Models and Algorithms for Probabilistic and Bayesian Logic

Pierre Hansen
Ecole des Hautes Etudes Commerciales
5255 avemie Decelles
Montreal (Quebec) H3T 1V6, CDN

Brigitte Jaumard
Ecole Poly technique de Montreal
C.P. 6079, Succ. Centre-Ville
Montreal (Quebec) H3C 3A7, CDN

Guy-Blaise Douanya Nguetse
Ecole Polytechnique de Montreal, CDN

Marcus Poggi de Aragao
Univ. Estadual de Campinas, Brazil

Abstract

An overview is given, with new results, of mathematical models and algorithms for probabilistic logic, probabilistic entailment and various extensions. Analytical and numerical solutions are considered, the former leading to automated generation of theorems in the theory of probabilities. Ways to restore consistency and relationship with Bayesian networks are also studied.

1 Introduction

Numerous models and algorithms have been proposed for reasoning under uncertainty in knowledge-based systems. Among these, models based on logic and the theory of probabilities are, after a period of relative disfavor, attracting much attention again. These models differ according to the independence assumptions made about the events or logical sentences under consideration and the amount of information requested from the expert or decision maker. At one extreme of the spectrum, well illustrated by the *probabilistic logic* and *probabilistic entailment* models of [Nilsson, 1986], no independance assumptions are made and only the available information is used. Moreover, this information may be vague, i.e., expressed by probability intervals instead of precise values. In probabilistic logic, the probabilities of being true of *m* logical sentences are given. It is asked whether these probabilities are consistent or not. In probabilistic entailment an additional logical sentence is considered and it is asked to find best possible lower and upper bounds on its probability of being true. In both cases the joint probability distribution on the set of possible outcomes is only partially specified. At the other extreme of the spectrum, the *Bayesian network* models (e.g., [Pearl, 1988], [Lauritzen *et ai,* 1988]) usually make strong independence assumptions and request sufficient information for the joint probability distribution to be entirely specified. When those requirements are satisfied the probability of any event may be computed, often in moderate time. Moreover, evidence can be efficiently propagated through the network. Attempts to combine both methodologies have been made by [Van der Gaag, 1990] [Van der Gaag, 1991] and by [Andersen *et ai,* 1994].

The purpose of this paper is to present an overview, with new results, of mathematical models and algorithms for probabilistic logic, probabilistic entailment and their extensions. Motivation stems from the facts that both problems have a long history and are the object of research dispersed among several literatures. This explains recent overly pessimistic statements as to the possibility of solving large instances. After stating the problems mathematically, analytical solution is studied. It is shown that one can use Fourier elimination or enumeration of vertices and extreme rays of polytopes. The latter approach leads to automated generation of theorems in the theory of probabilities. Numerical solution of large instances is then discussed. The column generation approach of linear programming, combined with specialized nonlinear 0-1 programming techniques to solve auxiliary subproblems (computation of the most negative or positive reduced costs) leads to algorithms efficient in practice. Extensions are then examined, i.e., use of probability intervals, conditional probabilities, Linear relations between probabilities and qualitative probabilities. Moreover, we show that *(i)* restoration of consistency through minimal changes in the probability intervals can be handled by the same type of models and *(11)* elimination of inconsistency through minimal deletion of logical sentences can be solved by combining column generation with branch-and-bound. The Bayesian logic model proposed by [Andersen *et a/.*, 1994] is finally investigated: we show that while this model is one of nonlinear nonconvex programming, many cases to which it applies can in fact be expressed as linear programs.

2 Probabilistic Satisfiability

The probabilistic logic problem of [Nilsson, 1986] may be expressed mathematically as follows. Let $S = \{S_1, S_2, \ldots, S_m\}$ be a set of m logical sentences defined on a set $X = \{x_1, x_2, \ldots, x_n\}$ of n boolean variables with the usual operators $\vee$ (disjunction), $\wedge$ (conjunction), and $\neg$ (negation). Let $\pi = (\pi_1, \pi_2, \ldots, \pi_m)$ be a vector of probabilities that these sentences are true. Are these sentences together with their probabilities consistent ? To make this question precise, consider all 2^n possible assignments X^k of truth values to the variables of X and let A^k be a m-vector such that $a_{ik} = 1$ if S_i is true for X^k and $a_{ik} = 0$ otherwise. Then the system of sentences and probabilities is consistent if and only if the linear system

$$\mathbf{1}.p = 1 \quad (1)$$
$$A.p = \pi \quad (2)$$
$$p \geq 0, \quad (3)$$

where $A = (A^k)$, has a solution. In other words, the system is consistent if and only if there is a probability distribution over the set of truth assignments such that, for each sentence the sum of probabilities of the truth assignments for which it is true is equal to its probability of being true.

In Nilsson's probabilistic entailment problem an additional sentence S_{m+1} is considered, a 2^n row vector A_{m+1} is defined by $a_{m+1,k} = 1$ if S_{m+1} is true for X^k and $a_{m+1,k} = 0$ otherwise. It is asked to find best possible lower and upper bounds on the probability that S_{m+1} is true. In other words, one seeks the optimal values of the linear programs

$$\min(\max)\ \pi_{m+1} = A_{m+1} \cdot p \quad (4)$$

subject to constraints (l)-(3). Note that [Nilsson, 1986] briefly discusses how to use standard techniques to reduce problems of first-order probabilistic logic to the propositional case. Instead of the names probabilistic logic and probabilistic entailment, [Georgakopoulos *et* a/., 1988] propose to use the name probabilistic satisfiability, in decision and optimization versions respectively. Indeed, [Nilsson, 1986] proposes useful *models* but not a *logici* i.e., a system of axioms and a study of inference rules, for reasoning about logic and probabilities. Such a logic extending the results of [Nilsson, 1986] has been explored by [Fagin *et* a/., 1990]. There are many other proposals in that area. Moreover, the name probabilistic satisfiability stresses the relationship of problem (l)-(3) with the classical satisfiability (SAT) problem of propositional logic (which corresponds to the case where is equal to 1). From now on, we use the name probabilistic satisfiability (PSAT).

The (PSAT) problem has a long history. The earliest occurrence of both versions appears to be in the classical work of [Boole, 1854] on *The Laws of Thought.* They are called *conditions of possible experience* and *general problem in the theory of probabihttes* respectively. Both problems also appear in the subjective approach to probability theory of [de Finetti, 1974], De Finetti's *fundamental theorem tn the theory of probability* ([de Finetti, 1974], p. 112) is indeed very close to Boole's *general problem.* The work of [Boole, 1854] on probability attracted little attention until it was revived in a seminal paper of [Hailperin, 1965] and discussed and extended in a subsequent book of the same author on *Boole a Logic and Probability* [Hailperin, 1986]. Several independent rediscoveries of (PSAT) have been made (including that of [Nilsson, 1986]).

3 Analytical Solution of PSAT

In his book of 1854 and in several contemporary and subsequent papers, [Boole, 1854] proposes procedures to solve (PSAT) approximately or exactly. The most efficient one works as follows: *(i)* express each sentence as a sum of products, each product involving all logical variables in direct or complemented form; *(it)* associate unknown probabilities to each of these products and identify the resulting sums to the given probabilities; *(iii)* eliminate in the equations so obtained and in the nonnegativity constraints on the probabilities the variables corresponding to the probabilities of the products.

More than a century later, [Hailperin, 1965] [Hailperin, 1986] discusses Boole's methods and shows that the above mentioned one is equivalent to Fourier elimination. Moreover, [Hailperin, 1965] expresses (PSAT) as the linear program (l)-(3) or (l)-(4) and shows that an analytical expression for the bounds on the probability 7r$_m$+i can be obtained by vertex enumeration of polytopes. To this effect, consider the dual D_{mm} (Anax) of(l)-(4):

$$\min(\max)\ y_0 + \pi y \quad (5)$$

subject to:

$$\mathbf{1} y_0 + A^t y \geq A^t_{m+1} \quad (\mathbf{1} y_0 + A^t y \leq A^t_{m+1}). \quad (6)$$

As the optimal solution of a linear program occurs at one (or several) of its extreme points, one has:

Theorem 1 *([Hailperin, 1965]) The best lower (upper) bound for π_{m+1} is given by the following convex (concave) piecewise linear function of the probability assignment:*

$$\pi_{m+1}(\pi) = \max_{j=1,2,\ldots,k_{max}} (1,\pi)^t y^j_{max} \quad (7)$$

$$(\pi_{m+1}(\pi) = \min_{j=1,2,\ldots,k_{min}} (1,\pi)^t y^j_{min}) \quad (8)$$

where y^j_{max} (y^j_{min}) for all j represent the k_{max} (k_{min}) extreme points of (D_{max}) ((D_{min})).

This result has recently been completed; the dual (D) of (1)-(3) is:

$$\min y_0 + \pi y \quad (9)$$

subject to:

$$\mathbf{1} y_0 + A^t y \leq 0. \quad (10)$$

Then, from the duality theory of linear programming

Theorem 2 *([Hansen* et al.*, 1995]) (PSAT) is consistent if and only if the inequality*

$$(1,\pi)^t r \leq 0 \quad (11)$$

holds for all extreme rays r of (D).

Theorems (1) and (2) lead to *complete analytical solutions* of instances of (PSAT) given in parametric form, i.e., with unspecified truth probabilities π_i. Once such solutions are at hand it suffices, for given values of the π_i to substitute in (11) to check consistency and in (7) and (8) to obtain best possible bounds.

As algorithms for enumeration of extreme points and rays of polytopes are readily available (e.g., [Chen *et al.*, 1991], [Dyer, 1983]) it is possible to obtain analytical solutions for given systems of sentences and probabilities in an entirely automated way. An example of such an automatically generated theorem in the theory of probabilities is the following ([Hansen *et al.*, 1995]):

Theorem 3 ***If logical sentences x_1 and x_2 have probability π_1 and π_2 respectively and the inference rule $(x_1 \vee x_2) \rightarrow x_3$ has probability π_3 then $\pi_1 + \pi_3 \geq 1$ and $\pi_2 + \pi_3 \geq 1$ must hold and the probability for x_3 to be true is between $\max\{\pi_1 + \pi_2 + \pi_3 - 2, 0\}$ and π_3. Moreover, these bounds are best possible.***

While results such as the above are easily obtained by direct reasoning, automation becomes useful when more sentences are considered as the numbers of conditions and of terms in the bounds increase rapidly.

4 Numerical Solution of PSAT

(PSAT) is NP-hard, as it is in NP and contains the NP-hard problem (SAT) as a particular case ([Georgakopoulos *et ai,* 1988]). Moreover, the problems (l)-(3) and (l)-(4) have a number of columns exponential in the size of the input when, as is usually the case, the size (or total number of variable occurrences) of the sentences S_i is bounded by a constant. (Note that this restriction on size is natural, as otherwise reading the input would require time exponential in the number of variables). So writing (l)-(3) or (l)-(4) explicitly already requires exponential time. This has led [Van der Gaag, 1990] [Van der Gaag, 1991] to surmise that solution of (PSAT) requires exponential time in general and not only in worst case. (In fact, many polynomial cases have been identified, see [Georgakopoulos *et ai,* 1988], [Kavvadias *et ai,* 1990], [Jaumard *et ai,* 1991]). [Nilsson, 1986] [Nilsson, 1993] stresses less formally, but as strongly, the difficulty of solving instances of (PSAT) with many variables and suggests Looking for heuristics. [Frisch *et at.,* 1994] propose under the name of *anytime deduction* a heuristic approach to (PSAT) based on sequential application of rules giving smaller and smaller intervals. This has the advantage of allowing reasoning to be followed step by step but may not yield best possible bounds. However, the powerful column generation technique of linear programming (see, e.g.,[Chvatal, 1983], chapter 18) can be brought to bear. This was proposed by [Zemel , 1982] for an application of (PSAT) to reliability, then for the general case by [Georgakopoulos *et ai,* 1988], whose work is extended in [Jaumard *et* a/., 1991], and ([Hooker , 1988], see also [Andersen *et ai,* 1994]). When solving a linear program by column generation a compact tableau is kept; at each iteration the entering column is found by solving a combinatorial subproblem and the tableau is updated following the rules of the revised simplex method. Finding the column with minimum (maximum) reduced cost at a current iteration is equivalent to minimization (maximization) of

$$a_{m+1,k} - u_0 - \sum_{i=1}^{m} a_{ik} u_i \qquad (12)$$

where the u_i are the dual variables associated with constraints (1) and (2). Associating the values true with 1 and *false* with 0, (12) may be rewritten

$$S_{m+1} - u_0 - \sum_{i=1}^{m} S_i u_i \qquad (13)$$

which is a nonlinear expression in the variables X_j with the operators V, A and -. These operators may be eliminated as $\neg x \equiv 1 - x$, $x \wedge y \equiv x \times y$ and $x \vee y \equiv x + y - x \times y$ where x and y are logical variables. Minimization (maximization) of the resulting nonlinear function in 0-1 variables can be done approximately by variable-depth search ([Kavvadias *et ai,* 1990]) or tabu search ([jaumard *et ai,* 1991]) and exactly by an algebraic method ([Jaumard *et ai,* 1991], [Crama *et ai,* 1990]) or by linearization ([Hooker , 1988], [Andersen *et ai,* 1994]). As an exact solution is only required when no more column with a reduced cost of adequate sign can be found heuristically, variable-depth and tabu search are useful even if one wants proved best possible bounds. Heuristics will be used as long as possible and followed by a usually more time-consuming exact method. The column generation technique has led to solve large instances of (PSAT), with up to 140 variables and 300 sentences ([Jaumard *et ai,* 1991]) in reasonable computing time. The number of columns generated is a very small proportion of the overall number in the instance (e.g., about 2100 columns for problems with 70 variables, and hence 2^{70} columns, and 200 sentences).

5 Extensions of PSAT

In addition to uncertainty, expressed by probabilities, expert knowledge often suffers from vagueness. Indeed giving a single value for the truth probability of a sentence is quite arbitrary in many situations. Vagueness may be expressed in (PSAT) by using probability intervals $[\underline{\pi}_i, \overline{\pi}_i]$ for the truth of sentences S, instead of single values. Then the expert is not forced to provide more information than he has. Generalizing (PSAT) in this way was already proposed by [Hailperin, 1965]. Constraints (2) are replaced by

$$\underline{\pi} \leq Ap \leq \overline{\pi}. \qquad (14)$$

The column generation technique for (PSAT) described above extends readily to this case, columns corresponding to slack or surplus variables being treated separately. The increase in computing time when replacing single probability values by intervals is moderate ([Jaumard *et ai,* 1991]).

Expert knowledge may also be precise in some situations only, which is expressed by using conditional probabilities $\pi_{i|j} = \text{prob}(S_i|S_j)$. Such conditional probabilities can be integrated into (PSAT) in several ways. As $\text{prob}(S_i|S_j) = \dfrac{\text{prob}(S_i S_j)}{\text{prob}(S_j)}$, one can use [Jaumard *et ai,* 1991] the two constraints:

$$A_\alpha p - \pi_{i|j}\pi_j = 0$$

$$A_\beta p = \pi_j$$

where $A_\alpha = (a_{\alpha,k})$ with $a_{\alpha,k} = 1$ if $S_i \wedge S_j$ is true and 0 otherwise, $A_\beta = (a_{\beta k})$ with $a_{\beta k} = 1$ if S_j is true for X^k and 0 otherwise. A more compact expression, obtained by elimination of π_j, is:

$$A_\gamma p = 0 \qquad (15)$$

where $A_\gamma = (a_{\gamma k})$ with $a_{\gamma k} = 1 - \pi_{i|j}$ if $S_i \wedge S_j$ is true, $-\pi_{i|j}$ if S_j is true and S_i is false, and $a_{\gamma k} = 0$ otherwise. Using (1) to add $\pi_{i|j}$ to all columns, one can also write

$$A_\delta p = \pi_{i|j}$$

where $A_\delta = (a_{\delta k})$ with $a_{\delta k} = 1$ if $S_i \wedge S_j$ is true, $a_{\delta k} = 0$ if S_j is true and S_i is false and $a_{\delta k} = \pi_{i|j}$ otherwise. This corresponds to the three-valued definition of conditional probability of [de Finetti, 1974]. Consistency conditions using this last form are derived by [Gilio, 1973], among others. If best possible bounds are sought for a conditional probability the objective function (4) must be replaced by:

$$\frac{A_\alpha p}{A_\beta p} \tag{16}$$

and the problem becomes a *hyperbolic* (or fractional) programming one. [Hailperin, 1986] observes that this problem can be reduced to a linear program with one more variable using a standard technique of [Charnes *et* a/., 1962]: one minimizes $A_a p$ adding to the constraints ABp = 1 and multiplying right-hand sides by a scaling factor *t;* once the solution is found, the probabilities p, are divided by *t.* Alternately [Jaumard *et ai,* 199l] one can apply the lemma of [Dinkelbach, 1967] for fractional programming and solve (l)-(3), (16) by a sequence of linear programs. Again column generation techniques apply and computing time is not much larger than for standard (PSAT) problems of the same size [Jaumard *et* a/., 1991]. Intervals for conditional probability values can be handled as for usual probabilities.

[Fagin *et* a/., 1990] develop a logic for reasoning about probabilities which extends the results of [Nilsson, 1986]. In particular they consider linear expressions in the probabilities *Wj.* If some of these are unknown, such expressions can be handled within the column generation approach, again by keeping separate explicit columns.

A step further is made by [Coletti, 1994] who considers qualitative probabilities or conditional probabilities: their values are unknown but a partial order on them is assumed to be given. The resulting generalization of (PSAT) remains linear for probabilities but is a nonlinear nonconvex problem for conditional probabilities and thus hard to solve. [Coletti, 1994] presents conditions of consistency for such problems.

If a system of sentences is not consistent, which easily happens after addition of rules by different experts, one may seek to restore consistency with minimal changes. A first criterion is to minimize the sum of increases of the probability intervals, possibly weighted to express the degree of confidence of the expert in his evaluations. This leads to the following linear program [jaumard *et* a/., 1991]:

$$\min \underline{w}\ell + \overline{w}u$$

subject to:

$$\mathbf{1}.p = 1, \quad \underline{\pi} - \ell \leq Ap \leq \overline{\pi} + u, \quad p, \ell, u \geq 0$$

where $\ell_i(u_i)$ in the decrease in the lower bound (increase in the upper bound) on the probability of S_i, $\underline{w}_i$ and $\overline{w}_i$ are attenuation factors for these changes.

Another criterion is to minimize the number of sentences to remove in order to restore consistency. The resulting problem, called probabilistic maximum satisfiability (PMAXSAT) as it generalizes the (MAXSAT) problem of propositional logic (e.g., [Hansen *et al.*, 1990]), is a mixed-integer programming one ([Hansen *et al.*, 1992]):

$$\min \mathbf{1}.y$$

subject to:

$$\mathbf{1}.p = 1, \quad \underline{\pi} - \ell \leq Ap \leq \overline{\pi} + u, \quad u + \ell \leq y,$$

$$p, \ell, u \geq 0, \quad y \in \{0,1\}^m.$$

Consistent problems with up to 185 sentences to which are added 15 to 25 more sentences, and which then become inconsistent, are readily solved, with a small number of branchings.

6 Bayesian Logic

In Bayesian networks, denoted $G = (V, E)$ (e.g., [Pearl, 1988]), nodes $v_j \in V$ are associated with simple events (or logical variables x_j, we assume here only two outcomes are possible for each event, i.e., true or false) and directed arcs (v_i, v_j) are used to represent probabilistic dependence among events. Morover, these netwroks are acyclic. The probabilities of each node conditioned on the values of its immediate predecessors are given. Then *the probability that a node is true, when conditioned on the truth values of all its non-successors, is equal to the probability that it is true, conditioned only on the truth values of its immediate predecessors.*

Example. [Andersen *et al.*, 1994] consider a network with six nodes and assume the following conditional probabilities to be given: $\text{prob}(x_4|x_5x_6) = 1$, $\text{prob}(x_4|(\neg x_5)x_6) = 1$, $\text{prob}(x_4|x_5(\neg x_6)) = 1$, $\text{prob}(x_4|(\neg x_5)(\neg x_6)) = 0$, $\text{prob}(x_2|x_4) = 0.4$, $\text{prob}(x_2|\neg x_4) = 0.05$, $\text{prob}(x_3|x_4) = 0.2$, $\text{prob}(x_3|\neg x_4) = 0.1$, $\text{prob}(x_1|x_2x_3) = 0.95$, $\text{prob}(x_1|(\neg x_2)x_3) = 0.8$, $\text{prob}(x_1|x_2(\neg x_3)) = 0.7$, $\text{prob}(x_1|(\neg x_2)(\neg x_3)) = 0.1$, as well as the marginal probabilities $\text{prob}(x_5) = 0.25$ and $\text{prob}(x_6) = 0.15$. As x_5 and x_6 are assumed to be independent, $\text{prob}(x_4) = \text{prob}(x_5 \vee x_6) = \text{prob}(x_5x_6) + \text{prob}((\neg x_5)x_6) + \text{prob}(x_5(\neg x_6)) = 0.3625$.

Let X_j denote x_j or $\neg x_j$. Then the probability of any truth assignment $X_1, X_2, \ldots, X_n$ can be computed by the chain rule

$$\text{prob}(X_1X_2\ldots X_n) =$$

$$\text{prob}(X_1|X_2X_3\ldots X_n)\text{prob}(X_2|X_3X_4\ldots X_n)\ldots\text{prob}(X_n)$$

and in view of the above mentioned independence assumption this can be done using the specified conditional probabilities only.

Example *(continued).* Removing nodes v_5 and v_6 as $\text{prob}(x_4)$ is known, one has

$$\text{prob}(X_1X_2X_3X_4) =$$

$$\text{prob}(X_1|X_2X_3X_4)\text{prob}(X_2|X_3X_4)\text{prob}(X_3|X_4)\text{prob}(X_4)$$

$$= \text{prob}(X_1|X_2X_3)\text{prob}(X_2|X_4)\text{prob}(X_3|X_4)\text{prob}(X_4);$$

for instance $\text{prob}(x_1(\neg x_2)(\neg x_3)(\neg x_4)) = 0.1 \times 0.95 \times 0.9 \times 0.7375 = 0.6306$.

The probability of any sentence S_i may be computed in a similar way. For other types of operations answerable by Bayesian networks and in particular for propagation of evidence, see, e.g., [Pearl, 1988], [Andersen *et ai,* 1994].

The *Bayesian Logic* proposed by [Andersen *et al,* 1994] consists in using the (PSAT) model to interpret probability statements associated with Bayesian networks and then to study various generalizations. To this effect conditional independence statements are encoded as additional *nonlinear* constraints. These constraints have the general form

$$\text{prob}(A, A_0|B, B_0, C, C_0) = \text{prob}(A, A_0|B, B_0) \qquad (17)$$

where *A, B* and *C* are sets of propositional variables, with *\A* = a, *\B* = 6, *\C* — *c* and Ao, Bo and Co are sets of fixed atomic propositions. [Andersen *et ai,* 1994] show that there are $(2^a - 1)2^b(2^c - 1)$ nonredundant constraints among those described by (17). From the definition of conditional probability, (17) is equal to

$$\text{prob}(A, A_0, B, B_0, C, C_0).\text{prob}(B, B_0)$$

$$= \text{prob}(A, A_0, B, B_0).\text{prob}(B, B_0, C, C_0).$$

[Andersen *et ai,* 1994] propose to solve the extended (PSAT) model with constraints (17) by generalized Benders decomposition. Following that approach the problem is split into a nonlinear master problem in the *n* variables and a linear subproblem in the p variables, of the (PSAT) type. The subproblem is used to generate from its dual, linear constraints in the *n* variables, called Benders cuts, as long as it is infeasible. These cuts are added to the master problem. The procedure stops after a finite number of steps when the master problem is infeasible or the subproblem is feasible. The master problem has the form of a signomial geometric program for which specialized algorithms exist. Such problems belong to global optimization and only instances with few variables can be solved in reasonable time. This apparently limits the scope of Bayesian logic, even if the number of *π* variables is much smaller than the number of *p* variables. Fortunately, there are many cases in which *one need only add linear constraints* to (PSAT) to express the independence assumptions of Bayesian networks and generalizations of them.

Theorem 4 *Computing the probability of a sentence S, in a Bayesian network can be expressed as a (*PSAT) *problem with conditional probabilities.*

Proof. Conditional probabilities for nodes given the truth value of their immediate predecessors can be expressed by (15) and marginal probabilities by (2). For independence conditions, let *Bj* denote the set of atomic propositions associated with immediate predecessors of V_J and *Aj* a similar set for non immediate predecessors *of vj.* Then the condition

$$\text{prob}(x_j|B_j, A_j) = \text{prob}(x_j|B_j) \qquad (18)$$

is equivalent to the expression

$$\text{prob}(x_j, B_j, A_j) = \text{prob}(x_j|B_j)\text{prob}(B_j, A_j)$$

which is linear as the right-hand side of (18) is given. Finally, independence of sets A of nodes without predecessors is expressed by identifying prob(A) with the product of the corresponding marginal probabilities π. ■

Example ***(continued).*** **Assume one seeks the value of prob($x_2|x_1x_3$). Associate the probabilities $p_1, p_2, \ldots, p_{16}$ to the truth assignements as follows:**

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8
x_1	1	1	1	1	1	1	1	1
x_2	1	1	1	1	0	0	0	0
x_3	1	1	0	0	1	1	0	0
x_4	1	0	1	0	1	0	1	0

	p_9	p_{10}	p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	p_{16}
x_1	0	0	0	0	0	0	0	0
x_2	1	1	1	1	0	0	0	0
x_3	1	1	0	0	1	1	0	0
x_4	1	0	1	0	1	0	1	0

Using the technique of [Charnes *et al.*, 1962], the objective function is expressed by

$$\min(\max)\, p_1 + p_2\ ;$$

subject to:

$$p_1 + p_2 + p_5 + p_6 = 1\ ;$$

the conditional probabilities, using (15), by

$$0.8p_1 - 0.2p_3 + 0.8p_5 - 0.2p_7 + 0.8p_9 - 0.2p_{11} + 0.8p_{13} - 0.2p_{15} = 0$$

$$0.9p_2 - 0.1p_4 + 0.9p_6 - 0.1p_8 + 0.9p_{10} - 0.1p_{12} + 0.9p_{14} - 0.1p_{16} = 0$$

$$0.95p_2 + 0.95p_4 - 0.05p_6 - 0.05p_8 + 0.95p_{10} + 0.95p_{12} - 0.05p_{14} - 0.05p_{16} = 0$$

$$0.6p_1 + 0.6p_3 - 0.4p_5 - 0.4p_7 + 0.6p_9 + 0.6p_{11} - 0.4p_{13} - 0.4p_{15} = 0$$

$$0.05p_1 + 0.05p_2 - 0.95p_9 - 0.95p_{10} = 0$$

$$0.3p_3 + 0.3p_4 - 0.7p_{11} - 0.7p_{12} = 0$$

$$0.2p_5 + 0.2p_6 - 0.8p_{13} - 0.8p_{14} = 0$$

$$0.9p_7 + 0.9p_8 - 0.1p_{15} - 0.1p_{16} = 0;$$

the marginal probability by

$$p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15} - 0.3625t = 0;$$

the normalizing constraint by

$$\sum_{i=1}^{16} p_i - t = 0$$

the independence constraints prob($X_2|X_3X_4$) = prob($X_2|X_4$) by

$$0.6p_1 - 0.4p_5 + 0.6p_9 - 0.4p_{13} = 0$$

$$0.95p_2 - 0.05p_6 + 0.95p_{10} - 005p_{14} = 0;$$

and the independence constraints prob($X_1|X_2X_3X_4$) = prob($X_1|X_2X_3$) by

$$0.05\,p_1 \quad -0.95\,p_9 \quad = 0$$

$$0.3\,p_3 \quad -0.7\,p_{11} \quad = 0$$

$$0.2\,p_5 \quad -0.8\,p_{13} \quad = 0$$

$$0.9\,p_7 \quad -0.1\,p_{15} \quad = 0.$$

The optimal value is 0.26863.

Clearly the number of sets of non-immediate predecessors of a node may be exponential. However, not all corresponding constraints need be written. [Lauritzen *et al*., 1988] explain how to represent independence relations by an undirected graph G' in which all pairs of immediate predecessors are joined and edges are added until the graph is triangulated. Then the joint probability distribution can be expressed as a product of marginal probability distributions on the maximal cliques of G', adequately scaled. [Van der Gaag, 1990] [Van der Gaag, 199l] proposes to use this property in a decomposition method for (PSAT), discussed in a companion paper ([Douanya *et al,* 1995]). It is shown there that the usual (PSAT) model gives the same bounds as the decomposition-based version. Consequently (PSAT) takes implicitly into account in the computation of the bounds the conditional independence constraints (18) involving variables which do not all belong to the same maximal clique.

Example *(continued).* A graph G' associated with the example after deletion of *v5* and *v6* is composed of triangles on v1,v2,V3 and on *V2,v3,V4.* This shows that when computing bounds on prob (x2Ix1 x3 one neeed not take explicitly into account the constraints prob$(X_1|X_2X_3X_4) = \text{prob}(X_1|X_2X_3)$, i.e., the four last ones listed above.

[Andersen c/ a/., 1994] also explore cases where the number of independence constraints is limited. The main interest of Bayesian logic is not, however, to propose an alternate method for the computation made in Bayesian networks, but to consider more general assumptions.

Example *(continued).* Assume as done by [Andersen *et* a/., 1994] that the atomic propositions *x5* and x6 are not independent. Then, 0.25 < prob(x) < 040. Replacing the line giving the marginal probability of x_4 by

$$p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15} - 0.25t \geq 0$$

$$p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15} - 0.40t \leq 0$$

minimizing and maximizing yields bounds of 0.21786 and 0.28358. Note that when using Benders decomposition the computation of the lower bound required 57 iterations, i.e., solution of 57 (PSAT) and 57 signomial geometric programming problems.

As discussed in [Andersen *et* a/., 1994], many other extensions of Bayesian networks can be considered within the (PSAT) framework: one can replace single probability values by intervals, add constraints of different types than the conditional implications, allow for networks with cycles, etc. Not all extensions will remain linear. For instance, if in the example, the marginal probabilities for *x5* and *X6* are replaced by intervals and the independence assumption is kept a quadratic constraint

$$\text{prob}(x_5 \wedge x_6) = \text{prob}(x_5)\text{prob}(x_6)$$

arises. The resulting quadratic programs can be solved in many ways using global optimization techniques. Finding which are most efficient is an open problem.

To conclude, (PSAT) appears to be a flexible and computationnally tractable model for reasoning under uncertainty. It has already been extended in many ways, while remaining linear. Further exploration of the problems which may be so expressed and of solution methods for the nonlinear case are attractive topics for future research.

References

[Andersen *et at.,* 1994] K.A. Andersen and J.N. Hooker. Bayesian Logic. *Decision Support Systems,* 11:191-210, 1994.

[Boole, 1854] G. Boole. *An Investigation of the Laws of Thought, on which are Founded the Mathematical Theories of Logic and Probabilities.* Walton and Maberley, London, 1854 (reprint, New York: Dover 1958).

[Charnes *et ai,* 1962] A. Charnes and W.W. Cooper. Programming with Linear Fractional Functional. *Naval Research Logistics Quaierly,* 9:181-186, 1962.

[Chen *et ai,* 1991] P.-C Chen, P. Hansen, and B. Jaumard. On-line and Off-line Vertex Enumeration by Adjacency Lists. *Operations Research Letters,* 10(7):403-409, 1991.

[Chvatal, 1983] V. Chvatal. *L inear Programming.* San Francisco, Freeman, 1983.

[Coletti, 1994] G. Coletti. Coherent Numerical and Ordinal Probabilistic Assessments. *IEEE Transactions on Systems, Man and Cybernetics,* 24:1747-1754, 1994.

[Crania *et ai,* 1990] Y. Crama, P. Hansen, and B. Jaumard. The Basic Algorithm for Pseudo-Boolean Programming Revisited. *Discrete Applied Mathematics,* 29:171-186, 1990.

[Dinkelbach, 1967] W. Dinkelbach. On Nonlinear Fractional Programming. *Management Science,* 13:492-498, 1967.

[Douanya *et ai,* 1995] G.B. Douanya Nguetse, P. Hansen, and B. Jaumard. Probabilistic Satisfiability and Decomposition. To appear in the Proceedings of the ECSQARU 95 Conference, *Lectures Notes tn Computer Science,* 1995.

[Dyer, 1983] M E. Dyer. On the Complexity of Vertex Enumeration Methods. *Mathematics of Operations Research,* 8(3):381-402, 1983.

[Fagin *et* a/., 1990] R. Fagin, J.Y. Halpern, and N. Megiddo. A Logic for Reasoning about Probabilities. *Information and Computation,* 87(l/2):78-128, 1990.

[de Finetti, 1974] B., de Finetti. *Theory of Probability.* Wiley, New York, Vol. 1 & 2, 1974, 1975.

[Frisch *et ai,* 1994] A.M. Frisch and P. Haddawy. Anytime Deduction for Probabilistic Logic. *Artificial Intelligence,* 69:93-122, 1994.

[Georgakopoulos *et* a/., 1988] G. Georgakopoulos, D. Kavvadias, and C.H. Papadimitriou. Probabilistic Satisfiability. *Journal of Complexity,* 4:1-11, 1988.

[Gilio, 1973] A. Gilio. Probabilistic Consistency of Knowledge Bases in Inference Systems. In M. Clarke *et al.* (editors) *Symbolic and Quantitative Approaches* *to Reasoning under Uncertainty,* Berlin: Springer *Lectures Notes* in *Computer Science,* 747:160-167, 1973.

[Hailperin, 1965] T. Hailperin. Best Possible Inequalities for the Probability of a Logical Function of Events. *American Mathematical Monthly,* 72:343-359, 1965.

[Hailperin, 1986] T. Hailperin. *Boole's Logic and Probability.* Studies in Logic and Foundations of Mathematics 85, North Holland, New York, 1986, 2nd edition (first edition, 1976).

[Hansen et *ai,* 1990] P. Hansen and B. Jaumard. Algorithms for the Maximum Satisfiability Problem, *Computing,* 44:279-303, 1990.

[Hansen *et al,* 1992] P. Hansen, B. Jaumard and M. Poggi de Aragao. Mixed-Integer Column Generation Algorithms and the Probabilistic Maximum Satisfiability Problem. *Integer Programming and Combinatorial Optimization II,* (Balas, E., G. Cornuejols and R. Kannan, Editors), Pittsburgh: Carnegie Mellon University, 165-180, 1992.

[Hansen *et ai,* 1995] P. Hansen, B. Jaumard, and M. Poggi de Aragao. Boole's Conditions of Possible Experience and Reasoning Under Uncertainty, to appear in *Discrete Applied Mathematics,* 1995.

[Hooker , 1988] J. Hooker. A Mathematical Programming Model for Probabilistic Logic. *Working paper* 05-88-89, GSIA, Carnegie Mellon University, 1988.

[Jaumard *et ai,* 199l] B. Jaumard, P. Hansen, and M. Poggi de Aragao. Column Generation Methods for Probabilistic Logic. *ORSA Journal on Computing,* 3:135-148, 1991.

[Kavvadias *et ai,* 1990] D. Kavvadias and C.H. Papadimitriou. Linear Programming Approach to Reasoning about Probabilities. *Annals of Mathematics and Artificial Intelligence,* 1:189-205, 1990.

[Lauritzen *et ai,* 1988] S.L. Lauritzen and D.J. Spiegelhalter. Local Computations with Probabilities on Graphical Structures and their Applications to Expert Systems. *Journal of the Royal Statistical Society,* B50:157-224, 1988.

[Nilsson, 1986] N.J. Nilsson. Probabilistic Logic. *Artificial Intelligence,* 28:71-87, 1986.

[Nilsson, 1993] N.J., Nilsson. Probabilistic Logic Revisited. *Artificial Intelligence,* 59:39-42, 1993.

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

[Van der Gaag, 1990] L.C. Van der Gaag. *Probability-Based Models for Plausible Reasoning.* Ph.D. Thesis, University of Amsterdam, 1990.

[Van der Gaag, 1991] L.C. Van der Gaag. Computing Probability Intervals Under Independency Constraints. In P.P. Bonissone, M. Henrion, L.N. Kanal and J.F. Lemmer (Editors) *Uncertainty in Artificial Intelligence,* 6:457-466, 1991.

[Zemel , 1982] E. Zemel. Polynomial Algorithms for Estimating Networks Reliability. *Networks,* 12:439-452, 1982.