

Representation of Bayesian Networks as Relational Databases

S.K.M. Wong, Y. Xiang and X. Nie

Department of Computer Science

University of Regina

Regina, Saskatchewan, Canada S4S 0A2

1 Introduction

A Bayesian network can be regarded as a summary of a domain expert's experience with an implicit population. A database can be regarded as a detailed documentation of such an experience with an explicit population. This connection between Bayesian networks and databases is well recognized and have been pursued for knowledge acquisition [1, 2, 11]. Existing databases are treated as information resources. Automatic generation of Bayesian networks from databases are studied as a way to bypass the knowledge acquisition bottleneck. Once Bayesian networks are generated, databases are regarded as no longer relevant to the evidential reasoning process in Bayesian networks.

Much evidence suggests that even deeper connection between Bayesian networks and relational databases exists. Relational databases manipulate tables of tuples, and Bayesian networks manipulate tables of probabilities. Relational databases answer queries that involve attributes in multiple relations by joining the relations and then projecting to the set of target attributes. In Bayesian networks, joint distributions are defined by products of local distributions, and belief updating [9] computes the marginalization of joint distributions. Relational databases make use of junction (join) trees in database design [7]. Multiply connected Bayesian networks are transformed into junction trees [5] or junction forests [13] to achieve inference efficiency. Scenario based explanation [3] in Bayesian networks uses the most likely configurations, which are equivalent to the universal tuples that repeat most frequently if we allow repetition in the database. Sequential updating (learning) of conditional probabilities [10, 12] in Bayesian networks makes use of new cases which are new tuples in databases.

The paper explores the connection between Bayesian networks and relational databases. We presents a representation and inference framework of Bayesian networks using relational databases. The framework is based on the junction tree representation of Bayesian networks [5]. We show that with some minor extension to the conventional relational database model, it can be used to represent Bayesian networks and to perform probabilistic inference. This unified framework formally reveals the close relationship

between Bayesian networks and relational databases. It allows widely available relational database management systems to be used for probabilistic reasoning, and thus potentially facilitates the development and deployment of knowledge-based systems based on Bayesian networks.

2 Bayesian Networks

A Bayesian network [9, 8, 6, 5] is a triplet (N, E, P) . N is a set of nodes. Each node is labeled with a random variable that is associated with a space. Since the labeling is unique, we shall use ‘node’ and ‘variable’ interchangeably. E is a set of arcs such that $D = (N, E)$ is a directed acyclic graph (DAG). The arcs signify the existence of direct causal influences between the linked variables. For each node $A_i \in N$, the strengths of the causal influences from its parent nodes π_i is quantified by a conditional probability distribution $p(A_i|\pi_i)$ of A_i conditioned on the values of A_i ’s parents. The basic dependency assumption embedded in Bayesian networks is that a variable is independent of its non-descendants given its parents. P is a joint probability distribution. For a Bayesian network with n nodes, P can be specified by the following factored form due to the assumption: $P = p(A_1 \dots A_n) = \prod_{i=1}^n p(A_i|\pi_i)$.

For example, the probability distribution for a Bayesian network shown in Figure 1 can be specified as

$$p(x_1, x_2, x_3, x_4, x_5, x_6) = p(x_1)p(x_2|x_1)p(x_3|x_1)p(x_4|x_1, x_2)p(x_5|x_2, x_3)p(x_6|x_5) \quad (1)$$

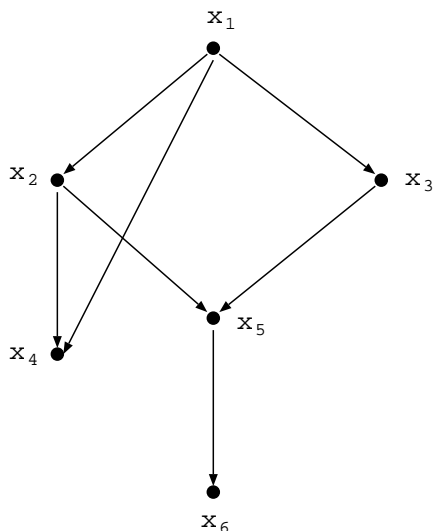


Figure 1: The DAG of a Bayesian Network.

The joint probability distribution of a Bayesian network can be equivalently factored based on an undirected graph derived from the original DAG [6, 5]. For example, the

above joint distribution $p(x_1, x_2, \dots, x_6)$ can be written as a product of the distributions of the cliques of the graph G (depicted in Figure 2) divided by a product of the distributions of their intersections, namely:

$$p(x_1, x_2, x_3, x_4, x_5, x_6) = \frac{p(x_1, x_2, x_3)p(x_1, x_2, x_4)p(x_2, x_3, x_5)p(x_5, x_6)}{p(x_1, x_2)p(x_2, x_3)p(x_5)} \quad (2)$$

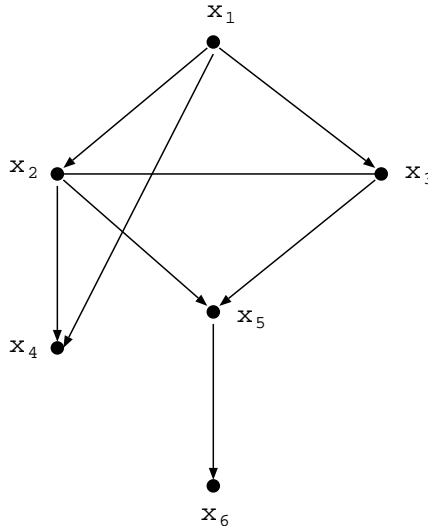


Figure 2: An Undirected Graph G derived from the DAG in Figure 1.

Note that the conditional independence, $p(x_2, x_3|x_4) = p(x_2|x_1)p(x_3|x_1)$, in equation (1) is not explicitly represented in equation (2). It will be shown, however, that a tree organization of the cliques of G as shown in Figure 1 provides a convenient way of representing a Bayesian network as a relational database system and of performing probabilistic inference in that network.

3 Representation of a Factored Probability Distribution as a Product on a Hypertree

In order to facilitate the development of a relational representation of a Bayesian network and to use it for answering queries that involve marginal distributions of the network, we first demonstrate how to express a joint probability distribution as a product on a *hypertree*¹.

¹Other terms like ‘junction tree’ and ‘joint tree’ have been used to denote hypertree [9, 4, 5, 7] in the literature. We will use ‘hypertree’ throughout the rest of the paper.

3.1 Hypergraphs and Hypertrees

Let \mathcal{L} denote a lattice. We say that \mathcal{H} is a *hypergraph*, if \mathcal{H} is a finite subset of \mathcal{L} . Consider, for example, the power set $2^{\mathcal{X}}$, where $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$ is a set of variables. The power set $2^{\mathcal{X}}$ is a lattice of all subsets of \mathcal{X} . Any subset of $2^{\mathcal{X}}$ is a hypergraph on $2^{\mathcal{X}}$. We say that an element t in a hypergraph \mathcal{H} is a *twig* if there exists another element b in \mathcal{H} , distinct from t , such that $t \cap (\cup(\mathcal{H} - \{t\})) = t \cap b$. We call any such b a *branch* for the twig t . A hypergraph \mathcal{H} is a *hypertree* if its elements can be ordered, say h_1, h_2, \dots, h_n , so that h_i is a twig in $\{h_1, h_2, \dots, h_i\}$, for $i = 2, \dots, n$. We call any such ordering a *hypertree construction ordering* for \mathcal{H} . Given a hypertree construction ordering h_1, h_2, \dots, h_n , we can choose, for i from 2 to n , an integer $b(i)$ such that $1 \leq b(i) \leq i - 1$ and $h_{b(i)}$ is a branch for h_i in $\{h_1, h_2, \dots, h_n\}$. We call the function $b(i)$ satisfying this condition a *branch function* for \mathcal{H} and h_1, h_2, \dots, h_n .

For example, let $\mathcal{X} = \{x_1, x_2, \dots, x_6\}$ and $\mathcal{L} = 2^{\mathcal{X}}$. Consider a hypergraph, $\mathcal{H} = \{h_1 = \{x_1, x_2, x_3\}, h_2 = \{x_1, x_2, x_4\}, h_3 = \{x_2, x_3, x_5\}, h_4 = \{x_5, x_6\}\}$, depicted in Figure 3. This hypergraph is in fact a hypertree; the ordering, for example, h_1, h_2, h_3, h_4 , is a hypertree construction ordering. Furthermore, we obtain: $b(1) = h_3, b(2) = h_1$, and $b(4) = h_3$.

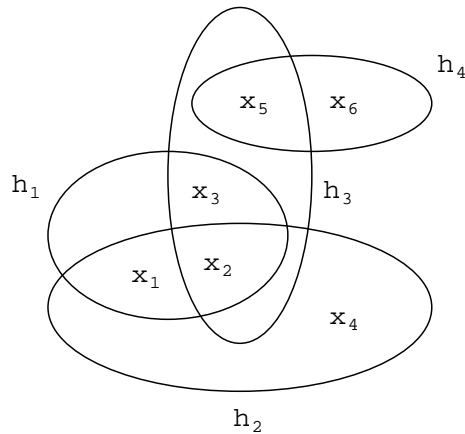


Figure 3: A Graphical Representation of the Hypergraph $\mathcal{H} = \{h_1 = \{x_1, x_2, x_3\}, h_2 = \{x_1, x_2, x_4\}, h_3 = \{x_2, x_3, x_5\}, h_4 = \{x_5, x_6\}\}$.

A hypertree \mathcal{K} on \mathcal{L} is called a *hypertree cover* for a given hypergraph \mathcal{H} on \mathcal{L} if for every element h of \mathcal{H} , there exists an element $k(h)$ of \mathcal{K} such that $h \subseteq k(h)$. In general, a hypergraph \mathcal{H} may have many hypertree covers. For example, the hypertree depicted in Figure 3 is a hypertree cover of the hypergraph

$$\{\{x_1, x_2\}, \{x_1, x_3\}, \{x_2, x_5\}, \{x_3, x_5\}, \{x_1, x_2, x_4\}, \{x_5, x_6\}\}.$$

3.2 Factored Probability Distributions

Let $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$ denote a set of variables. A factored probability distribution $p(x_1, x_2, \dots, x_n)$ can be written as

$$p(x_1, x_2, \dots, x_n) = \phi = \phi_{h_1} \phi_{h_2} \dots \phi_{h_n}$$

where each h_i is a subset of \mathcal{X} , i.e., $h_i \in 2^{\mathcal{X}}$, and ϕ_{h_i} is a real-valued function on h_i . Moreover, $\mathcal{X} = h_1 \cup h_2 \cup \dots \cup h_n = \bigcup_{i=1}^n h_i$. By definition, $\mathcal{H} = \{h_1, h_2, \dots, h_n\}$ is a hypergraph on the lattice $2^{\mathcal{X}}$. Thus, a factored probability distribution can be viewed as a product on a hypergraph \mathcal{H} , namely:

$$\phi = \prod_{h \in \mathcal{H}} \phi_h.$$

Let v_x denote the discrete frame (state space) of the variable $x \in \mathcal{X}$. We call an element of v_x a *configuration* of x . We define v_y to be the cartesian product of the frames of the variables in a subset $y \subseteq \mathcal{X}$:

$$v_y = \times_{x \in y} v_x.$$

We call v_y the frame of y , and we call its elements configurations of y .

Let $h, k \in 2^{\mathcal{X}}$, and $h \subseteq k$. If c is a configuration of k , i.e., $c \in v_k$, we write $c^{\downarrow h}$ for the configuration of h obtained by deleting the values of the variables in k and not in h . For example, let $h = \{x_1, x_2\}$, $k = \{x_1, x_2, x_3, x_4\}$, and $c = (c_1, c_2, c_3, c_4)$, where $c_i \in v_{x_i}$. Then, $c^{\downarrow h} = (c_1, c_2)$.

If h and k are disjoint subsets of \mathcal{X} , c_h is a configuration of h , and c_k is a configuration of k , then we write $(c_h * c_k)$ for the configuration of $h \cup k$ obtained by *concatenating* c_h and c_k . In other words, $(c_h * c_k)$ is the unique configuration of $h \cup k$ such that $(c_h * c_k)^{\downarrow h} = c_h$ and $(c_h * c_k)^{\downarrow k} = c_k$. Using the above notation, a factored probability distribution ϕ can be defined as follows:

$$\phi(c) = \left(\prod_{h \in \mathcal{H}} \phi_h \right)(c) = \prod_{h \in \mathcal{H}} \phi_h(c^{\downarrow h})$$

where $c \in v_{\mathcal{X}}$ is any configuration.

3.3 Marginalization

Consider a function ϕ_k on a set k of variables. If $h \subseteq k$, then $\phi_k^{\downarrow h}$ denotes the function on h defined as follows:

$$\phi_k^{\downarrow h}(c) = \sum_{c'} \phi_k(c * c')$$

where c is a configuration of h , c' is a configuration of $k - h$, and $c * c'$ is a configuration of k . We shall call $\phi_k^{\downarrow h}$ the *marginal* of ϕ_k on h .

A major task in probabilistic reasoning in Bayesian networks is to compute marginals as new evidence becomes available.

4 Representation of a Factored Probability Distribution as a Generalized Acyclic Join Dependency

$$\phi_h =$$

x_1	x_2	\dots	\dots	\dots	x_7	f_{ϕ_h}
c_{11}	c_{12}	\dots	\dots	\dots	c_{17}	$\phi_h(c_1)$
c_{21}	c_{22}	\dots	\dots	\dots	c_{27}	$\phi_h(c_2)$
\vdots	\vdots	\dots	\dots	\dots	\vdots	\vdots
\vdots	\vdots	\dots	\dots	\dots	\vdots	\vdots
c_{s1}	c_{s2}	\dots	\dots	\dots	c_{s7}	$\phi_h(c_s)$

Figure 4:

Let c be a configuration of \mathcal{X} . Consider a factored probability distribution

$$\phi(c) = \prod_{h \in \mathcal{H}} \phi_h(c^{\downarrow h})$$

We can equivalently express each function ϕ_h in the above product as a *relation*. Suppose $h = \{x_1, x_2, \dots, x_l\}$. The function ϕ_h can be expressed as a relation on the set $\{x_1, x_2, \dots, x_l, f_{\phi_h}\}$ of *attributes* shown in Figure 4. In Figure 4, a configuration $c_i = (c_{i1}, c_{i2}, \dots, c_{il})$ is expressed as one row excluding the last element in the row, and s is the cardinality of v_h .

By definition, the product $\phi_h \cdot \phi_k$ of any two function ϕ_h and ϕ_k is given by:

$$(\phi_h \cdot \phi_k)(c) = \phi_h(c^{\downarrow h}) \cdot \phi_k(c^{\downarrow k})$$

where $c \in v_{h \cup k}$. We can therefore express the product $\phi_h \cdot \phi_k$ equivalently as a *join* of relation ϕ_h and ϕ_k , written $\phi_h \otimes \phi_k$, which is defined as follows:

- (i) Compute the *natural join*, $\phi_h \bowtie \phi_k$, of the two relations of ϕ_h and ϕ_k .
- (ii) Add a new column with attribute $f_{\phi_h \cdot \phi_k}$ to the relation $\phi_h \bowtie \phi_k$ on $h \cup k$. Each value of $f_{\phi_h \cdot \phi_k}$ is given by $\phi_h(c^{\downarrow h}) \cdot \phi_k(c^{\downarrow k})$ where $c \in v_{h \cup k}$.
- (iii) Obtain the resultant relation $\phi_h \otimes \phi_k$ by projecting the relation obtained in step (ii) on the set of attributes $h \cup k \cup \{f_{\phi_h \cdot \phi_k}\}$.

For example, let $h = \{x_1, x_2\}$, $k = \{x_2, x_3\}$, and $v_h = v_k = \{0, 1\}$. The join $\phi_h \otimes \phi_k$ is illustrated in Figure 5.

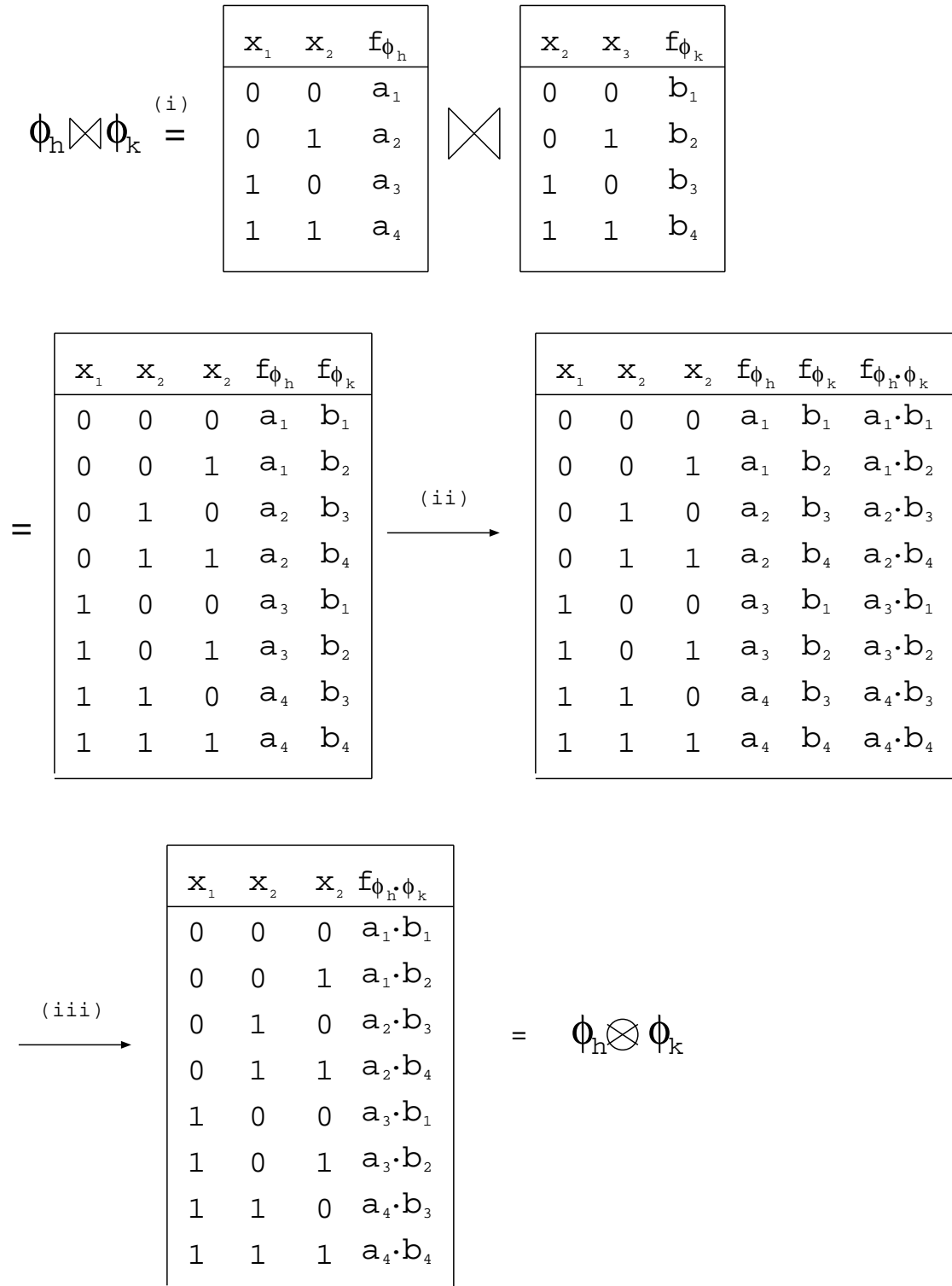


Figure 5: The join of two relations: ϕ_h and ϕ_k .

$$\phi_h =$$

x_1	x_2	\dots	x_7	f_{ϕ_h}
c_{11}	c_{12}	\dots	c_{11}	1
c_{21}	c_{22}	\dots	c_{21}	1
\vdots	\vdots		\vdots	
\vdots	\vdots		\vdots	
c_{s1}	c_{s2}	\dots	c_{s1}	1

Figure 6: A conventional relation with an unit function attribute added.

If all the functions ϕ 's are *unit* functions (see Figure 6), then the binary operator \otimes defined above is identical to the natural join operator \bowtie in conventional databases.

Since the operator \otimes is both commutative and associative, we can express a factored probability distribution as a join of relations:

$$\phi = \prod_{h \in \mathcal{H}} \phi_h = \bigotimes_{h \in \mathcal{H}} \phi_h.$$

we can also define marginalization as a relational operation. Let $\phi_k^{\downarrow h}$ denote the relation obtained by marginalizing ϕ_k on $h \subseteq k$. We can construct $\phi_k^{\downarrow h}$ in two steps:

- (a) Project the relation ϕ_k on the set of attributes $h \cup \{f_{\phi_k}\}$, without eliminating identical configurations.
- (b) For every $c_h \in v_h$, replace the set of configurations of $h \cup \{f_{\phi_k}\}$ in the relation obtained from step (a) by the singleton configuration $c_h * (\sum_{c_{k-h}} \phi_k(c_h * c_{k-h}))$.

Consider, for example, the relation ϕ_k with $k = \{x_1, x_2, x_3\}$ as shown in Figure 7. Suppose we want to compute $\phi_k^{\downarrow h}$ for $h = \{x_1, x_2\}$. From step (a), we obtain the relation in Figure 8 by projecting ϕ_k on $h \cup \{f_{\phi_k}\}$. The result after step (b) is shown in Figure 9.

Two important properties are satisfied by the relational operator of marginalization:

$$\phi_k = \begin{array}{|c|c|c|c|} \hline \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \mathbf{f}_{\phi_k} \\ \hline 0 & 0 & 0 & \mathbf{d}_1 \\ 0 & 0 & 1 & \mathbf{d}_2 \\ 0 & 1 & 0 & \mathbf{d}_3 \\ 0 & 1 & 1 & \mathbf{d}_3 \\ 1 & 0 & 0 & \mathbf{d}_4 \\ 1 & 0 & 1 & \mathbf{d}_4 \\ 1 & 1 & 0 & \mathbf{d}_5 \\ 1 & 1 & 1 & \mathbf{d}_6 \\ \hline \end{array}$$

Figure 7: A relation ϕ_k with attributes $x_1, x_2, x_3, f_{\phi_k}$.

Lemma 1

(1) If ϕ_k is a relation on k , and $h \subseteq g \subseteq k$, then

$$(\phi_k^{\downarrow g})^{\downarrow h} = \phi_k^{\downarrow h}.$$

(2) If ϕ_h and ϕ_k are relations on h and k , respectively, then

$$(\phi_h \otimes \phi_k)^{\downarrow h} = \phi_h \otimes (\phi_k^{\downarrow h \cap k}).$$

Proof:

(1) By definition, a configuration of the set $g \cup \{f_{\phi_k^{\downarrow g}}\}$ of attributes in the relation $\phi_k^{\downarrow g}$ is

$$c_g * \phi_g(c_g) = c_g * \left(\sum_{c_{k-g}} \phi_k(c_g * c_{k-g}) \right)$$

where $c_g \in v_g$. Similarly, a configuration of the set of attributes $h \cup \{f_{\phi_g^{\downarrow h}}\}$ in the relation $\phi_g^{\downarrow h} = (\phi_k^{\downarrow g})^{\downarrow h}$ is

$$c_h * \sum_{c_{g-h}} \phi_g(c_h * c_{g-h}) = c_h * \left(\sum_{c_{g-h}} \sum_{c_{k-g}} \phi_k(c_h * c_{g-h} * c_{k-g}) \right) = c_h * \left(\sum_{c_{k-h}} \phi_k(c_h * c_{k-h}) \right)$$

which is a configuration of the set of attributes in the relation $\phi_k^{\downarrow h}$.

(2) A configuration of the set of attributes $h \cup k \cup \{f_{\phi_h \otimes \phi_k}\}$ in the relation $\phi_h \otimes \phi_k$ is

$$c * (\phi_h(c^{\downarrow h}) \cdot \phi_k(c^{\downarrow k}))$$

where $c \in v_{h \cup k}$. Thus a configuration of the set of attributes $h \cup f_{\phi_h \otimes \phi_k}$ in the relation $(\phi_h \otimes \phi_k)^{\downarrow h}$ is

$$c^{\downarrow h} * (\phi_h(c^{\downarrow h}) \cdot \sum_{c_{k-h}} \phi_k(c^{\downarrow h \cap k} * c_{k-h}))$$

\mathbf{x}_1	\mathbf{x}_2	\mathbf{f}_{ϕ_k}
0	0	\mathbf{d}_1
0	0	\mathbf{d}_2
0	1	\mathbf{d}_3
0	1	\mathbf{d}_3
1	0	\mathbf{d}_4
1	0	\mathbf{d}_4
1	1	\mathbf{d}_5
1	1	\mathbf{d}_6

Figure 8: The intermediate result (after step (a)) of marginalization $\phi_k^{\downarrow h}$ of the relation ϕ_k in Figure 7 relative to $h = \{x_1, x_2\}$.

\mathbf{x}_1	\mathbf{x}_2	$\mathbf{f}_{\phi_k^{\downarrow h}}$
0	0	$\mathbf{d}_1 + \mathbf{d}_2$
0	1	$\mathbf{d}_3 + \mathbf{d}_3$
1	0	$\mathbf{d}_4 + \mathbf{d}_4$
1	1	$\mathbf{d}_5 + \mathbf{d}_6$

Figure 9: The marginalization $\phi_k^{\downarrow h}$ of the relation ϕ_k in Figure 7 relative to $h = \{x_1, x_2\}$.

On the other hand, by definition, a configuration of the set of attributes $(h \cap k) \cup \{f_{\phi_k^{\downarrow h \cap k}}\}$ in the relation $\phi_k^{\downarrow h \cap k}$ is

$$(c^{\downarrow h \cap k} * \sum_{c_{k-h}} \phi_k(c^{\downarrow h \cap k} * c_{k-h})).$$

Also, a configuration of the set of attributes $h \cup \{f_{\phi_h}\}$ in the relation ϕ_h is

$$(c^{\downarrow h} * \phi_h(c^{\downarrow h})) = (c^{\downarrow h-k} * c^{\downarrow h \cap k} * \phi_h(c^{\downarrow h})).$$

By the definition of the join operation \otimes , we can therefore conclude that

$$(\phi_h \otimes \phi_k)^{\downarrow h} = \phi_h \otimes \phi_k^{\downarrow h \cap k}. \quad \square$$

It should be noted that when all the relations ϕ involved represent *unit* functions, the join operator \otimes is equivalent to the natural join operator \bowtie , marginalization

becomes projection, and the relation $(\phi_h \otimes \phi_k) \downarrow^h$ is the semi-join of ϕ_h and ϕ_k as defined in standard relational database systems. Both equalities $(\phi_k \downarrow^g) \downarrow^h = \phi_k \downarrow^h$ and $(\phi_h \otimes \phi_k) \downarrow^h = \phi_h \otimes \phi_k \downarrow^{h \cap k}$ are trivially satisfied. Before discussing the computation of marginals of a factored distribution, let us first state the notion of computational feasibility introduced by Shafer. We call a set of attributes *feasible* if it is feasible to represent relations on these attributes, join them, and marginalize on them. We assume that any subset of feasible attributes is also feasible. Furthermore, we assume that the factored distribution is represented on a hypertree and every element in \mathcal{H} is feasible.

Lemma 2 *Let $\phi = \otimes_{h \in \mathcal{H}} \phi_h$ be a factored probability distribution on a hypertree \mathcal{H} . Let t be a twig in \mathcal{H} and b be a branch for t . Then the followings hold:*

- (i) $(\otimes\{\phi_h|h \in \mathcal{H}\})^{\cup \mathcal{H}^{-t}} = (\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}) \otimes \phi_t \downarrow^{t \cap b}$.
- (ii) *If $k \subseteq \cup \mathcal{H}^{-t}$, then $(\otimes\{\phi_h|h \in \mathcal{H}\}) \downarrow^k = (\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}) \downarrow^k$ where \mathcal{H}^{-t} denotes the set of hyperedges $\mathcal{H} - \{t\}$, $\phi_b^{-t} = \phi_b \otimes \phi_t \downarrow^{t \cap b}$, and $\phi_h^{-t} = \phi_h$ for all other h in \mathcal{H}^{-t} .*

Proof: Since $\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}$ is a relation on $\cup \mathcal{H}^{-t}$ and $t \cap (\cup \mathcal{H}^{-t}) = t \cap b$, we obtain from property (2) of Lemma 1:

$$\begin{aligned} (\otimes\{\phi_h|h \in \mathcal{H}\}) \downarrow^{\cup \mathcal{H}^{-t}} &= ((\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}) \otimes \phi_t) \downarrow^{\cup \mathcal{H}^{-t}} \\ &= (\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}) \otimes \phi_t \downarrow^{t \cup \mathcal{H}^{-t}} \\ &= (\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}) \otimes \phi_t \downarrow^{t \cup b}. \end{aligned}$$

The right-hand side of the above equation can be rewritten as:

$$(\otimes\{\phi_h|h \in \mathcal{H}^{-t}\}) \otimes \phi_t \downarrow^{t \cup b} = \otimes\{\phi_h^{-t}|h \in \mathcal{H}\}.$$

Since $k \subseteq \mathcal{H}^{-t}$, by property (1) of Lemma 1, it follows:

$$\begin{aligned} (\otimes\{\phi_h|h \in \mathcal{H}\}) \downarrow^k &= ((\otimes\{\phi_h|h \in \mathcal{H}\}) \downarrow^{\cup \mathcal{H}^{-t}}) \downarrow^k \\ &= (\otimes\{\phi_h^{-t}|h \in \mathcal{H}^{-t}\}) \downarrow^k. \quad \square \end{aligned}$$

We now describe an algorithm for computing $\phi \downarrow^k$ for $k \in \mathcal{H}$, where $\phi = \otimes\{\phi_h|h \in \mathcal{H}\}$ and \mathcal{H} is a hypertree. Choose a hypertree construction ordering for \mathcal{H} that begins with $h_1 = k$ as the root, say h_1, h_2, \dots, h_n , and choose a branching $b(i)$ for constructing ordering. For $i = 1, 2, \dots, n$, let

$$\mathcal{H}^i = \{h_1, h_2, \dots, h_i\}.$$

This is a sequence of sub-hypertree, each larger than the last; $\mathcal{H}^1 = \{h_1\}$ and $\mathcal{H}^n = \mathcal{H}$. The element h_i is a twig in \mathcal{H}^i . To compute $\phi \downarrow^k$, we start with \mathcal{H}^n going backwards in this sequence. We use Lemma 3 each time to perform the *reduction*. At the step from

\mathcal{H}^i to \mathcal{H}^{i-1} , we go from $\phi^{\downarrow \cup \mathcal{H}^i}$ to $\phi^{\downarrow \cup \mathcal{H}^{i-1}}$. we omit h_i in \mathcal{H}^i to \mathcal{H}^{i-1} and change the relation on $h_{b(i)}$ in \mathcal{H}^{i-1} from $\phi_{h_{b(i)}}^i$ to

$$\phi_{h_{b(i)}}^{i-1} = \phi_{h_{b(i)}}^i \otimes (\phi_{h_i}^i)^{\downarrow h_i \cap h_{b(i)}},$$

and the other relations in \mathcal{H}^{i-1} are not changed. The collection of relations with which we begin, $\{\phi_h^n | h \in \mathcal{H}^n\}$, is simply $\{\phi_h | h \in \mathcal{H}\}$, and the collection with which we end, $\{\phi_h^1 | h \in \mathcal{H}^1\}$, consists of the single relation $\phi_h^1 = \phi^{\downarrow h_1}$.

Consider a factored probability distribution $\phi = \otimes \{\phi_h | h \in \mathcal{H}\}$ on a hypertree $\mathcal{H} = \{h_1, h_2, \dots, h_n\}$. We say that ϕ satisfies the acyclic join dependency (AJD), $*[h_1, h_2, \dots, h_n]$, if ϕ decomposes losslessly onto h_1, h_2, \dots, h_n , i.e., ϕ can be expressed as:

$$\phi = \phi^{\downarrow h_1} \otimes' \phi^{\downarrow h_2} \otimes \dots \otimes \phi^{\downarrow h_n},$$

where \otimes' is a generalized join operator defined by:

$$\phi^{\downarrow h_i} \otimes' \phi^{\downarrow h_j} = \phi^{\downarrow h_i} \otimes \phi^{\downarrow h_j} \otimes \phi^{\downarrow h_i \cap h_j}.$$

Theorem 3 Any factored probability distribution $\phi = \otimes \{\phi_h | h \in \mathcal{H}\}$ on a hypertree, $\mathcal{H} = \{h_1, h_2, \dots, h_n\}$, decomposes losslessly onto h_1, h_2, \dots, h_n . That is, ϕ satisfies the AJD, $*[h_1, h_2, \dots, h_n]$.

Proof:

Suppose $t \in \mathcal{H}$ is a twig. By the properties of conditional independence in a factored probability distribution, we can always define a set of relation $\{\phi'_{h_1}, \phi'_{h_2}, \dots, \phi'_t, \dots, \phi'_{h_n}\}$ such that

$$\phi = \otimes \{\phi_h | h \in \mathcal{H}\} = \otimes \{\phi'_h | h \in \mathcal{H}\}$$

and $\phi'_t = p(t|t \cap b)$, where b is a branch of t in \mathcal{H} . By Lemma 2,

$$\begin{aligned} \phi^{\downarrow \cap \mathcal{H}} &= ((\otimes \{\phi'_h | h \in \mathcal{H}^{-t}\}) \otimes \phi'_t)^{\downarrow \cap \mathcal{H}^{-t}} \\ &= (\otimes \{\phi'_h | h \in \mathcal{H}^{-t}\}) \otimes \phi'_t{}^{\downarrow t \cap \mathcal{H}^{-t}} \\ &= (\otimes \{\phi'_h | h \in \mathcal{H}^{-t}\}) \otimes \phi'_t{}^{\downarrow t \cap b}. \end{aligned}$$

Thus,

$$\phi^{\downarrow \cup \mathcal{H}^{-t}} \otimes \left(\frac{\phi'_t}{\phi'_t{}^{\downarrow t \cap b}} \right) = (\otimes \{\phi'_h | h \in \mathcal{H}^{-t}\}) \otimes \phi'_t{}^{\downarrow t \cap b} \otimes \left(\frac{\phi'_t}{\phi'_t{}^{\downarrow t \cap b}} \right) = \phi.$$

Since $\phi'_t = p(t|t \cap b)$, we obtain

$$\phi'_t{}^{\downarrow t \cap b} = 1.$$

It follows:

$$\frac{\phi'_t}{\phi'_t{}^{\downarrow t \cap b}} = p(t|t \cap b) = \frac{p(t)}{p(t \cap b)} = \frac{\phi^{\downarrow t}}{\phi^{\downarrow t \cap b}}.$$

therefore, ϕ can be written as:

$$\phi = \phi^{\downarrow \cup \mathcal{H}^{-t}} \otimes \frac{\phi^{\downarrow t}}{\phi^{\downarrow t \cap b}} = \phi^{\downarrow \cup \mathcal{H}^{-t}} \otimes \phi^{\downarrow t}.$$

Also, we have:

$$\phi^{\downarrow \cup \mathcal{H}^{-t}} = \otimes \{\phi'_h | h \in \mathcal{H}\}.$$

We can immediately apply the same procedure to $\phi^{\downarrow \cup \mathcal{H}^{-t}}$ for further reduction. Thus, by applying this procedure recursively, we obtain the desired result. \square

References

- [1] C.K. Chow and C.N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, (14):462–467, 1968.
- [2] G.F. Cooper and E. Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, (9):309–347, 1992.
- [3] M. Henrion and M.J. druzdel. Qualitative propagation and scenario-based approaches to explanation of probabilistic reasoning. In *Proc. Sixth Conference on Uncertainty in Artificial Intelligence*, pages 10–20, Cambridge, Mass., 1990.
- [4] F.V. Jensen. Junction tree and decomposable hypergraphs. Technical report, JUDEX, Aalborg, Denmark, February 1988.
- [5] F.V. Jensen, S.L. Lauritzen, and K.G. Olesen. Bayesian updating in causal probabilistic networks by local computations. *Computational Statistics Quarterly*, (4):269–282, 1990.
- [6] S.L. Lauritzen and D.J. Spiegelhalter. Local computation with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, (50):157–244, 1988.
- [7] D. Maier. *The Theory of Relational Databases*. Computer Science Press, 1983.
- [8] R.E. Neapolitan. *Probabilistic Reasoning in Expert Systems*. John Wiley and Sons, 1990.
- [9] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.
- [10] D.J. Spiegelhalter, R.C.G. Franklin, and K. Bull. Assessment, criticism and improvement of imprecise subjective probabilities for a medical expert system. In *Proc. Fifth workshop on uncertainty in artificial intelligence*, pages 335–342, Windsor, Ontario, 1989.

- [11] W.X. Wen. From relational databases to belief networks. In B. D'Ambrosio, P. Smets, and P.P. Bonissone, editors, *Proc. Seventh Conference on Uncertainty in Artificial Intelligence*, pages 406–413. Morgan Kaufmann, 1991.
- [12] Y. Xiang, M.P. Beddoes, and D. Poole. Sequential updating conditional probability in bayesian networks by posterior probability. In *Proc. 8th Biennial Conf. Canadian Society for Computational Studies of Intelligence*, pages 21–27, Ottawa, 1990.
- [13] Y. Xiang, D. Poole, and M. P. Beddoes. Multiply sectioned bayesian networks and junction forests for large knowledge based systems. *Computational Intelligence*, 9(2):171–220, 1993.