - x)/(1 - x)]}$$

(xxxvii)

$$\frac{\mathsf{Indep}(\alpha, \delta, \beta)}{\mathsf{Indep}(\beta, \delta, \alpha)}$$

(xxxviii)

$$\frac{\mathsf{Indep}(\alpha, \delta, \gamma)}{\mathsf{Indep}(\beta, \delta\gamma)}$$

provided α and β are equivalent

Figure 7: Inference rules for independence.

$\mathsf{Indep}(\phi, \delta, \xi)$. The associated semantic definition is

$$[\![\mathsf{Indep}(\phi, \delta, \xi)]\!]^M = \text{True} \quad \text{iff} \quad [\![\mathbf{P}(\phi \wedge \xi \wedge \delta)]\!]^M = [\![\mathbf{P}(\phi \wedge \delta)]\!]^M \cdot [\![\mathbf{P}(\xi \wedge \delta)]\!]^M$$

This independence operator is similar to that used by Pearl [29] to axiomatize properties of independence.

Figure 7 shows six rules for reasoning about independence. Rules (xxxiii)–(xxxvi) deal with conjunctions and disjunctions of independent propositions. These rules are similar to rules (xxv)–(xxviii), but use independence information to derive tighter bounds. Rules (xxxiii) and (xxxiv) taken together are a generalization of the INFERNO rule labeled "A conjoins-independent $\{s_1, s_2, ..., s_n\}$." Rules (xxxv) and (xxxvi) are a generalization of the INFERNO rule labeled "A disjoins-independent $\{s_1, s_2, ..., s_n\}$."

7 Related Work

This section first compares the notion of anytime deduction to other work on anytime computation in artificial intelligence, and then compares our method for computing probabilistic entailment to other methods.

7.1 Other Work on Anytime Computation

The notion of anytime computation in AI can be traced back to our original proposal for performing deduction in probabilistic logic [16].[12,13] In that work the deduction method is called "convergent deduction" because the focus is on the convergent nature of the inference process. We call the procedure "anytime deduction" here because our principal focus is the ability to provide partial information at any time, regardless of whether convergence is ultimately attained.

Dean and Boddy [9] coined the term *anytime algorithm* in the context of work that explored issues in time-dependent planning. They defined an anytime algorithm as one that

> *i*) lends itself to preemptive scheduling techniques, *ii*) can be terminated at any time and will return some answer, and *iii*) returns answers that improve in some well-behaved manner as a function of time.

Criterion (*iii*) means that we have some notion of the marginal improvement in an answer as a function of a marginal increase in computation time. Since then researchers have presented anytime algorithms for other AI problems [10; 37]. The concept of an algorithm that produces approximate answers that improve with time has also appeared in the literature on data base theory [33; 8; 6; 36] and real-time systems [22; 21; 7]. In the literature on real-time systems, such algorithms are known as imprecise, monotone algorithms.

Concurrent with Dean and Boddy's work, Horvitz [19] presented a general decision-theoretic framework for reasoning about the optimality of computational methods operating under constraints on time and other resources. He contrasts *partial strategies* with traditional algorithms that either find a solution in the time given or provide no information. A partial strategy computes partial results that have some utility in an amount of time less than that required to compute a complete solution. Partial strategies are weaker than Dean and Boddy's anytime algorithms because they need not improve in a "well-behaved manner." Horvitz goes on to identify a stronger class of partial result strategies called *incremental-refinement* policies that improve their solutions as a "continuous or bounded-discontinuous, monotonically increasing function of allocated resources." Horvitz's incremental refinement policies correspond roughly to Dean and Boddy's anytime algorithms.

Because they must provide partial answers that are correct and improve monotonically, anytime deduction procedures are stronger than Horvitz's partial strategies. However, since they need not provide partial answers that are well-behaved in the manner specified by Dean and Boddy, anytime deduction procedures are weaker than anytime algorithms.

7.2 Other Methods for Computing Probabilistic Entailment

Toward the end of his paper, Nilsson suggests a way to use his method to compute bounds on conditional probabilities. Since $\mathbf{P}(\alpha \mid \beta) = \mathbf{P}(\alpha \wedge \beta)/\mathbf{P}(\beta)$, he suggests using the matrix

[12] Quinlan's [31] INFERNO system is also an anytime deduction procedure but he did not present or discuss it as such.

[13] In 1986 Haddawy [15] presented an inference system based on Michalski and Winston's [23] Variable Precision Logic that could vary the precision of its inference to produce an answer to a given query within a specified amount of time. That system, however, lacks the important property that all partial answers are correct.

method to compute bounds on the numerator and denominator and then using those bounds to compute bounds on $\mathbf{P}(\alpha \mid \beta)$. To obtain the lower bound one would divide the lower bound of the numerator by the upper bound of the denominator. The upper bound would be obtained similarly. The problem with this approach is that the numerator and denominator may be logially related in such a way that a tighter bound could be obtained. For example, suppose we are given that $\mathbf{P}(A \wedge (A \wedge B)) \in [.1\ .5]$ and $\mathbf{P}(A \wedge B) \in [.2\ .3]$ and we wish to compute $\mathbf{P}(A \mid (A \wedge B))$. From the two premises Nilsson's matrix method would determine that $\mathbf{P}(A \wedge (A \wedge B)) \in [.2\ .3]$. Computing the conditional probability as described above would produce $\mathbf{P}(A \mid (A \wedge B)) \in [.67\ 1]$. This is a sound inference but it is not tight. The tightest entailed interval, computed by our rule (xiii), is [1 1].

Several elaborations on Nilsson's basic method for computing probabilistic entailment have been presented in the literature. Van der Gaag [35] shows how a probabilistic entailment problem involving independence constraints may be structured using Pearl's [29] notion of an I-map and the linear optimization method applied to local groups of sentences. The approach requires the explicit representation of all probabilistic dependencies between sentences and thus treats sentences as atomic propositions. Shvaytser [32] presents a specialized inference procedure that identifies the set of sentences assigned probability one by a given model. Ursic [34] proposes computing probabilistic entailment for unconditional propositional probabilistic logic by using linear programming techniques in a more flexible way than Nilsson does. The approach allows one to partition the set of propositions and perform optimization on only a subset of the propositions at a time. The more propositions one considers simultaneously, the greater the precision of the method. As elaborations on Nilsson's basic method, all of these approaches have the drawback that they require finding consistent assignments of truth values to a set of sentences.

Bundy [4] presents a method for computing probabilistic entailment that he calls incidence calculus. Incidence calculus represents the probability of a logical sentence as an incidence or set of points. A point can be thought of as a possible world or as a particular outcome in a sample space. Points have probabilities associated with them (typically the distribution is taken to be uniform) and the probability of a sentence is just the sum of the probabilities of its associated points. The set of points need not be exhaustive—the greater the number of points, the greater the accuracy of the probabilities. Probabilistic inferences are made by performing set operations on the incidences.

Bundy presents an algorithm, the Legal Assignment Finder, that, given bounds on the incidences of a set of sentences, computes increasingly tighter bounds on the incidences of those and other sentences. The bounds are represented in terms of relations among the supremum and infimum of the incidences of the sentences. Since bounds on incidences induce bounds on the probabilities of their associated sentences, by tightening the bounds on the incidences of sentences the algorithm in effect computes increasingly tighter bounds on the probabilities of the associated sentences. This algorithm fits our definition of an anytime deduction procedure and is shown to be both sound and complete for unconditional sentences of $\mathcal{L}_{PL}$ [5].

The major drawback of Bundy's approach is the lack of a general mechanism for assigning the initial incidences to a set of sentences, given their probabilities. The difficulty arises because each point has a probability associated with it so that assigning incidences to sentences can result in specifying correlations among the sentences that were not specified in the original probabilities associated with the sentences. The need to assign initial incidences poses the same computational problems for incidence calculus as does the problem of computing consistent truth assignments for Nilsson's method for computing probabilistic entailment. Assigning initial incidences from probabilities requires determining consistent truth assignments to the set of sentences. This problem is NP-complete for propositional logic and undecidable for first-order logic. So the Legal Assignment Finder is an anytime deduction algorithm for only part of the probabilistic entailment problem.

If one is concerned only with finding a complete solution to a probabilistic entailment problem in propositional probabilistic logic, then anytime deduction may be no faster than systems based on the methods of Nilsson or Bundy. However, our approach has the major advantage that it can produce partial answers without expending great computational effort. Although we do not have a complete inference method for general probabilistic entailment, even an incomplete anytime deduction procedure may be preferable to a complete procedure based on Nilsson's or Bundy's methods.

Our anytime deduction procedure is most similar in spirit to Quinlan's [31] INFERNO system. INFERNO encodes formulas as a network, with nodes representing propositions and links between them representing relations between a node and a set of nodes. The relation may indicate that a node represents the conjunction, disjunction, or negation of another node or nodes. It may also indicate that the conditional probability of a node given another node is bounded by some value. Quinlan provides a set of combination and propagation rules, and points out that they are sound but not complete. In operation, INFERNO is an anytime deduction procedure much like ours. It computes probability bounds by initially assigning all propositions the trivial bounds [0 1] and then computing increasingly narrower intervals as some nodes are assigned tighter bounds and information is propagated. As in our method, different probability bounds for the same proposition can be intersected.

INFERNO differs from our approach in one major way: INFERNO can reason about only those logical dependencies that are are explicitly represented by links between nodes. For example, to represent the conjunction $A \wedge B$, the network must contain a node for A, a node for B, and a node for the conjunction. So INFERNO cannot generate new formulas from old formulas and subformulas. It can only reason about the probabilities of formulas and their subformulas explicitly encoded in its network. As a consequence, INFERNO cannot perform the inferences in Examples 2, 3, or 4. This limitation is inherent in the network approach and could not be remedied by simply adding inference rules.

In our approach, logical dependencies are represented by the logical structure of sentences and we have inference rules that combine logical sentences to create new logical sentences. For example, by introducing new formulas, which are not subformulas of the premises, rules (xxii)–(xxv) perform inferences of which INFERNO is not capable.

Amarger, DuBois and Prade [2] present a network propagation based method of computing bounds on conditional probabilities of interest given bounds on other conditional probabilities. Nodes in their networks represent propositions and edges represent conditional probabilities. They extend INFERNO with two new rules: the generalized Bayes' theorem and the quantified syllogism rule, which we have included as rules (viii) and (ix), respectively. They point out that the two rules are sound and locally complete. Although they do not define local completeness precisely, it seems to correspond to our notion of quasi-tightness. But the inference system is not globally complete. In addition to these two rules, they have the multiple derivation rule, so their inference system is also an anytime deduction procedure. They further extend INFERNO's network approach with rules for introducing new nodes. These new nodes are limited to conjunctions and disjunctions of nodes in the network.

Ng and Subrahmanian [25; 24] formulate a sound and complete inference system based on SLD deduction (as used in logic programming) for a class of probabilistic entailment problems that is almost totally disjoint from those addressed in this paper. Their system computes the tight entailment of a sentence of the form

$$\exists(\mathbf{P}(\alpha_1) \in I_1 \ \wedge \ \cdots \ \wedge \ \mathbf{P}(\alpha_n) \in I_n), \quad n \geq 1,$$

from a finite set of sentences, each of which is of the form

$$\forall(\mathbf{P}(\alpha_1) \in I_1 \ \wedge \ \cdots \ \wedge \mathbf{P}(\alpha_m) \in I_m \ \rightarrow \ \mathbf{P}(\beta) \in I), \quad m \geq 0$$

where β and each α_i is a conjunction or disjunction of atomic formulas. (In [25], β can only be an atomic formula.) In their latter paper [24] each interval I_i can be a complex expression containing variables, thus allowing one to express that the conditional probability of A given B is in the interval $[x\ y]$ by writing

$$\forall(\mathbf{P}(B) \in [V_1\ V_2] \rightarrow \mathbf{P}(A \wedge B) \in [x * V_1\ y * V_2]),$$

where V_1 and V_2 are variables ranging over probability values. Though the atomic formulas in their language may contain variables, they contain no function symbols of arity greater than zero, and the variables range over the zero-arity function symbols. Thus each sentence in the language is schematic for a finite set of variable free sentences and therefore their system is essentially propositional. Inference is performed by an initial stage in which new definite clauses are inferred from the original definite clauses followed by a second stage that reasons backward from the query in the style of logic programs. Because the need for the multiple derivation rule is "compiled out" by the first stage, their system does not obtain any of the characteristics of anytime deduction.

8 Summary and Conclusions

This paper has identified and discussed the important notion of anytime deduction procedures. Anytime deduction procedures can return useful partial information even before a complete proof is found by exploiting the capacity of any multi-valued logic (in this case probabilistic logic) to express intermediate results. Before the ultimate value, or value interval, of a target sentence is computed, it is possible to have an intermediate result stating that some truth values have been eliminated. Furthermore, an anytime deduction procedure based on inference rules—such as the one developed here for probabilistic logic—yields a proof that explains the line of reasoning used and justifies the conclusion.

We have presented a sound and quasi-tight set of inference rules for a propositional probabilistic logic that includes conditional probabilities. Although the inference rules embody no assumptions concerning the probability distribution over possible worlds, we have shown how they can be extended to exploit independence information when it is available. We have identified a subset of our inference rules that are complete for type-A probabilistic entailment problems.

Several directions remain open to future research. It remains to be seen whether a complete inference procedure can be formulated for all of propositional probabilistic logic. Beyond propositional logic, the applicability of anytime deduction to first-order probabilistic logic needs to be explored since the addition of quantification produces a language with greatly increased expressive power. The advantages of anytime deduction would be even greater for first-order probabilistic logic. Another direction worth exploring is the application of anytime deduction to multi-valued logics other than probabilistic logic.

Probabilistic inference is primarily useful because it provides information for decision making. Furthermore, the time/precision tradeoff made available by anytime deduction can be controlled effectively only in a particular decision context. Thus, the use of anytime deduction procedures in the context of decision procedures should be explored.[14] For example, anytime deduction could be used in a planning system to compute increasingly narrow bounds on the expected utility of alternative plans. These bounds would impose a partial order over the set of alternative plans. As the inference process progressed this set would becomed more ordered. Such a procedure could be stopped at any time to yield the current set of undominated plans, allowing a planning system to adapt flexibly under time constraints. Guidance in applying anytime deduction procedures to decision problems could be provided by Horvitz, Cooper, and Heckerman's [20] framework for reasoning about actions under time constraints. Their work explores how to optimally control methods of computing probability bounds in Bayes nets, showing how knowledge of the rate of convergence of the probability bounds can be used to control the optimal allocation of inference time.

Acknowledgements

We would like to thank Patrick Maher for his insightful comments on a draft of this paper. We thank Pravin Vaidya for sharing his expertise on optimization. This paper benefited from discussions with Yong Ma and Tomás Uribe and from the valuable comments of two anonymous referees. This work was partially supported by the second author's Shell doctoral dissertation fellowship and by NSF grant #IRI-9207262.

References

[1] E. Adams and H. Levine. On the uncertainties transmitted from premises to conclusions in deductive inferences. *Synthese*, 30:429–460, 1975.

[2] S. Amarger, D. Dubois, and H. Prade. Constraint propagation with imprecise conditional probabilities. In B. D'Ambrosio, P. Smets, and P. Bonissone, editors, *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, pages 26–34, July 1991.

[14] Pittarelli [30] discusses several alternative approaches to decision making based on probability intervals.

[3] F. Bacchus. *Representing and Reasoning With Probabilistic Knowledge.* MIT Press, Cambridge, Mass, 1990.

[4] Alan Bundy. Incidence calculus: A mechanism for probabilistic reasoning. *Journal of Automated Reasoning*, 1:263–283, 1985.

[5] Alan Bundy. Correctness criteria of some algorithms for uncertain reasoning using incidence calculus. *Journal of Automated Reasoning*, 2:109–126, 1986.

[6] P. Buneman, S. Davidson, and A. Watters. A semantics for complex objects and approximate queries. In *Proceedings of the Seventh Symposium on the Principles of Database Systems*, pages 305–314, March 1988.

[7] J.Y. Chung, J.W.S. Liu, and K.J. Lin. Scheduling periodic jobs that allow imprecise results. *IEEE Transactions on Computers*, July 1991. (to appear).

[8] S.B. Davidson and A. Watters. Partial computation in real-time database systems. In *Proceedings of the Fifth Workshop on Real-Time Software and Operating Systems*, pages 117–121, May 1988.

[9] T. Dean and M. Boddy. An analysis of time-dependent planning. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 49–54, Saint Paul, MN, August 1988.

[10] M. Drummond and J. Bresina. Anytime synthetic projection: Maximizing the probability of goal satisfaction. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 138–144, Boston, MA, July 1990.

[11] D. Dubois and H. Prade. Combination and propagation of uncertainty with belief functions. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 111–113, Los Angeles, California, August 1985.

[12] R. Fagin, J.Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. Technical Report RJ 6190 (60900), IBM Almaden Reasearch Center, April 1988. (also to appear in *Information and Computation*).

[13] T.D. Garvey, J.D. Lowrance, and M.A. Fischler. An inference technique for integrating knowledge from disparate sources. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 319–325, August 1981.

[14] B.N. Grosof. An inequality paradigm for probabilistic knowledge: The logic of conditional probability intervals. In J.F Kanal and J.F. Lemmer, editors, *Uncertainty in Artificial Intelligence.* Elsevier, 1986.

[15] P. Haddawy. Implementation of and experiments with a variable precision logic inference system. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 238–242, Philadelphia, August 1986.

[16] P. Haddawy and A.M. Frisch. Convergent deduction for probabilistic logic. In *Proceedings of the Third Workshop on Uncertainty in Artificial Intelligence*, pages 278–286, Seattle, Washington, July 1987.

[17] P. Haddawy and A.M. Frisch. Modal logics of higher-order probability. In R. Shachter, T.S. Levitt, J. Lemmer, and L.N. Kanal, editors, *Uncertainty in Artificial Intelligence 4*, pages 133–148. Elsevier Science Publishers, Amsterdam, 1990.

[18] J.Y. Halpern. An analysis of first-order logics of probability. *Artificial Intelligence*, 46:311–350, 1991.

[19] E.J. Horvitz. Reasoning under varying and uncertain resource constraints. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 111–116, Saint Paul, MN, August 1988.

[20] E.J. Horvitz. Reflection and action under scarce resources: Theoretical principles and empirical study. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 1121–1127, Detroit, MI, August 1989.

[21] V.R. Lesser, J. Pavlin, and E. Durfee. Approximate processing in real-time problem solving. *AI Magazine*, 9(1):49–61, Spring 1988.

[22] J.W.S. Liu, K.J. Lin, and S. Natarajan. Scheduling real-time, periodic jobs using imprecise results. In *Proceedings of the IEEE Real-Time Systems Symposium*, San Jose, California, December 1987.

[23] R.S. Michalski and P.H. Winston. Variable precision logic. *Artificial Intelligence*, 29(2), August 1986.

[24] Raymond Ng and V. S. Subrahmanian. A semantical framework for supporting subjective and conditional probabilities in deductive databases. *Journal of Automated Reasoning*, 10(2):191–235, 1993.

[25] Raymond T. Ng and V. S. Subrahmanian. Probabilistic reasoning in logic programming. In Z. W. Ras, M. Zemankova, and M. L. Enrich, editors, *Methodologies for Intelligent Systems,* **5**, pages 9–16. Elsevier, Amsterdam, 1990.

[26] Nils J. Nilsson. Probabilistic logic revisited. *Artificial Intelligence*, 59(1-2):39–42, February 1993.

[27] N.J. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–87, 1986.

[28] G. Paass. Probabilistic logic. In P. Smets, A. Mamdani, D. Dubois, and H. Prade, editors, *Non-Standard Logics for Automated Reasoning*, pages 213–244. Academic Press, 1988.

[29] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.

[30] M. Pittarelli. Decision making with linear constraints on probabilities. In *Proceedings of the Fourth Workshop on Uncertainty in Artificial Intelligence*, pages 283–290, University of Minnesota, August 1988.

[31] J.R. Quinlan. INFERNO: A cautious approach to uncertain inference. *The Computer Journal*, 26(3):255–269, 1983.

[32] H. Shvaytser. Probabilities that imply certainties. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 665–670, Boston, July 1990.

[33] K.P. Smith and J.W.S. Liu. Monotonically improving approximate answers to relational algebra queries. In *Proceedings of the IEEE Compsac*, Orlando, Florida, September 1989.

[34] S. Ursic. Generalizing fuzzy logic probabilistic inferences. In *Proceedings of the Third Workshop on Uncertainty in Artificial Intelligence*, pages 303–310, University of Pennsylvania, August 1986.

[35] L. van der Gaag. Computing probability intervals under independency constraints. In *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*, pages 491–497, Cambridge, MA, July 1990.

[36] S.V. Vrbsky, J.W.S. Liu, and K.P. Smith. An object-oriented query processor that returns monotonically improving approximate answers. Report UIUCDCS-R-90-1568, Univ. of Illinois, Feb. 1990.

[37] M. Zweben, M. Deale, and R. Garan. Anytime rescheduling. In *Proceedings of the 1990 DARPA Workshop on Innovative Approaches to Planning, Scheduling, and Control*, San Diego, November 1990.