Graphical Belief Function Models: Theory, Computation, and Applications

Prakash P. Shenoy

University of Kansas, School of Business, Department of Analytics, Information, and Operations

October 28–November 1, 2023

Outline

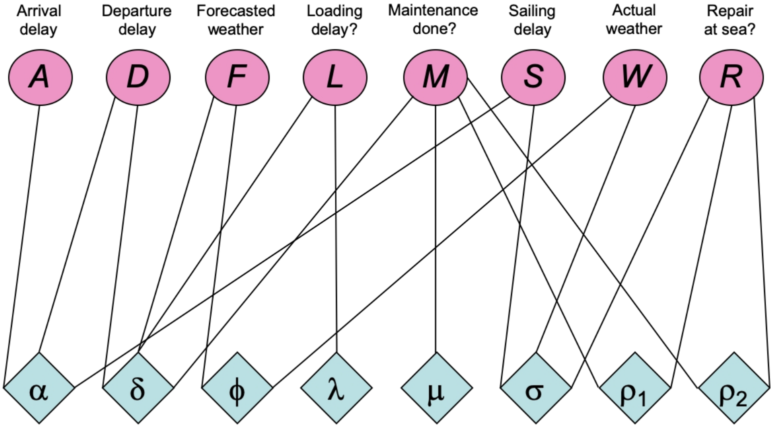
Outline

Valuation-based Systems

- A valuation-based System (VBS) is an abstract framework for representing and reasoning with knowledge.
- It has two parts. A static part that is concerned with representation of knowledge, and a dynamic part that is concerned with reasoning with knowledge.
- The static part consists of:
  - Variables: A finite set $\mathcal{V}$ of variables $\{X, Y, Z, \ldots\}$. Subsets of $\mathcal{V}$ will be denoted by $r, s, t, \ldots$.
  - Valuations: A finite set Ψ of valuations $\{\rho, \sigma, \tau, \ldots\}$. Each valuation encodes knowledge about a subset of variables. Thus, we say, ρ is a valuation for r, where $r \subseteq \mathcal{V}$.
- A graphical representation of a VBS is called a valuation network.

Valuation-based Systems

- The valuation network for the Captain's problem: A bipartite graph with variables and valuations as nodes. Each valuation is linked to the variables in its domain.



Valuation-based Systems

- The dynamic part consists of several operators:
  - Combination: $\oplus : \Psi \times \Psi \to \Psi$ that enables us to aggregate knowledge.
  - The combination operator has the following properties:
    - (Domain) If ρ is a valuation for r, σ is a valuation for s, and ρ and σ are distinct, then $\rho \oplus \sigma$ is a valuation for $r \cup s$
    - (Commutativity) $\rho \oplus \sigma = \sigma \oplus \rho$
    - (Associativity) $\rho \oplus (\sigma \oplus \tau) = (\rho \oplus \sigma) \oplus \tau$
  - In general $\rho \oplus \rho \neq \rho$. One must be careful not to double-count non-idempotent knowledge. Hence, one should only combine distinct valuations.
  - The sequence in which knowledge is aggregated should make no difference.
  - The combination of all valuations, $\oplus \Psi$, is called the joint valuation.
  - In large models ($|\mathcal{V}|$ is large), the joint cannot be explicitly computed.

Valuation-based Systems

- Another operator is marginalization
- Marginalization: $-X : \Psi \to \Psi$ that allows us to coarsen knowledge marginalizing X out of the domain of a valuation.
- Properties of Marginalization:
  - (Domain) If ρ is a valuation for r, and $X \in r$, then ρ^{-X} is a valuation for $r \setminus \{X\}$.
  - (Order does not matter) If ρ is a valuation for r, $X, Y \in r$, then $(\rho^{-X})^{-Y} = (\rho^{-Y})^{-X} = \rho^{-\{X,Y\}}$.
  - (Local computation) If ρ and σ are valuations for r and s, respectively, $X \in r$, and $X \notin s$, then $(\rho \oplus \sigma)^{-X} = (\rho^{-X}) \oplus \sigma$.
- We will sometimes denote $\rho^{-\{X\}}$ by $\rho^{\downarrow r \setminus \{X\}}$.

Valuation-based Systems

- Making inference means finding marginals of the joint valuation for the variables of interest
- Thus, if X is a variable of interest, we compute $(\oplus\Psi)^{\downarrow X} = (\oplus\Psi)^{-(\mathcal{V}\setminus\{X\})}$ by marginalizing all variables in $\mathcal{V} \setminus \{X\}$ out of the joint valuation $\oplus\Psi$.

Valuation-based Systems

- VBS is an abstraction of several uncertainty calculi
  - propositional calculus
  - probability theory
  - belief function theory
  - Spohn's epistemic belief calculus
  - possibility theory
  - ...
- VBS can also be considered as an abstraction of
  - Optimization
  - Bayesian decision theory
  - Solving systems of equations
  - Relational database theory
  - ...

Outline

Basics of D-S Theory: Static and Dynamic

- Static: We represent knowledge using either:
  - basic probability assignment (BPA) m,
  - belief function Bel,
  - plausibility function Pl,
  - commonality function Q.
- Dynamic: We make inferences using:
  - Dempster's rule of combination
  - Marginalization rule
- Inference: Given a set of belief functions (BPA, plausibility, belief, or commonality) representing knowledge of the domain and all evidence, we would like to find the marginals of the joint for some variables of interest.
- The joint belief function is obtained by combining all belief functions using Dempster's rule of combination.

Basics of D-S Theory: Static and Dynamic

- Suppose $\mathcal{V}$ denotes a finite set of variables
- For each $X \in \mathcal{V}$, Ω_X denotes a finite set of states of X
- For every non-empty subset $s \subseteq \mathcal{V}$,

$$\Omega_s = \times_{X \in s} \Omega_X$$

  denotes the states of s
- Let 2^{Ω_s} denote the set of all subsets of Ω_s
- A basic probability assignment (BPA) m for s is a function $m : 2^{\Omega_s} \to [0, 1]$ such that:

$$m(\emptyset) = 0, \quad (1)$$

$$\sum_{\mathsf{a} \in 2^{\Omega_s}} m(\mathsf{a}) = 1 \quad (2)$$

- Subsets $\mathsf{a} \in 2^{\Omega_s}$ such that $m(\mathsf{a}) > 0$ are called focal elements of m.

Basics of D-S Theory: Static and Dynamic

- The combination rule in D-S theory of belief functions is Dempster's rule, which Dempster called the "product-intersection" rule.
- The product of the BPA masses is assigned to the intersection of the focal elements, any mass assigned to the empty set is discarded, and the remaining masses are re-normalized.
- In general $m \oplus m \neq m$.
- Thus, in combining, e.g., m_1 and m_2 by Dempster's rule, it is important that m_1 and m_2 are distinct pieces of evidence, and there is no double-counting of uncertain knowledge.

Basics of D-S Theory: Static and Dynamic

Dempster's rule satisfies all properties of the combination operator.

- (Domain) If m_1 is a BPA for s_1 and m_2 is a BPA for s_2, then $m_1 \oplus m_2$ is a BPA for $s_1 \cup s_2$
- (Commutativity) $m_1 \oplus m_2 = m_2 \oplus m_1$
- (Associativity) $m_1 \oplus (m_2 \oplus m_3) = (m_1 \oplus m_2) \oplus m_3$

Basics of D-S Theory: Static and Dynamic

- Marginalization in belief function theory is addition.
- Projection of states: If $\mathsf{a} \in \Omega_s$, and $X \in s$, then $\mathsf{a}^{\downarrow s \setminus \{X\}}$ (or a^{-X}) is the state of $s \setminus \{X\}$ obtained from a by dropping the state of X. For example, $(x, y)^{\downarrow X} = x$.
- Projection of subset of states: If $\mathsf{a} \subseteq \Omega_s$, then a^{-X} (or $\mathsf{a}^{\downarrow s \setminus \{X\}}$) is

$$\mathsf{a}^{-X} = \{\mathsf{b}^{-X} \,:\, \mathsf{b} \in \mathsf{a}\}$$

- If m is a BPA for s, and $X \in s$, then m^{-X} is a BPA for $s \setminus \{X\}$ defined as follows:

$$m^{-X}(\mathsf{a}) = \sum_{\mathsf{b} \in 2^{\Omega_s} \,:\, \mathsf{b}^{-X} = \mathsf{a}} m(\mathsf{b})$$

for all $\mathsf{a} \in 2^{s \setminus \{X\}}$.

Basics of D-S Theory: Static and Dynamic

An Example of Marginalization:

- Suppose M and R are variables with $\Omega_M = \{t_M, f_M\}$ and $\Omega_R = \{t_R, f_R\}$,
- Suppose m is a BPA for $\{M, R\}$ such that

$$\begin{aligned} m(\{(t_M, t_R), (f_M, t_R), (f_M, f_R)\}) &= 0.1, \\ m(\{(t_M, f_R), (f_M, t_R), (f_M, f_R)\}) &= 0.7, \\ m(\Omega_{\{M,R\}}) &= 0.2. \end{aligned}$$

- Then m^{-M} is a BPA for $\{R\}$ such that:

$$\rho^{-M}(\{t_R, f_R\}) = 1.$$

- And m^{-R} is a BPA for $\{M\}$ such that:

$$m^{-R}(\{t_M, f_M\}) = 1.$$

- Thus, m by itself tells us nothing about M or R.

Basics of D-S Theory: Static and Dynamic

- The definition of marginalization of BPAs satisfies the properties of marginalization:
  - (Domain) If m is a BPA for r, and $X \in r$, then m^{-X} is a BPA for $r \setminus \{X\}$.
  - (Order does not matter) If m is a BPA for r, $X, Y \in r$, then $(\rho^{-X})^{-Y} = (\rho^{-Y})^{-X} = \rho^{-\{X,Y\}} = \rho^{\downarrow r \setminus \{X,Y\}}$.
  - (Local computation) If m_r and m_s are BPAs for r and s, respectively, $X \in r$, and $X \notin s$, then $(m_r \oplus m_s)^{-X} = (m_r^{-X}) \oplus m_s$.

Basics of D-S Theory: Conditional Independence

In probability theory, conditional independence is defined as follows:

Definition (CI in probability theory (Dawid 1979))

Suppose P is a joint probability distribution for $\mathcal{V}$, and suppose r, s, and t are disjoint subsets of $\mathcal{V}$. We say r and s are conditionally independent given t with respect to P, written as $r \perp\!\!\!\perp_P s \mid t$, iff

$$P^{\downarrow r \cup s \cup t} = P_{r \cup t} \otimes P_{s \cup t}, \tag{3}$$

where $P_{r \cup t}$ is a probability potential for $r \cup t$, $P_{s \cup t}$ is a probability potential for $s \cup t$, $P_{s \cup t}$ and $P_{s \cup t}$ are distinct, and $\otimes$ is the probabilistic combination operator (pointwise multiplication followed by normalization).

Basics of D-S Theory: Conditional Independence

The definition of CI in the D-S theory is similar:

Definition (CI in D-S theory (Shenoy 1994))

Suppose m is a BPA for $\mathcal{V}$, and suppose r, s, and t are disjoint subsets of $\mathcal{V}$. We say r and s are conditionally independent given t with respect to m, written as $r \perp\!\!\!\perp_m s \mid t$, iff

$$m^{\downarrow r \cup s \cup t} = m_{r \cup t} \oplus m_{s \cup t}, \tag{4}$$

where $m_{r \cup t}$ is a BPA for $r \cup t$, $m_{s \cup t}$ is a BPA for $s \cup t$, and $m_{r \cup t}$ and $m_{s \cup t}$ are distinct.

- There are several definitions of conditional independence in the D-S theory.
- The definition above is key to defining graphical models in the D-S theory.
- Like in probability theory, the definition above satisfies the graphoid properties of conditional independence (Pearl and Paz 1987).

Basics of D-S Theory: Conditional BPAs

Conditional BPAs

- In probability theory, we construct Bayesian network models by using conditional probability tables (CPTs) for each variable in the network.
- What is an analog of CPT in the belief function theory?

Definition (Conditional BPA (Jiroušek et al 2023))

Suppose r and s are disjoint subsets of variables and $r' \subseteq r$. BPA $m_{r'|s}$ for $r' \cup s$ is a conditional BPA for r' given s if and only if

1. $(m_{r'|s})^{\downarrow s}$ is a vacuous BPA for s, and
2. If m_r is a BPA for r, and m_r and $m_{r'|s}$ are distinct, then $m_r \oplus m_{r'|s}$ is a BPA for $r \cup s$.

- r' is called the head of the conditional $m_{r'|s}$, and s its tail.
- In the second condition, m_r and $m_{r'|s}$ are distinct iff $s \perp\!\!\!\perp_{m_r \oplus m_{r'|s}} (r \setminus r') \mid r'$ [Jiroušek et al. 2023].

Basics of D-S Theory: Conditional BPAs

Where do conditionals come from?

- One source of conditionals is Smets' conditional embedding.
- Suppose that when $X = x$, there is a BPA for Y, denoted by m_{Y_x}, that represents our conditional knowledge of Y in the context $X = x$.
- The knowledge of Y encoded in m_{Y_x} is valid only in the case $X = x$, but m_{Y_x} is not a conditional BPA.
- Using Smets' conditional embedding, we convert the conditional BPA m_{Y_x} for Y to an conditional BPA $m_{Y|x}$ for (X, Y) as follows: Each focal element b of m_{Y_x} is converted to a corresponding focal element $(\{x\} \times \mathsf{b}) \cup ((\Omega_X \setminus \{x\}) \times \Omega_Y)$ of $m_{Y|x}$ with the same mass, i.e.,

$$m_{Y|x}((\{x\} \times \mathsf{b}) \cup ((\Omega_X \setminus \{x\}) \times \Omega_Y)) = m_{Y_x}(\mathsf{b}).$$

Conditional BPA $m_{Y|x}$ for (X, Y) has the following properties:

1. $m_{Y|x}$ is a conditional for Y given x, i.e., $(m_{Y|x})^{\downarrow X}$ is a vacuous BPA for X.
2. Suppose $m_{X=x}$ is a deterministic BPA for X as follows: $m_{X=x}(\{x\}) = 1$. Then, $(m_{Y|x} \oplus m_{X=x})^{\downarrow Y} = m_{Y_x}$.

Basics of D-S Theory: Conditional BPAs

An Example

- Suppose X and Y are variables with $\Omega_X = \{x, \bar{x}\}$ and $\Omega_Y = \{y, \bar{y}\}$.
- Suppose m_{Y_x} is a BPA for Y as follows:

$$\begin{aligned} m_{Y_x}(\{y\}) &= 0.8, \\ m_{Y_x}(\Omega_Y) &= 0.2. \end{aligned}$$

- m_{Y_x} is conditional knowledge of Y in the context $X = x$, but it is not a conditional BPA.
- Conditional BPA $m_{Y|x}$ for $\{X, Y\}$ is as follows:

$$\begin{aligned} m_{Y|x}(\{(x, y), (\bar{x}, y), (\bar{x}, \bar{y})\}) &= 0.8, \\ m_{Y|x}(\Omega_{X,Y}) &= 0.2. \end{aligned}$$

Basics of D-S Theory: Conditional BPAs

- Consider $m_{Y|x}$:

$$
\begin{aligned}
m_{Y|x}(\{(x,y),(\bar{x},y),(\bar{x},\bar{y})\}) &= 0.8, \\
m_{Y|x}(\Omega_{X,Y}) &= 0.2.
\end{aligned}
$$

- It is clear that $(m_{Y|x})^{\downarrow X}$ is vacuous for X.
- Consider $m_{Y|x} \oplus m_{X=x}$:

$m_{Y\|x} \oplus m_{X=x}$	$\{(x,y),(\bar{x},y),(\bar{x},\bar{y})\}$ 0.8	$\Omega_{\{X,Y\}}$ 0.2
$\{(x,y),(x,\bar{y})\}$ 1	$\{(x,y)\}$ 0.8	$\{(x,y),(x,\bar{y})\}$ 0.2

- Thus, $(m_{Y|x} \oplus m_{X=x})(\{(x,y)\}) = 0.8$, $(m_{Y|x} \oplus m_{X=x})(\{(x,y),(x,\bar{y})\}) = 0.2$, and
- $(m_{Y|x} \oplus m_{X=x})^{\downarrow Y} = m_{Y_x}$.

Basics of D-S Theory: BF Directed Graphical Models

Notation:

- A directed graph G is a pair $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{X_1, \ldots, X_n\}$ denotes the set of nodes and $\mathcal{E}$ denotes the set of directed edges (X_i, X_j) between two distinct variables in $\mathcal{V}$.
- For any node $X \in \mathcal{V}$, let $Pa_G(X)$ denote the parents of X in G, i..e,

$$Pa_G(X) = \{Y \in \mathcal{V} : (Y, X) \in \mathcal{E}\}.$$

- A directed graph is said to be acyclic if and only if there exists a sequence of the nodes of the graph, say $(X_1, \ldots, X_n)$ such that if there is a directed edge $(X_i, X_j) \in \mathcal{E}$ then X_i must precede X_j in the sequence.
- The sequence $(X_1, \ldots, X_n)$ is called a topological sequence as it depends only on the structure of the directed graph. It may not be unique.

Basics of D-S Theory: BF Directed Graphical Models

Definition (BFDGM)

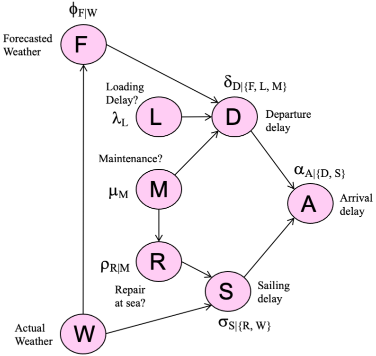
Suppose we have a directed acyclic graph $G = (\mathcal{V}, \mathcal{E})$ with n nodes in $\mathcal{V}$. A belief-function directed graphical model $(BFDGM)$ is a pair $(G, \{m_1, \ldots, m_n\})$ such that BPA m_i associated with node X_i is a conditional BPA for X_i given $Pa_G(X_i)$, for $i = 1, \ldots, n$. A fundamental assumption of a BFDGM is that $m_1, \ldots, m_n$ are distinct, and the joint BPA m for $\mathcal{V}$ associated with the model is given by

$$m = \bigoplus_{i=1}^{n} m_i. \tag{5}$$

- The assumption in Def. (BFDGM) that all conditionals are distinct allows the combination in Eq. (5).
- Given m, the joint BPA for $\mathcal{V}$ as defined in Eq. (5) follows from the Definition the following CI relations hold. Suppose $(X_1, \ldots, X_n)$ is a topological sequence associated with BFDGM $(G, \{m_1, \ldots, m_n\})$. Then for each X_i, $i = 2, \ldots, n$,

$$X_i \perp\!\!\!\perp_m (\{X_1, \ldots X_{i-1}\} \setminus Pa_G(X_i)) \mid Pa_G(X_i).$$

Basics of D-S Theory: BF Directed Graphical Models



- A topological sequence is: (W, L, M, F, D, R, S, A).
- Some CI assumptions: Let m denote the joint BPA associated with the model.
  - $A \perp\!\!\!\perp_m \{W, L, M, F, R\} \mid \{D, S\}$
  - $S \perp\!\!\!\perp_m \{L, M, F, D\} \mid \{W, R\}$, etc.

Basics of D-S Theory: Undirected Graphical Models

Notation:

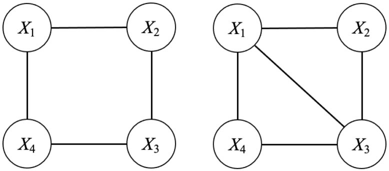
- An undirected graph G is a pair $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{X_1, \ldots, X_n\}$ denotes the set of nodes, and $\mathcal{E}$ denotes the set of undirected edges $\{X_i, X_j\}$ between two distinct nodes in $\mathcal{V}$.
- Consider node $X_i \in \mathcal{V}$. The Markov boundary of X_i in G, denoted by $Ma_G(X_i)$, is as follows:

$$Ma_G(X_i) = \{X_j \in \mathcal{V} : \{X_i, X_j\} \in \mathcal{E}\}.$$

  In other words, the Markov boundary of X_i is the set of nodes in $\mathcal{V}$ that are directly connected to X_i by an edge.
- A clique in G is a maximal completely connected subset of nodes in $\mathcal{V}$.
- We assume there are k cliques $r_1, \ldots, r_k$ in G, where $r \leq |\mathcal{E}|$.

Basics of D-S Theory: BF Undirected Graphical Models

Two examples:



For the UG model on the left:

- There are four nodes, four edges, and four cliques: the nodes in each edge is a clique.
- $Ma_G(X_1) = \{X_2, X_4\}$, $Ma_G(X_2) = \{X_1, X_3\}$, etc.

For the UG model on the right:

- There are four nodes, five edges, and two cliques: $\{X_1, X_2, X_3\}$, and $\{X_1, X_3, X_4\}$.
- $Ma_G(X_1) = \{X_2, X_3, X_4\}$, $Ma_G(X_2) = \{X_1, X_3\}$, etc.

Basics of D-S Theory: BF Undirected Graphical Models

Definition (BFUGM)

Suppose we have an undirected graph $G = (\mathcal{V}, \mathcal{E})$ with r cliques $r_1, \ldots, r_k$. A belief-function undirected graphical model (BFUGM) is a pair $(G, \{m_1, \ldots, m_k\})$ such that BPA m_i associated with clique r_i is a BPA for r_i. A fundamental assumption of a BFUGM is that $m_1, \ldots, m_k$ are distinct, and the joint BPA m for $\mathcal{V}$ associated with the model is given by

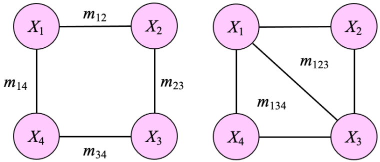
$$m = \bigoplus_{i=1}^{k} m_i \qquad (6)$$

- The assumption in Def. (BFUGM) that all BPAs are distinct allows the combination in Eq. (6).
- Given m, the joint BPA for $\mathcal{V}$ in Eq. (6), it follows from the definition that the following CI relations hold: For each $X_i \in \mathcal{V}$,

$$X_i \perp\!\!\!\perp_m \mathcal{V} \setminus (Ma_G(X_i) \cup \{X_i\}) \mid Ma_G(X_i)$$

Basics of D-S Theory: BF Undirected Graphical Model

Two examples:



For the UG model on the left: Let m denote $m_{12} \oplus m_{23} \oplus m_{34} \oplus m_{14}$. There are two CI assumptions:

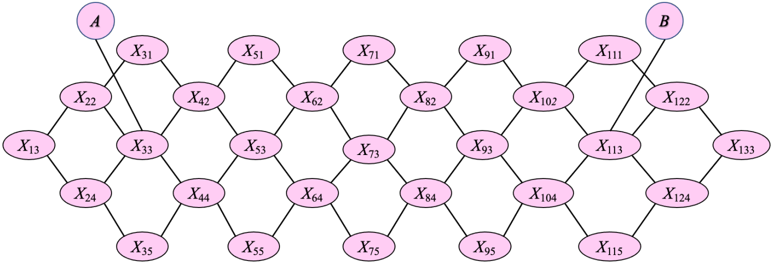
- $X_1 \perp\!\!\!\perp_m X_3 \mid \{X_2, X_4\}$. This follows from $m = (m_{12} \oplus m_{14}) \oplus (m_{23} \oplus m_{34})$.
- $X_2 \perp\!\!\!\perp_m X_4 \mid \{X_1, X_3\}$. This follows from $m = (m_{12} \oplus m_{23}) \oplus (m_{14} \oplus m_{34})$.

For the UG model on the right: Let m denote $m_{123} \oplus m_{134}$. There is one CI assumption:

- $X_2 \perp\!\!\!\perp_m X_4 \mid \{X_1, X_3\}$. This follows from $m = m_{123} \oplus m_{134}$.

Basics of D-S Theory: BF Undirected Graphical Models

An example of a BFUGM:



- We have a grid of $5+6+7+6+5=29$ communication nodes. There are 44 undirected edges in the graph. The reliability of each link is 90%.
- Nodes A and B are connected to the grid with links of reliability 80%.
- What is the reliability of the connection between A and B?

Outline

Local Computation: Captain's Problem

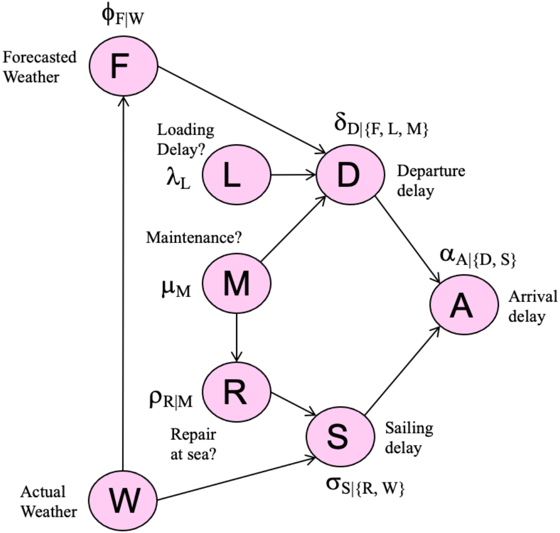
- Captain's Problem (R. Almond, *Graphical Belief Modeling*, Chapman and Hall, 1995)
  - A ship's captain is concerned about how many days his ship may be delayed before arrival at a destination.
  - The arrival delay may be a result of a delay in departure and delay in sailing.
  - Delay in departure may be a result of maintenance (at most one day), delay in loading (at most one day), or due to forecast of bad weather (at most one day).
  - Delay in sailing may result from bad weather (at most one day) and whether repairs may be needed at sea (at most one day).
  - If maintenance is done before sailing, the chance of repairs at sea is less likely.
  - Weather forecast says small chance of bad weather (.2), good chance of good weather (0.6). The forecast is 80% reliable.
  - Captain has some knowledge of loading delay and whether maintenance is done before departure.

Local Computation: Captain's Problem

- Variables
  - A (arrival delay), $\Omega_A = \{0, 1, 2, 3, 4, 5\}$.
  - D (departure delay), $\Omega_D = \{0, 1, 2, 3\}$.
  - S (sailing delay), $\Omega_S = \{0, 1, 2\}$.
  - L (is loading delayed?), $\Omega_L = \{t, f\}$.
  - F (weather forecast), $\Omega_F = \{b, g\}$.
  - W (actual weather), $\Omega_W = \{b, g\}$.
  - M (is maintenance done before sailing?), $\Omega_M = \{t, f\}$.
  - R (is a repair at sea needed?), $\Omega_R = \{t, f\}$.

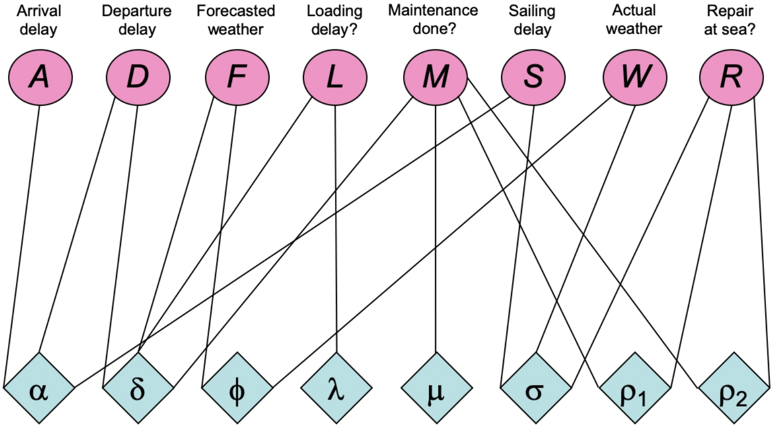
Local Computation: Captain's Problem

- The Captain problem can be described by a causal directed acyclic graph (DAG) as follows:



Local Computation: Captain's Problem

- Valuation Network: A bipartite graph with variables and valuations as nodes. Each valuation is linked to the variables in its domain.



Local Computation: Captain's Problem

- Consider the piece of knowledge: Arrival delay is the sum of departure delay and sailing delay
- We model this piece of knowledge by a BPA α for $\{A, D, S\}$ such that

$$\alpha(\{(0,0,0),(1,1,0),(2,2,0),(3,3,0),\\ (1,0,1),(2,1,1),(3,2,1),(4,3,1),\\ (2,0,2),(3,1,2),(4,2,2),(5,3,2)\}) = 1.$$

- α has one focal set. Such BPAs are called deterministic.
- α can be considered as a conditional for A given $\{D, S\}$ because $\alpha^{\downarrow\{D,S\}}$ is a vacuous BPA for $\{D, S\}$.

Local Computation: Captain's Problem

- Loading delay, bad weather forecast, and maintenance each add one day to departure delay
- We model this piece of knowledge by a deterministic BPA δ for $\{D, L, F, M\}$ such that

$$\delta(\{(0, f, g, f), (1, t, g, f), (1, f, b, f), (1, f, g, t), (2, f, b, t), (2, t, g, t), (2, t, g, f), (3, t, b, t)\}) = 1.$$

- δ can be considered as a conditional BPA for D given $\{L, F, M\}$ because $\delta^{\downarrow\{L,F,M\}}$ is a vacuous BPA for $\{L, F, M\}$.

Local Computation: Captain's Problem

- At least 90% of the time, bad weather and repair at sea each add one day to the sailing delay
- We model this by BPA σ for $\{S, W, R\}$ such that

$$\begin{aligned} \sigma(\{(0,g,f),(1,b,f),(1,g,t),(2,b,t)\}) &= 0.9, \\ \sigma(\Omega_{\{S,A,R\}}) &= 0.1 \end{aligned}$$

- σ can be considered as a conditional for S given $\{W, R\}$ because $\sigma^{\downarrow\{W,R\}}$ is a vacuous BPA for $\{W, R\}$.

Local Computation: Captain's Problem

- Forecast is 80% reliable
- This piece of knowledge is represented by BPA ϕ_1 for $\{F, W\}$ such that

$$\begin{aligned}\phi_1(\{(b,b),(g,g)\}) &= 0.8,\\ \phi_1(\Omega_{\{F,W\}}) &= 0.2.\end{aligned}$$

- ϕ_1 can be regarded as a conditional for F given W because $\phi_1^{\downarrow W}$ is a vacuous BPA for W.
- Forecast predicts bad weather with a chance of 0.2 and good weather with a chance of 0.6.
- This piece of knowledge is represented by BPA ϕ_2 for $\{F\}$ such that

$$\begin{aligned}\phi_2(\{b\}) &= 0.2,\\ \phi_2(\{g\}) &= 0.6,\\ \phi_2(\Omega_{\{F\}}) &= 0.2.\end{aligned}$$

- ϕ_2 is not a conditional BPA. It represents evidence for F.

Local Computation: Captain's Problem

- Loading is delayed with chance 0.3 and on schedule with chance 0.5.
- This piece is model by BPA λ for $\{L\}$ such that

$$\begin{aligned}\lambda(\{t\}) &= 0.3,\\ \lambda(\{f\}) &= 0.5,\\ \lambda(\Omega_{\{L\}}) &= 0.2.\end{aligned}$$

- No maintenance was done on the ship before departure
- This piece of knowledge is represented by BPA μ for $\{M\}$ such that

$$\mu(\{f\}) = 1.$$

- λ and μ are pieces of evidence for L and M, respectively.

Local Computation: Captain's Problem

- If maintenance was done before sailing, then the chance of repair at sea is between 10 and 30%. This is conditional knowledge denoted by BPA $\rho_{R_{M=t}}$ for R as follows:

$$\begin{aligned} \rho_{R_{M=t}}(\{t\}) &= 0.1, \\ \rho_{R_{M=t}}(\{f\}) &= 0.7, \\ \rho_{R_{M=t}}(\{t, f\}) &= 0.2. \end{aligned}$$

- After conditional embedding, ρ_1 is a conditional BPA for R given M as follows:

$$\begin{aligned} \rho_1(\{(t,t),(f,t),(f,f)\}) &= 0.1, \\ \rho_1(\{(t,f),(f,t),(f,f)\}) &= 0.7, \\ \rho_1(\Omega_{\{M,R\}}) &= 0.2. \end{aligned}$$

- Notice that $\rho_1^{\downarrow M}$ is a vacuous BPA for M.

Local Computation: Captain's Problem

- If maintenance was not done before sailing, then the chance of repair at sea is between 20 and 80%. This is conditional knowledge denoted by BPA $\rho_{R_{M=f}}$ for R as follows:

$$\begin{aligned} \rho_{R_{M=f}}(\{t\}) &= 0.2, \\ \rho_{R_{M=f}}(\{f\}) &= 0.2, \\ \rho_{R_{M=f}}(\{t, f\}) &= 0.6. \end{aligned}$$

- After conditional embedding, ρ_2 is a conditional BPA for R given M as follows:

$$\begin{aligned} \rho_2(\{(f,t),(t,t),(t,f)\}) &= 0.2, \\ \rho_2(\{(f,f),(t,t),(t,f)\}) &= 0.2, \\ \rho_2(\Omega_{\{M,R\}}) &= 0.6. \end{aligned}$$

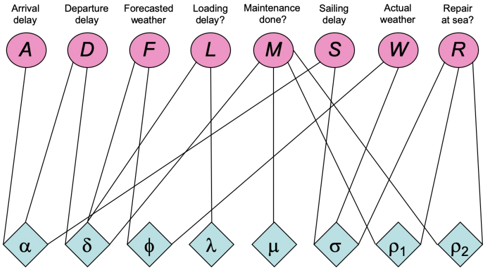
- Notice that $\rho_2^{\downarrow M}$ is a vacuous BPA for M.

Local Computation

- Making inference means finding marginals of the joint valuation $\oplus\Psi$ for the variables of interest.
- If there are many variables in $\mathcal{V}$, computing the joint valuation $\oplus\Psi$ for $\mathcal{V}$ is intractable.
- However, one can compute the marginal of the joint for X, $(\oplus\Psi)^{\downarrow X}$, without computing the joint explicitly, using so-called local computation.
- The axiom that allows local computation is the local computation axiom:
  If ρ and σ are BPAs for r and s, respectively, $X \in r$, and $X \notin s$, then $(\rho \oplus \sigma)^{-X} = (\rho^{-X}) \oplus \sigma$.

Local Computation

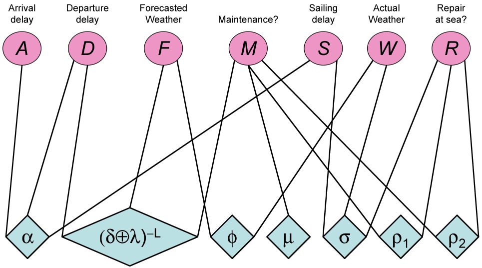
- Consider the Captain's problem. We want to compute the marginal of the joint for A. So, we have to marginalize all other variables from the joint.



- Consider L. It is only in the domain of δ and λ. The local computation axiom guarantees that if we replace δ and λ by $(\delta \oplus \lambda)^{-L}$, then the product of all valuations will give us $(\oplus \Psi)^{-L}$.

Local Computation

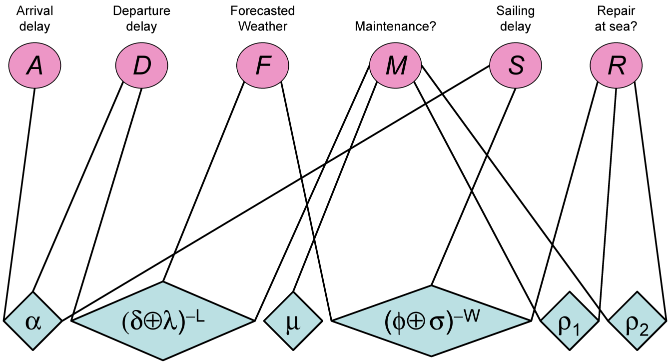
- The reduced VN is as follows:



- We can recursively remove all but A from the VN.
- Consider W. It is in the domain of ϕ for $\{F, W\}$ and σ for $\{S, W, R\}$. Thus, $(\phi \oplus \sigma)^{-W}$ will be a BPA for $\{F, S, R\}$.

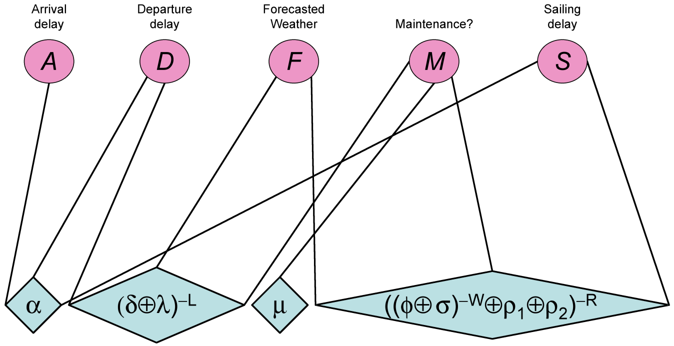
Local Computation

- After deletion of $\{L, W\}$:



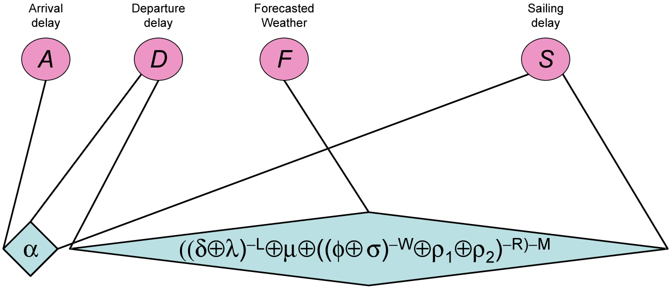
Local Computation

- After deletion of $\{L, W, R\}$:



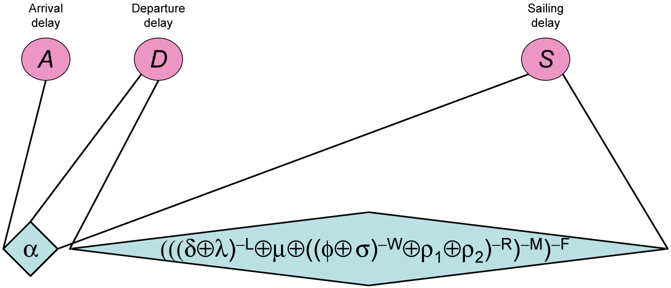
Local Computation

- After deletion of $\{L, W, R, M\}$:



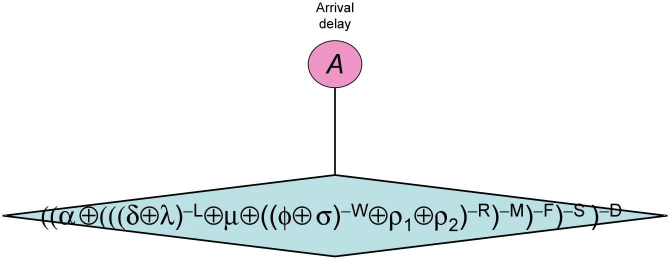
Local Computation

- After deletion of $\{L, W, R, M, F\}$:



Local Computation

- After deletion of $\{L, W, R, M, F, S, D\}$ in this order:



- If we combine all valuations, we have the marginal of the joint for A.

Local Computation

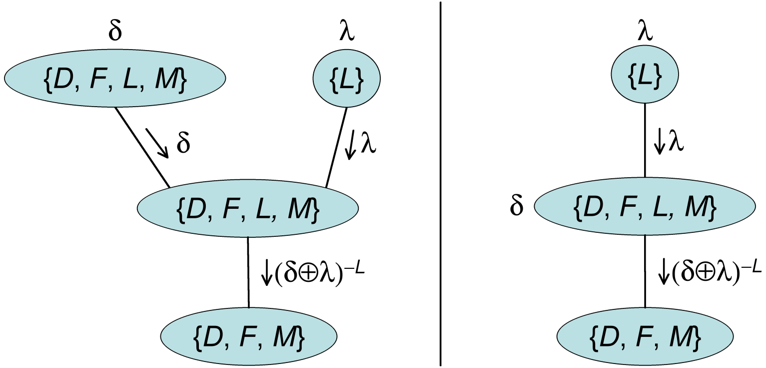
- In finding the marginal for A, we used deletion sequence $L\,W\,R\,M\,F\,S\,D$.
- The order does not matter axiom allows us to use any deletion sequence (and obtain the same marginal).
- Some deletion sequences involve less computation than others.
- Finding an optimal deletion sequence is a hard problem.
- So we use heuristics to select a sequence such as one-step-look-ahead: The variable to be marginalized next is the one that leads to a combination on the smallest domain.
- A local computation algorithm for finding marginals is implemented in Belief Function Machine.

Local Computation as Message Passing

- If we can find the marginal for A, we can find the marginal for any other variable similarly.
- However, there may be duplication of computations.
- We can avoid duplication by saving intermediate computations in a data structure called a "join tree."
- A join tree is a tree with subsets of variables as nodes with the property that if a variable appears in two different nodes, it appears in all nodes in the path between them.

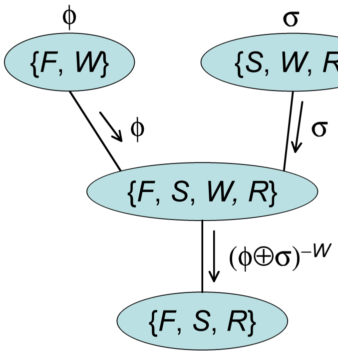
Local Computation as Message Passing

- Consider deletion of L. We can describe the computation as messages between nodes as follows:



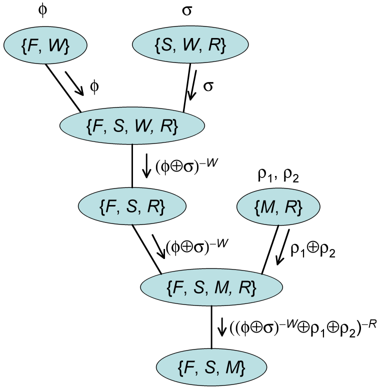
Local Computation as Message Passing

- Similarly, deletion of W can be described as follows:



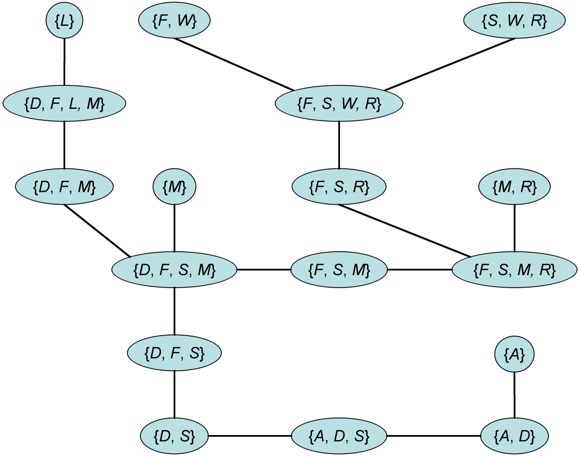
Local Computation as Message Passing

- Similarly, deletion of R can be described as follows:



Local Computation as Message Passing

- The tree thus constructed is a join tree.

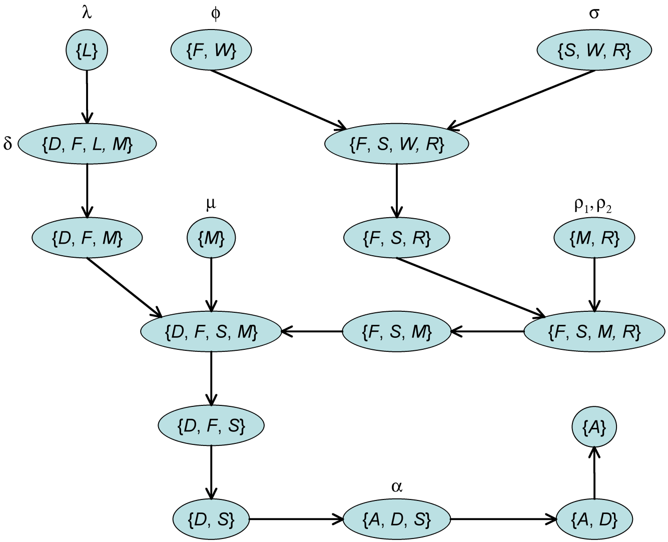


Local Computation as Message Passing

- To find the marginal for a variable, say A, orient all edges toward A, which is now the tree's root.
- Each node sends a message to its inward neighbor, which is the combination of all messages it receives from its outward neighbors plus what it has suitably marginalized.
- Timing: Leaves (nodes with no outward neighbors) can send messages immediately. Non-leaves have to wait till they receive a message from all their outward neighbors.
- Process is finished when the root has received a message from its outward neighbors. The root then combines all the messages it receives plus what it has.

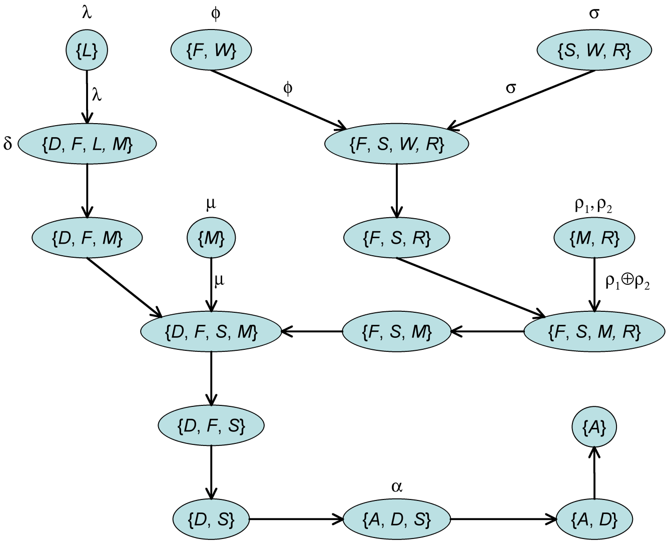
Local Computation as Message Passing

- At the beginning:



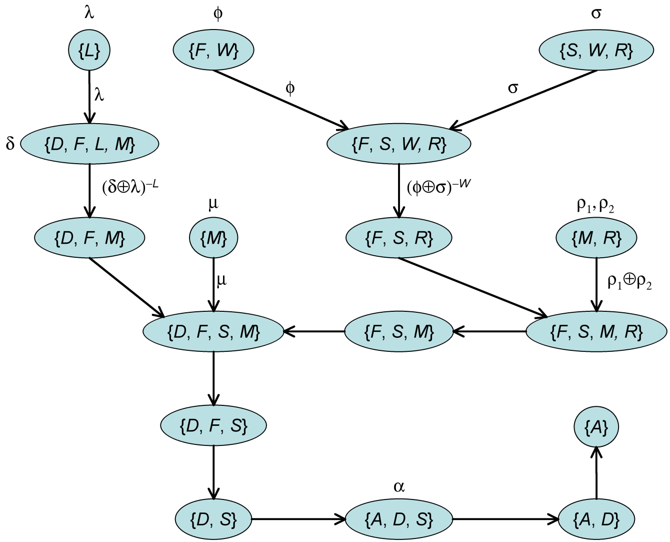
Local Computation as Message Passing

- Time step 1:



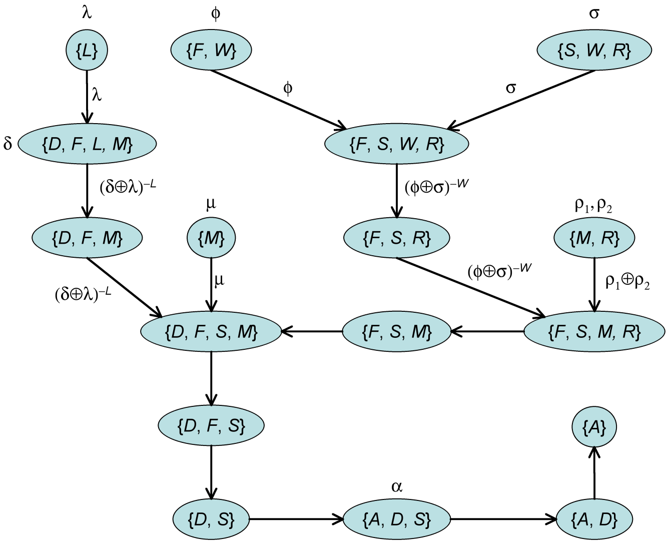
Local Computation as Message Passing

- Time step 2:



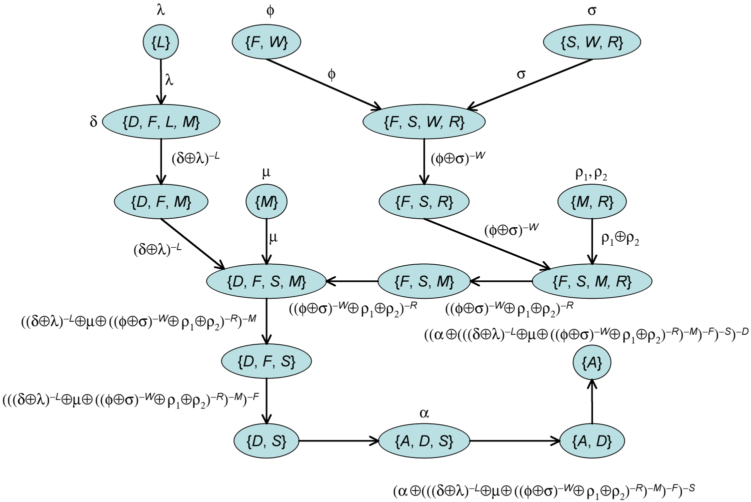
Local Computation as Message Passing

- Time step 3:



Local Computation

- Computing a marginal can be described as the propagation of messages in a join tree:

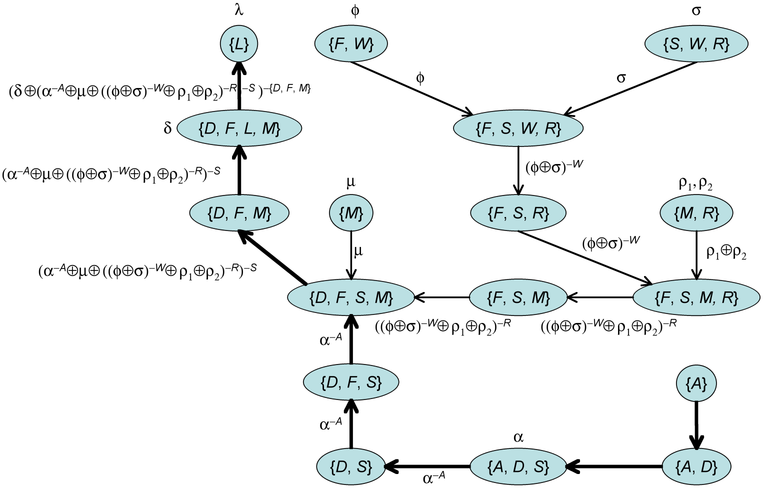


Local Computation as Message Passing

- Suppose we now wish to find the marginal for L.
- We re-orient some edges so L is the root and compute new messages for the re-oriented edges.

Local Computation as Message Passing

- End of propagation for L:



Outline

Applications: Belief Function Machine

- A software to build belief function models and compute marginals using local computation
- Implemented in MATLAB
- Written in 2002 by Phan Giang under the supervision of Philippe Smets, Thierry Denoeux, and I. Further developed by Sushila Shenoy in 2003.
- Features:
  - Belief function model is input as a text file using a language called UIL (unified input language)
  - Can solve "large" models/
  - Solve means finding the marginal of the joint for variables of interest.
  - Can reduce the marginal belief function to probabilities.
  - Can do sensitivity analysis.
- Can be downloaded for free from https://pshenoy.ku.edu/Papers/BFM072503.zip

Applications: Captain's Problem

- Suppose we wish to solve the Captain's Problem
- Input the problem as a UIL file "captain.txt"
    - Define variables and their state spaces
    - Define valuations and their domains
    - Describe the details of each valuation as a BPA or as a BPA representing conditional knowledge
    - Conditional knowledge BPAs are converted to conditional BPAs using Smets' conditional embedding

Applications: Captain's Problem

A snippet of UIL file Captain.txt for Captain's Problem:
DEFINE VARIABLE A { 0 1 2 3 4 5 6 };
DEFINE VARIABLE D { 0 1 2 3 };
DEFINE VARIABLE S { 0 1 2 3 };
DEFINE VARIABLE L { t f };
etc.
The arrival delay is the sum of departure and sailing delays
DEFINE RELATION ADS { A D S };
SET VALUATION ADS { (3 3 0) (5 2 3) (5 3 2) (3 2 1) (1 1 0) (3 1 2) (3 0 3) (1 0 1) (4 2 2) (2 2 0) (6 3 3) (4 3 1) (2 1 1) (4 1 3) (2 0 2) (0 0 0) } 1.0;
Heavy weather and repair at sea add one day to the sailing delay. # This proposition is true about 90% of the time DEFINE RELATION SWR { S W R };
SET VALUATION SWR { (2 t t) (1 f t) (1 t f) (0 f f) } 0.9;
Given the ship has undergone maintenance, the chance it needs a repair # at sea is between 10% to 30%
DEFINE CONDITIONAL RELATION RM1 { R } GIVEN { M };
SET CONDITIONAL VALUATION RM1 GIVEN {t } { (t) } 0.1, { (f) } 0.7;

Applications: Captain's Problem

Solution of Captain's problem using BFM in Matlab.

Script:

```
uil2bm('Captain.txt','bmcaptain');
global BELIEF VARIABLE ATTRIBUTE STRUCTURE FRAME QUERY BELTRACE NODE BJTREE
TRANPROTOCOL;
belall = condiembed([BELIEF(:).number]);
keepbel = belall;
showbel(solve(belall,'A'));
showbel(bel2prob(38));
```

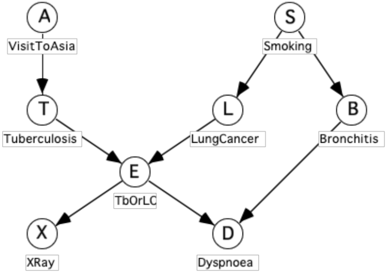
Belief Function Machine: Captain's Problem

MatLab output
Belief # 38 in plausibilistic probability

A	value
2	0.23852
1	0.2356
3	0.23366
0	0.15473
4	0.1240
5	0.0134

Applications: Chest Clinic

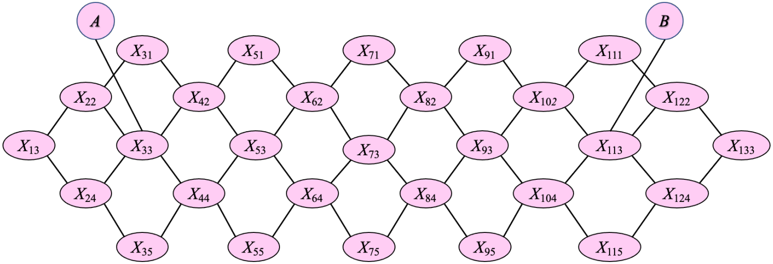
- BFM can solve complete Bayes net (BN) models using belief functions.
- Consider Chest Clinic example from [Lauritzen-Spiegelhalter 1988].
- BFM gives the same answers as Bayes net software.
- If we are missing some priors/conditionals, using Bayes net software is not an option. We can use BFM to solve such incomplete BNs.



P(A):	$p(a) = .01$	P(E|L,T):	$p(e \mid l, t) = 1$ $p(e \mid l, \sim t) = 1$ $p(e \mid \sim l, t) = 1$ $p(e \mid \sim l, \sim t) = 0$
P(T|A):	$p(t \mid a) = .05$ $p(t \mid \sim a) = .01$		
P(S):	$p(s) = .50$	P(X|E):	$p(x \mid e) = .98$ $p(x \mid \sim e) = .05$
P(L|S):	$p(l \mid s) = .10$ $p(l \mid \sim s) = .01$	P(D|E,B):	$p(d \mid e, b) = .90$ $p(d \mid e, \sim b) = .70$ $p(d \mid \sim e, b) = .80$ $p(d \mid \sim e, \sim b) = .10$
P(B|S):	$p(b \mid s) = .60$ $p(b \mid \sim s) = .30$		

Applications: Communication Network

- Communication network [Haenni-Lehmann 2002]
- We have a grid of $29 = 5 + 6 + 7 + 6 + 5$ communication nodes arranged in 5 rows
- There are 44 links, and each link has 90% reliability
- Nodes A and B are connected to the grid with links having 80% reliability
- What is the reliability of the connection between A and B? (Ans: $\approx 64\%$)



Applications: Communication Network

Solution of Communication network using BFM in Matlab.

Script:

```
uil2bm('comm5.txt', 'bmcomm5');
global BELIEF VARIABLE ATTRIBUTE STRUCTURE FRAME QUERY BELTRACE NODE BJTREE TRANPROTOCOL;
showbel(solve([1:46], [1,2]));
```

Output in MatLab:

A B | value

---

t t | 0.63802
f f |

****| 0.36198

Outline

References

- VBS: Shenoy, PP, "A Valuation-Based Language for Expert Systems," *Int. J. of Approx. Reas.*, 3(2), 1989, 383–411.
- Conditional Independence: Shenoy, PP, "Conditional independence in valuation-based systems," *Int. J. of Approx. Reas.*, 10(3), 1994, 203–234.
- Conditional Belief Functions: Jiroušek, R, V Kratochvíl & PP Shenoy, "On conditional belief functions in directed graphical models in the Dempster-Shafer theory," *Int. J. of Approx. Reas.*, 160(9), 2023.
- Local Computation: Shenoy, PP & G Shafer, "Axioms for Probability and Belief-Function Propagation," in Shachter, RD, T Levitt, JF Lemmer and LN Kanal (eds.), *Uncert. in Art. Int.*, 4, 1990, 169–198.
- Captain's Problem: Almond, R, *Graphical Belief Modeling*, 1995, Chapman & Hall
- Chest Clinic: SL Lauritzen and DJ Spiegelhalter, "Local computations with probabilities on graphical structures and their application to experts systems" *JRSS*, ser. B, 50(2), 1988, 157–224.
- Incomplete BNs: PP Shenoy, "Making inferences in incomplete Bayesian networks: A Dempster-Shafer belief function approach," *Int. J. of Approx. Reas.*, 160(9), 2023.
- Communication network: R Haenni and N Lehmann, "Resource bounded and anytime approximation of belief function computations," *Int. J. of Approx. Reas.*, 31(1–2), 2002, 103–154.

Questions

Questions?