Geometric Foundations for Interval-Based Probabilities

Vu Ha and Peter Haddawy

Decision Systems and Artificial Intelligence Lab

Dept. of EE&CS

University of Wisconsin-Milwaukee

Milwaukee, WI 53201

$\{vu, haddawy\}@cs.uwm.edu$

Abstract

The need to reason with imprecise probabilities arises in a wealth of situations ranging from pooling of knowledge from multiple experts to abstraction-based probabilistic planning. Researchers have typically represented imprecise probabilities using intervals and have developed a wide array of different techniques to suit their particular requirements. In this paper we provide an analysis of some of the central issues in representing and reasoning with interval probabilities. We use the well-developed area of convex geometry as the underlying foundation for our analysis. In particular, we point out the ubiquity of the probability cross-product operator, a generalization of which is known in the convex geometry literature as the Minkowski operator. We perform an extensive study of this operator, the result of which provides insight into the sources of the strengths and weaknesses of various approaches to handling probability intervals. We demonstrate the application of our results to the problems of inference in interval Bayesian networks and projection of abstract probabilistic actions.

1 Introduction

There emerges in recent years a wide range of problems that demand effective methods for handling imprecise probabilities. Imprecise probabilities occur, for example, when the available domain knowledge is insufficient to specify exact probabilities [7]. They are inherent in aggregating knowledge from multiple experts, and in group decision making [15]. They may result from the abstraction of more detailed probabilistic models [9], and are useful in studying sensitivity and robustness in probabilistic inference (for example, inference in Bayesian networks [2]). Finally, imprecise probabilities are advocated as alternative representations of uncertainty by researchers who do not feel comfortable with the *Strict Bayesianism* paradigm that is required in exact probabilistic models [12]. Chrisman [1] provides a wealth of references to various works on imprecise probabilities.

Imprecise probabilities are often captured by intervals. For example, the sensitivity analysis of Bayesian network inference typically involves varying probabilistic parameters on some intervals and examining the range of the answer to a probabilistic query, which is also an interval. Researchers working on the problems mentioned in the previous paragraph have thus developed a number of techniques for handling probability intervals. But to this date, we still lack a uniform yet comprehensive analysis of the central issues concerning handling probability intervals. In this paper, we attempt to make the first steps towards such an analysis. Our approach rests on the well-developed area of finite convex geometry, and thus is applicable for problems with discrete finite probability distributions only.

Researchers have previously used concepts of convex geometry such as convex sets, especially polytopes, to represent sets of probability distributions associated with imprecise probabilities. The main contribution of this paper is a thorough analysis of the so-called *cc-operator*. Cc-operator is an interval generalization of the probability cross-product, an operator that is ubiquitous in computations such as action/plan projection, expected utility evaluation, and Bayesian network propagation. Cc-operator turns out to be a special form of the well-known Minkowski operator. This analysis provides insight into many issues of the abstraction-based probabilistic planning framework of [9]. Using the cc-operator, we develope an algorithm that is based on Dechter's Bucket Elimination algorithm [3] and that answers probabilistic queries in interval Bayesian networks. The probability bounds computed by this algorithm are correct but, unfortunately, not tight. We suggest an explanation for the difficulty of computing tight probabilistic bounds in the above frameworks.

Cc-operator can also be used to construct a data structure - called cc-trees - that represents sets of probability distributions. We compare cc-trees with existing structures of sets of distributions such as polytopes, lower probability functions, and belief functions, in different situations such as action/plan projection and Bayesian evidential belief updating [1].

2 Probability Cross-Product and the CC-Operator

We start with a brief introduction of some basic concepts of convex geometry. For more details, see e.g. [8].

The *d-dimension Euclidean Space* is the d-dimension vector space R^d equipped with the inner product $\langle\rangle$, which is defined for any pair of points $x, y \in R^d$ as: $\langle x, y\rangle = \sum_{i=1}^{d} x_i y_i$. A *convex combination* of the points $x_1, \ldots, x_n \in R^d$ is a linear combination $\lambda_1 x_1 + \cdots + \lambda_n x_n$, where $\lambda_i \geq 0, i = 1, \ldots, n$, and $\lambda_1 + \cdots + \lambda_n = 1$. The coefficients λ_i are called *convex coefficients*, and the sum is denoted by $\sum^{cc}$. This sum is also called the *probability cross product* of 2 vectors whose components are λ_i and x_i. For any set $K \subseteq R^d$, the *convex hull* of K, denoted by $\mathrm{conv}(K)$ is the set of all convex combinations of the points from K. A set $K \subseteq R^d$ is said to be *convex* if $K = \mathrm{conv}(K)$. A mapping ϕ from a convex set $K \subseteq R^d$ to R^e is said to be a *convex mapping* if it preserves convex combinations, i.e. $\phi(\sum_{i=1}^{n}{}^{cc} \lambda_i x_i) = \sum_{i=1}^{n}{}^{cc} \lambda_i \phi(x_i)$, for all points $x_i \in K$ and convex combinations λ_i. A convex mapping always maps convex sets into convex sets. An intersection of finitely many closed halfspaces is called a *polyhedron*. Polyhedra are convex. Bounded polyhedra are called *polytopes*. Polytopes can be alternatively defined as convex hulls of finite sets of points. The set of *extreme points* (or *vertices*)

[1] This part is left out of this extended abstract due to length limitations.

of a polytope K is the smallest set S such that $K = \text{conv}(S)$. The polytope whose vertices are the points $(1, 0, \ldots, 0), (0, 1, \ldots, 0), \ldots, (0, 0, \ldots, 1)$ is exactly the set $\mathcal{S}$ of all probability distributions over a sample space Ω of size d and is called the *probability simplex.* For example, the probability distributions P over the sample space $\Omega = \{s_1, s_2, \ldots, s_d\}$ corresponds to the point $(P(s_1), P(s_2), \ldots, P(s_d))$ in $\mathcal{S}$.

We now define the generalized version of the probability cross product, the cc-operator. In the rest of the paper, an interval is implicitly understood as a closed subinterval of $[0, 1]$.

Definition 1 (The CC-Operator) [2] *The cc-operator $\otimes$ defined by an interval vector $\vec{\Lambda} = (\Lambda_1, \ldots, \Lambda_n)$ is the function that maps a vector $\vec{w} = (w_1, \ldots, w_n)$ of sets of points in R^d to the set $\vec{\Lambda} \otimes \vec{w}$ that consists of all points of the form $\vec{\lambda} \otimes \vec{x} = \sum_{i=1}^{n}{}^{cc} \lambda_i x_i$, where $\lambda_i \in \Lambda_i, x_i \in w_i, i = 1, \ldots, n$. We sometimes write $\vec{\Lambda} \otimes \vec{w}$ as $\sum_{i=1}^{n}{}^{cc} \Lambda_i w_i$.*

The cc-operator sum $\vec{\Lambda} \otimes \vec{w}$ is a special case of Minkowski sum, a familiar notion in convex geometry. The difference between the two sums is that Minkowski sum does *not* require the coefficients λ_i to be convex coefficients. Below are a few properties of the cc-operator, some of which were reported in [9]. These results constitute the basis for our discussion of abstraction-based probabilistic planning in Section 3 and interval Bayesian network inference in Section 4.

1. *Probability distributions and the cc-operator* [9]. Any distribution $P \in \mathcal{S}$ can be written as $P = (P(s_1), \ldots, P(s_d)) \otimes (s_1, \ldots, s_d)$, where s_i represents the ith vertice of the probability simplex.

2. *$\otimes$ is closed on $2^{\mathcal{S}}$* [9]. If w_i are sets of distributions, then $\vec{\Lambda} \otimes \vec{w}$ is also a set of distributions.

3. *The cc-operator is closed on polytopes.* If w_i are polytopes, then $K = \vec{\Lambda} \otimes \vec{w}$ is also a polytope.

4. *Polytopes and the cc-operator* [9]. If each of the intervals Λ_i is the entire interval $[0, 1]$, we denote $\vec{\Lambda} \otimes \vec{w}$ as $cc(\vec{w})$, or $cc(w_1, \ldots, w_n)$. If K is a polytope with extreme points $\{x_1, \ldots, x_n\}$, then $K = \text{conv}(\{x_1, \ldots, x_n\}) = cc(\{x_1\}, \ldots, \{x_n\})$.

[2] "cc" stands for "convex combinations".

5. *Interval cross-croduct* [9]. If w_i are closed intervals of R (they need not be subintervals of [0 1]), then $\sum_{i=1}^{n} {}^{cc}\Lambda_i w_i$ is also a closed interval whose bounds can be computed using the greedy knapsack algorithm [3]. This special cc-operator sum is denoted by $\sum_{i=1}^{n} {}^{icc}\Lambda_i w_i$ (the "i" letter stands for "interval").

6. *The cc-operator and convex hull commute* [9]. $\text{conv}\left(\sum_{i=1}^{n} {}^{cc}\Lambda_i w_i\right) = \sum_{i=1}^{n} {}^{cc}\Lambda_i \text{conv}(w_i)$.

7. *The cc-operator and convex mappings commute.* Suppose that $K \subseteq R^d$ is a convex set and $f : K \to R^e$ is a convex mapping. For any $S \subseteq K$, defined $f(S)$ as $f(S) = \{f(x)|x \in S\}$. Then $f(\sum_{i=1}^{n} {}^{cc}\Lambda_i w_i) = \sum_{i=1}^{n} {}^{cc}\Lambda_i f(w_i)$.

3 Abstraction-Based Probabilistic Planning

Probabilistic Actions [4]. A *probabilistic action* A is usually defined as an $A : \Omega \to \mathcal{S}$ function that maps a state $s \in \Omega$ of the world into a probability distribution $A(s)$ over possible states of the world. For example, the action in Figure 1.a maps state s_1 into the probability distribution $P = A(s_1)$, where $P(s_1) = .4$ and $P(s_2) = .6$, and maps states s_2 and s_3 to some other distributions $A(s_2), A(s_3)$ over $\Omega = \{s_1, s_2, s_3\}$. This function $(A : \Omega \to \mathcal{S})$ is then extended to a function that maps a probability distribution $P \in \mathcal{S}$ to a probability distribution $Q = A(P) \in \mathcal{S}$ as follows: $Q = E_P[A(s)] = \sum_{s \in \Omega} P(s)A(s)$

Probabilistic Plans, Expected Utility Evaluation of Plans. A *probabilistic plan pl* is a finite sequence of probabilistic actions: $pl = \{A_1, \ldots, A_n\}$. Given an initial world $w_{ini} \in \mathcal{S}$ that is a probability distribution over the state space Ω, the projection of pl on w_{ini} is the process of determining the final world $w_{final} = pl(w_{ini}) = A_n(\ldots A_1(w_{ini})\ldots) \in \mathcal{S}$. Given a utility function $u : \Omega \to R$ that assigns to every state of the world a real-valued utility, the expected utility of a plan pl is defined as $E_u(pl) = \sum_{s \in \Omega} w_{final}(s)u(s)$, where $w_{final} = pl(w_{ini})$. It is convenient to extend the definition of

[3]Also called *annihilation/reinforcement algorithm* by Tessem [16].

[4]This model for probabilistic action has been used - albeit in slightly modified form - in various probabilistic planners such as BURIDAN [11] and DRIPS [10].

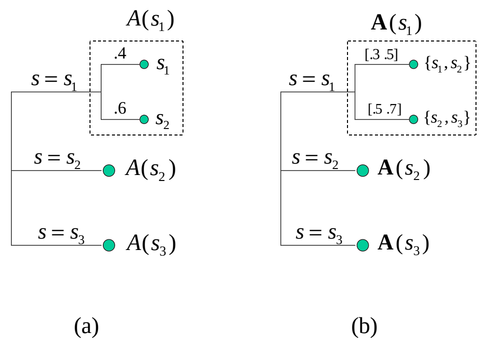

Figure 1: Examples: (a) a concrete action and (b) an abstract action

the utility function u as an $u : \mathcal{S} \to R$ function, where $u(P) = E_P[u(s)] = \sum_{s\in\Omega} P(s)u(s), \forall P \in \mathcal{S}$. Now, the expected utility of a plan pl, executed on an initial world w_{ini} can be written as $u(w_{final}) = u(pl(w_{ini}))$. A plan pl is *optimal* if it has the maximum expected utility among all candidate plans.

Probabilistic actions and utility functions, as functions that have $\mathcal{S}$ as their domains, have the following important property.

Theorem 1 *Probabilistic actions, probabilistic plans, and utility functions are convex mappings and thus commute with the cc-operator. As a consequence, if the initial world w_{ini} is a probability distribution over the states: $w_{ini} = \sum^{cc}\lambda_i s_i$, then the expected utility of a plan pl can be written as $u(pl(\sum^{cc}\lambda_i s_i)) = \sum^{cc}\lambda_i u(pl(s_i))$.*

Abstract Actions and Plans. From now on, we will call probabilistic actions *concrete actions*. We generalize concrete actions into *abstract actions*, according to [9], by allowing the probabilities to be interval-values instead of point-values, and the resulting worlds to be sets of states instead of states. This definition can be best illustrated by an example (see Figure 1.b). An abstract action is interpreted as a set of concrete actions each of which is obtained by *instantiation*, a process that replaces the probability intervals with consistent probability numbers, and sets of states with consistent states. The consistency condition for probability numbers means that they must be in the corresponding intervals, and obey the axioms of probability. The consistency conditions for states means that they must be in the corresponding sets. For example, the concrete action A in Figure 1.a is an instantiation of the abstract action $\mathbf{A}$ in Figure 1.b. We denote the instantiation relationship as $A \in \mathbf{A}$. The semantics of an abstract action $\mathbf{A}$ is that it is a function $\mathbf{A} : \mathcal{S} \rightarrow 2^{\mathcal{S}}$ that maps a probability distribution $P \in \mathcal{S}$ into a set of probability distributions $\{A(P) | A \in \mathbf{A}\}$.

An *abstract plan* $\mathbf{pl}$ is a sequence of actions, among which some actions may be abstract actions. An *instantiation plan* pl of an abstract plan $\mathbf{pl}$ is obtained by replacing each abstract action in $\mathbf{pl}$ with one of its instantiated actions. We denote this relationship as $pl \in \mathbf{pl}$. The projection of an abstract plan $\mathbf{pl}$ on an initial world w_{ini} is the process of determining the final world w_{final}, which is defined as: $w_{final} = \{pl(w_{ini}) | pl \in \mathbf{pl}\}$. The expected utility of a plan $\mathbf{pl}$ is defined as the $u(\mathbf{pl}(w_{ini})) = \{u(pl(w_{ini})) | pl \in \mathbf{pl}\}$.

These notions of abstract actions and plans are introduced in [9] to model sound abstraction, the central concept of the work of Haddawy et al [10, 4] on abstraction-based probabilistic planning. In this work, an abstract action $\mathbf{A}$ is called a *sound abstraction* of n concrete actions $A_1, \ldots, A_n$ if for any initial world $w_{ini} \in \mathcal{S}$, we have $A_i(w_{ini}) \in \mathbf{A}(w_{ini}), i = 1, \ldots, n$. For example, the abstract action $\mathbf{A}$ in Figure 1.b is a sound abstraction of the action A in Figure 1.a and an action A' such that $A'(s_1) = P'$, where $P'(s_2) = .3$ and $P'(s_3) = .7$. The use of abstraction allows us to evaluate an abstract plan in a single step, instead of a set of concrete plans in multiple steps, which can substantially reduce the time complexity of finding the optimal plan. For this purpose, evaluating an abstract plan means computing the minimum and the maximum of the set $u(\mathbf{pl}(w_{ini})) \subseteq R$, which are exactly the endpoints of the interval $\text{conv}(u(\mathbf{pl}(w_{ini})))$. Since u is a convex mapping, this interval is identical with the interval $u(\text{conv}(\mathbf{pl}(w_{ini})))$.

We now present two theorems. The first theorem, which is an abstract version of Theorem 1, says that abstract actions (and thus abstract plans) semi-commute with the cc-operator, while the second theorem shows that the first two projection rules in [9] are equally tight with respect to expected utility evaluation.

Theorem 2 *Let* $\mathbf{A}$ *be an abstract action or an abstract plan. Then* $\mathbf{A}(\sum_{i=1}^{n}{}^{cc}\Lambda_i w_i) \subseteq \sum_{i=1}^{n}{}^{cc}\Lambda_i \mathbf{A}(w_i)$*. As a consequence, if the initial world* w_{ini} *can be written as* $w_{ini} = \sum{}^{cc}\Lambda_i w_i \subseteq \mathcal{S}$*, then the expected utility of an abstract plan* $\mathbf{pl}$ *can be approximated as:*

$$\begin{aligned} u(\mathrm{conv}(\mathbf{pl}(w_{ini}))) &= u(\mathrm{conv}(\mathbf{pl}(\sum{}^{cc}\Lambda_i w_i))) \\ &\subseteq u(\mathrm{conv}(\sum{}^{cc}\Lambda_i \mathbf{pl}(w_i))) \\ &= \sum{}^{icc}\Lambda_i u(\mathrm{conv}(\mathbf{pl}(w_i))) \end{aligned}$$

Theorem 3 *Suppose* $w \subseteq \mathcal{S}$ *is a set of probability distributions over the state space* Ω*, and* $\mathbf{A}$ *is an abstract action or an abstract plan. Then* $\mathrm{conv}(\mathbf{A}(w)) = \mathrm{conv}(\mathbf{A}(\mathrm{conv}(w)))$*.*

4 Interval Bayesian Networks

Bayesian Networks [14] A Bayesian network is a pair consisting of a directed acyclic graph, and a collection of probability tables associated with the nodes. Each node of the graph represents a random variable, and its associated probability table specifies the conditional probability of that node given its parents. A Bayesian network represents a joint probability distribution over the random nodes that is obtained by multiplying the conditional probability tables. Figure 2.a depicts a 4-node Bayesian network N, where the conditional probabilities to be specified are $P(X), P(Y|X), P(Z|X)$, and $P(T|Y,Z)$ (for example, $P(y|x) = .4$, and $P(\bar{y}|x) = .6$). This Bayesian network represents the joint probability distribution P over $\{X,Y,Z,T\}$: $P(X,Y,Z,T) = P(X)P(Y|X)P(Z|X)P(T|Y,Z))$.

Interval Bayesian networks (IBN), studied by [7, 16, 5] are a generalization of Bayesian networks where we allow the probabilities to be interval-values. An IBN represents a set of Bayesian networks, each of which is obtained from the IBN by *instantiation*, a process that replaces each interval probability with a *consistent* point probability, where consistency means that the points must be in their corre-

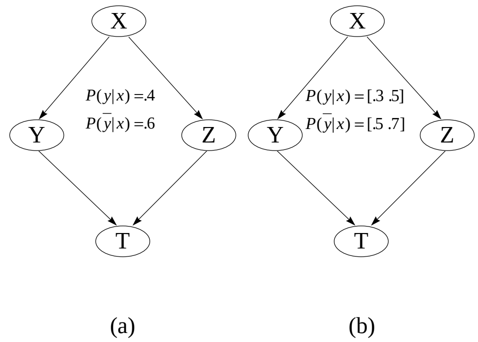

Figure 2: Examples: (a) a Bayesian network and (b) an interval Bayesian network.

sponding intervals, and obey the axioms of probability. Figure 2.b depicts a 4-node interval Bayesian network $\mathbf{N}$ where $\mathbf{P}(y|x) = [.3\ .5]$ and $\mathbf{P}(\bar{y}|x) = [.5\ .7]$. The regular Bayesian network in Figure 1.a (N) is an instantiation of $\mathbf{N}$. We denote the instantiation relationship by $N \in \mathbf{N}$.

Given such an IBN $\mathbf{N}$, the answer to any probabilistic query is defined to be an interval whose endpoints are the minimum and maximum of that query, ranging over all Bayesian network instantiations of $\mathbf{N}$. For example, in the IBN in Figure 1.b, we might be interested in computing $\mathbf{P}(t)$, which is defined as $\{P_N(t)|N \in \mathbf{N}\}$. Usually, we are only interested in computing the extreme point of $\mathbf{P}(t)$.

Efficient algorithms for probabilistic inference were developed in [14, 13]. Dechter [3] shows that most of these algorithms, as well as other probabilistic inference algorithms for finding the most probable explanation, the maximum a posteriori hypothesis, and the maximum expected utility, can be cast into a unifying algorithm schema called *bucket elimination*. In this section, we present an algorithm that is an instance of this schema and that computes the bounds for the probability of any node in an IBN. We assume that there is no evidence available; the case when there is evidence can be derived - albeit somewhat complicately - from this case, and is out of the scope of this paper.

The main idea of this algorithm can be best understood through an example. Suppose that in the IBN $\mathbf{N}$ in Figure 2.b, we are interested in computing the probability bounds for the random node T taking on some value t. Suppose that P is the joint distribution represented by a BN instantiation N of $\mathbf{N}$. Note that $P(t)$ can be written as $P(t) = \sum_{X,Y,Z} P(X,Y,Z,t) = \sum_{X,Y,Z} P(X)P(Y|X)P(Z|X)P(t|Y,Z)$.

Our goal is to factorize this last sum into a series of probability cross-product, which can be generalized into a series of cc-operator. For example, we have that:

$$\begin{aligned} P(t) &= \sum_{X,Y,Z} P(X)P(Y|X)P(Z|X)P(t|Y,Z) \\ &= \sum_X P(X) \sum_Y P(Y|X) \sum_Z P(Z|X)P(t|Y,Z). \end{aligned}$$

This factorization corresponds exactly to what we get by applying the bucket elimination algorithm to the ordering $\{X, Y, Z, T\}$ of the random nodes. From the fact that $P(X) \in \mathbf{P}(X), \sum_X P(X) = 1$, we can see that this formula can be generalized into a cc-operator sum. Now, given that X is fixed at some value x, we have that $P(Y|x) \in \mathbf{P}(Y|x), \sum_Y P(Y|x) = 1$, which means that we can generalized the inside sum into a cc-operator sum. It is obvious that all of the remaining sums in this formula can be generalized into cc-operator sums. Thus we have:

$$\begin{aligned} P(t) &= \sum_{X,Y,Z} P(X)P(Y|X)P(Z|X)P(t|Y,Z) \\ &= \sum_X P(X) \sum_Y P(Y|X) \sum_Z P(Z|X)P(t|Y,Z) \\ &\in \sum_X{}^{icc} \mathbf{P}(X) \sum_Y{}^{icc} \mathbf{P}(Y|X) \sum_Z{}^{icc} \mathbf{P}(Z|X)\mathbf{P}(t|Y,Z). \end{aligned}$$

References

[1] L. Chrisman. Propagation of 2-monotone lower probabilities on an undirected graph. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, pages 178–185, August 1996.

[2] F. Cozman. Robustness analysis of bayesian networks with local convex sets of distributions. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pages 108–115,

August 1997.

[3] R. Dechter. Bucket elimination: A unifying framework for probabilistic inference algorithms. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, pages 211–219, August 1996.

[4] A. Doan and P. Haddawy. Sound abstraction of probabilistic actions in the constraint mass assignment framework. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, pages 228–235, Portland, August 1996.

[5] D. Draper. *Localized Partial Evaluation of Bayesian Belief Networks.* PhD thesis, Computer Science Dept., Univ. of Washington, Seattle, November 1995.

[6] K.W. Fertig and J.S. Breese. Interval influence diagrams. In *Proceedings of the Fifth Workshop on Uncertainty in AI*, pages 102–111, Windsor, Ontario, 1989.

[7] K.W. Fertig and J.S. Breese. Probability intervals over influence diagrams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3):280–286, March 1993.

[8] B. Grunbaum. *Convex Polytopes.* Interscience, London, New York, 1967.

[9] V. Ha and P. Haddawy. Theoretical foundations for abstraction-based probabilistic planning. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, pages 291–298, Portland, August 1996.

[10] P. Haddawy, A. Doan, and R. Goodwin. Efficient decision-theoretic planning: Techniques and empirical analysis. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 229–236, Montreal, August 1995.

[11] N. Kushmerick, S. Hanks, and D. Weld. An algorithm for probabilistic least-commitment planning. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 1073–1078, Seattle, 1994.

[12] H. Kyburg and M. Pittarelli. Set-based bayesianism. *IEEE Transactions on Systems, Man, and Cybernetics*, 26:324–339, 1996.

[13] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society*, Series B50(2):157–224, 1988.

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

[15] T. Seidenfeld, M. Schervish, and J. B. Kadane. On the shared preferences of two bayesian decision makers. *Journal of Philosophy*, 86:225–244, 1989.

[16] B. Tessem. Interval probability propagation. *International Journal of Approximate Reasoning*, 7:95–120, 1992.