Kybernetika

Francesco M. Malvestuto

Tree and local computations in a cross–entropy minimization problem with marginal constraints

Kybernetika, Vol. 46 (2010), No. 4, 621--654

Persistent URL: http://dml.cz/dmlcz/140775

KYBERNETIKA — VOLUME *46* (2010), NUMBER 4, PAGES 621–654

TREE AND LOCAL COMPUTATIONS IN A CROSS–ENTROPY MINIMIZATION PROBLEM WITH MARGINAL CONSTRAINTS

Francesco M. Malvestuto

In probability theory, Bayesian statistics, artificial intelligence and database theory the minimum cross-entropy principle is often used to estimate a distribution with a given set P of marginal distributions under the proportionality assumption with respect to a given "prior" distribution q. Such an estimation problem admits a solution if and only if there exists an extension of P that is dominated by q. In this paper we consider the case that q is not given explicitly, but is specified as the maximum-entropy extension of an auxiliary set Q of distributions. There are three problems that naturally arise: (1) the existence of an extension of a distribution set (such as P and Q), (2) the existence of an extension of P that is dominated by the maximum entropy extension of Q, (3) the numeric computation of the minimum cross-entropy extension of P with respect to the maximum entropy extension of Q. In the spirit of a divide-and-conquer approach, we prove that, for each of the three above-mentioned problems, the global solution can be easily obtained by combining the solutions to subproblems defined at node level of a suitable tree.

Keywords: cross-entropy, acyclic hypergraph, connection tree, junction tree, probabilistic database, relational database

Classification: 93E12, 62A10

1. INTRODUCTION

The maximum entropy (ME) and minimum cross-entropy (MCE) principles are powerful tools used in probability theory, Bayesian statistics, in database theory and artificial intelligence to estimate probability distributions when only partial information is available. The ME and MCE principles have sound theoretical bases [9, 21, 24, 53]. A classical MCE problem consists in estimating an unknown probability distribution with given marginals under the hypothesis of proportionality to a given prior distribution [12, 13, 18, 25, 54]:

> Given a finite set X of discrete variables, a set P of distributions over X and a distribution q over X, if there exists an extension of P dominated by q, then compute the MCE extension of P with respect to q.

This problem can be solved as follows. Let Ω be the state space of X and let $P = \{p_1, \ldots, p_n\}$, where p_i is over the subset A_i of $X (1 \leq i \leq n)$. First of all, the

existence of an extension p of P dominated by q (the consistency of P w.r.t. q, for short) is tested by solving the following linear-programming problem:

$$\text{maximize } \sum_{x \in \|q\|} p(x)$$

subject to the linear constraint system

$$\begin{cases} p[A_i] & = p_i \quad (1 \le i \le n) \\ p(x) & \ge 0 \quad\quad (x \in \Omega) \end{cases} \tag{1}$$

where the term $\|q\|$ in the objective function denotes the "support" of q, that is, $\|q\| = \{x \in \Omega : q(x) > 0\}$, and $p[A_i]$ denotes the marginal of p on A_i $(1 \le i \le n)$. Of course, there exists an extension of P (P is consistent, for short) if and only if the linear constraint system (1) is consistent; in other words, the extensions of P are exactly the feasible solutions of the linear-programming problem above. Moreover, for every extension p of P one has $\sum_{x \in \|q\|} p(x) \le 1$ where the equality holds if and only if p is dominated by q (that is, $\|p\| \subseteq \|q\|$). Therefore, P is consistent w.r.t. q if and only if the linear-programming problem above is feasible and its optimum is one. Suppose that this is the case. Then, the MCE extension of P with respect to q (the q-MCE extension, for short) is computed by applying the *Iterative Proportional Fitting Procedure* (the IPFP, for short) to P with prior distribution q (cf. [7]). The convergence of the IPFP is well-established [11, 15, 52]. The above MCE criterion has been also used to estimate the unknown distribution p of a nonnegative summary statistic (e. g., total income) with given marginals (P), which is called the *target variable*, under the hypothesis of proportionality to the given distribution (q) of another nonnegative summary statistic (e. g., population), which is called the *auxiliary variable* [2, 30, 31, 45, 50, 51, 55]. In a more general framework [46], the distribution q of the auxiliary variable is not known explicitly, but a set Q of its marginals is given. Then, one can obtain an estimate of the unknown distribution of the target variable by solving the following MCE problem:

> Given a finite set X of discrete variables, a set P of distributions over X and a set Q of distributions over X, if Q is consistent and P is consistent w.r.t. the ME extension q of Q, then compute the q-MCE extension of P.

It should be noted that the MCE problem above requires solving three problems, two of decisional type and the other of numeric computation, which read:

(*Consistency*) — Are the distribution sets P and Q consistent?

(*Relative consistency*) — Is the distribution set P consistent w.r.t. the ME extension of Q?

(*MCE extension*) — Compute the q-MCE extension of the distribution set P, where q denotes the ME extension of the distribution set Q.

The computational complexity of each of these problems is proportional to the size of the state space Ω of X. In the spirit of a divide-and-conquer approach, for each of the three problems above we want to find a "decomposition", that is, a

family $\mathbf{F}$ of subsets of X such that the global solution can be "easily" obtained by combining the solutions to the corresponding subproblems induced by sets Y in $\mathbf{F}$ defined as follows:

- the subproblem of the consistency of P induced by Y is specified by the projection $P(Y)$ of P onto Y, that is, $P(Y) = \{p_1[A_1 \cap Y], \ldots, p_n[A_n \cap Y]\}$;
- the subproblem of the relative-consistency problem induced by Y is specified by $P(Y)$ and by the support $\|q[Y]\|$ of the marginal on Y of the ME extension q of Q;
- the subproblem of the MCE extension problem induced by Y is specified by $P(Y)$ and by the marginal $q[Y]$ on Y of the ME extension q of Q.

Since the computational complexity of such a subproblem is proportional to the size of the state space of Y, the divide-and-conquer approach allows to find a global solution in time proportional to the number of sets in $\mathbf{F}$ and to the maximal size of state spaces of sets in $\mathbf{F}$. In this paper, for each of the three problems above, we will construct a "decomposition tree", that is, a tree whose nodes represent the sets of a decomposition of the problem. Such decomposition trees are found by viewing set families as hypergraphs [6] and exploiting the hypergraph-theoretic characterizations [40] of the probability-theoretic notions of decomposability [17] and collapsibility [1].

Related Work. The first results on the consistency problem go back to Kellerer [23] who suggested to verify a huge set of inequalities. Fienberg and Meyer [16] proposed to check the convergence of the IPFP, which is not practical since each iterative step requires the execution of $O(|\Omega|)$ arithmetic operations. A linear-programming approach can be found in [37]. Finally, Matúš [48] addressed the problem of the consistency for the special class of graphical distribution sets (i.e., its schemes are conformal hypergraphs), and succeeded in decomposing it using the so-called "canonical triangularization".

Our results on the MCE extension problem generalize previous results on the ME extension problem, which can be viewed as a special instance of the MCE extension problem in the case that each distribution in Q is uniform since, in this case, P is consistent w.r.t. the ME extension of Q if and only if P is consistent and, if P is consistent, then the MCE extension of P w.r.t. the ME extension of Q coincides with the ME extension of P. The ME extension problem has been widely studied in probability theory [10, 22, 23, 37, 47, 49, 58, 59], and in statistics [7, 17, 28]; moreover, tree-computation methods [7, 17, 20, 27, 28, 36, 38] and local-computation methods [3, 38] have been provided to compute the ME extension of a consistent set of distributions.

The paper is organized as follows. In Section 2 we recall some more-or-less standard definitions of hypergraph theory as well as the notions of an acyclic hypergraph, of an acyclic cover of a hypergraph and of a closed vertex set. Section 3 contains basic results on ME and MCE extensions as well as some relational algebra to process supports of distributions. In Section 4 we give a procedure to construct a "tree-representation" of the ME extension of a consistent set of distributions. In

Sections 5, 6 and 7 we apply the divide-and-conquer strategy to solve the consistency problem, the relative-consistency problem and the MCE extension problem, respectively. Section 8 contains some closing notes.

2. HYPERGRAPHS

In this section we recall some more-or-less standard notions and results on hypergraphs [6] which will be used in the sequel.

A *hypergraph* is a finite (possibly empty) set of nonempty sets, which are called the *(hyper)edges* of the hypergraph. The union of edges of a hypergraph is called its *vertex set*. If a hypergraph $\mathbf{A}$ has vertex set X, we say that $\mathbf{A}$ is a hypergraph on X. A hypergraph is *empty* if it has no edges, is *trivial* if it has exactly one edge and is a *graph* if its edges have all size less than 3.

Let $\mathbf{A}$ be a hypergraph on X. Two distinct vertices of $\mathbf{A}$ are *adjacent* (or *neighbours*) if they appear together in some edge of $\mathbf{A}$. A *partial edge* of $\mathbf{A}$ is a nonempty set of vertices that is contained (properly or improperly) in some edge of $\mathbf{A}$. A vertex is a *leaf* if it belongs to exactly one edge, and an edge is *redundant* if it is contained in another edge. A hypergraph is *simple* if no edge is redundant. The *simple reduction* of a hypergraph $\mathbf{A}$ is the simple hypergraph whose edges are exactly the maximal (with respect to set inclusion) edges of $\mathbf{A}$.

A *subhypergraph* of $\mathbf{A}$ is either an empty hypergraph or a hypergraph whose edges are all partial edges of $\mathbf{A}$. If $\mathbf{B}$ is a hypergraph with the same vertex set as $\mathbf{A}$ and $\mathbf{A}$ is a subhypergraph of $\mathbf{B}$, we say that $\mathbf{A}$ is *finer* than $\mathbf{B}$, written $\mathbf{A} \leq \mathbf{B}$. A hypergraph $\mathbf{B}$ is a *cover* of $\mathbf{A}$ if $\mathbf{B}$ is a simple hypergraph and $\mathbf{A}$ is finer than $\mathbf{B}$. Two hypergraphs $\mathbf{A}$ and $\mathbf{B}$ are *equivalent* if $\mathbf{A}$ is finer than $\mathbf{B}$ and $\mathbf{B}$ is finer than $\mathbf{A}$.

A *path* in hypergraph $\mathbf{A}$ is a sequence of distinct edges of $\mathbf{A}$ where every two consecutive edges have a nonempty intersection. Two edges A and B of $\mathbf{A}$ are *connected* if there is a path $(A_1, \ldots, A_l)$ joining them, that is, with $A_1 = A$ and $A_l = B$; analogously, two vertices a and b of $\mathbf{A}$ are *connected* if there is a path $(A_1, \ldots, A_l)$ such that a belongs to A_1 and b belongs to A_l. Finally, $\mathbf{A}$ is *connected* if every two edges (or vertices) are connected. Let $\mathbf{A}$ be a hypergraph on X and let Y be a proper nonempty subset of X. The *subhypergraph* of $\mathbf{A}$ *induced* by Y, denoted by $\mathbf{A}(Y)$, is the hypergraph $\{A \cap Y : A \in \mathbf{A}\} \setminus \{\emptyset\}$. The *connected components* of $\mathbf{A}$ are the subhypergraphs of $\mathbf{A}$ induced by its maximal sets of pairwise connected vertices. Let Y be a proper nonempty subset of X. The *hypergraph* $\mathbf{A} - Y$ is the subhypergraph of $\mathbf{A}$ induced by $X \setminus Y$; the *boundary* of a connected component $\mathbf{A}'$ of $\mathbf{A} - Y$ is the set of vertices in Y that are adjacent to at least one vertex of $\mathbf{A}'$. Two connected vertices of $\mathbf{A}$ are *separated* by Y if they belong to distinct connected components of $\mathbf{A} - Y$. A *minimal vertex separator* (a *separator*, for short) of $\mathbf{A}$ is a partial edge Y of $\mathbf{A}$ such that there exist two connected vertices of $\mathbf{A}$ that are separated by Y and are not separated by any proper subset of Y. An *articulation pair* of $\mathbf{A}$ is a pair $\{A, B\}$ of distinct edges of $\mathbf{A}$ such that $A \cap B$ is a separator of $\mathbf{A}$.

2.1. Acyclic hypergraphs

A hypergraph $\mathbf{A}$ is *acyclic* [5] (or "decomposable" [27]) if either $\mathbf{A}$ consists of a single edge or there exists a *running-intersection ordering* (an RIO, for short) of $\mathbf{A}$, that is, an ordering $(A_1, \ldots, A_n)$ of the edges of $\mathbf{A}$ such that, for each i $(2 \leq i \leq n)$, one has

$$(A_1 \cap \cdots \cap A_{i-1}) \cap A_i \subseteq A_j$$

for some $j < i$. It is well-known [5] that, if $\mathbf{A}$ is an acyclic, simple hypergraph, then each edge of $\mathbf{A}$ is the leading term of some RIO of $\mathbf{A}$. A hypergraph is *cyclic* if it is not acyclic. Let $\mathbf{A}$ be an acyclic, simple and nontrivial hypergraph and let $(A_1, \ldots, A_n)$ be an RIO of $\mathbf{A}$. Let

$$S_i = (A_1 \cap \cdots \cap A_{i-1}) \cap A_i \quad (2 \leq i \leq n).$$

It is well-known [44] that S is a separator of $\mathbf{A}$ if and only if, for some i, $S = S_i$ and $S_i \neq \emptyset$. The *multiplicity* of a separator S of $\mathbf{A}$ is the number of the terms S_i of the list $(S_2, \ldots, S_n)$ such that $S = S_i$. (Note that the multiplicity of a separator of $\mathbf{A}$ is the same for every RIO of $\mathbf{A}$).

Acyclic hypergraphs can be represented by forests of trees. We shall make use of two tree-representations whose definitions are now recalled. Let $\mathbf{A}$ be an acyclic hypergraph with separator set $\mathbf{S}$. Without loss of generality, we assume that $\mathbf{A}$ is a connected, simple hypergraph. A *junction tree* [26] (also called a "join tree" [5, 34]) for $\mathbf{A}$ is a tree T with node set $\mathbf{A}$ (that is, each node of T is an edge of $\mathbf{A}$), where for every two nodes Y and Z, the set $Y \cap Z$ is a subset of each node along the (unique) path joining Y and Z. An efficient procedure for constructing a junction tree can be found in [34]. A *connection tree* (also called an "edge-divider tree" [3]) for $\mathbf{A}$ is a tree T with node set $\mathbf{A} \cup \mathbf{S}$ (that is, each node of T is either an edge of $\mathbf{A}$ or a separator of $\mathbf{A}$), where: (1) each arc has one endpoint in $\mathbf{A}$ and the other in $\mathbf{S}$, and (2) for every two nodes Y and Z, the set $Y \cap Z$ is a subset of each node along the (unique) path joining Y and Z. An efficient procedure for constructing a connection tree can be found in [3]. Let T be a connection tree for $\mathbf{A}$. Henceforth, a node in $\mathbf{A}$ (or in $\mathbf{S}$) is called an *edge-node* (a *separator-node*, respectively) of T. Let S be a separator-node of T; the number of neighbours of S in T minus one comes out to be equal to the multiplicity of S in $\mathbf{A}$ [3].

Example 2.1. The connected and simple hypergraph $\mathbf{A} = \{abc, abd, abe, bf\}$ is acyclic. The separators of $\mathbf{A}$ are ab and b with multiplicities 2 and 1, respectively. Figure 1 shows a junction tree (left) and a connection tree (right) for $\mathbf{A}$.

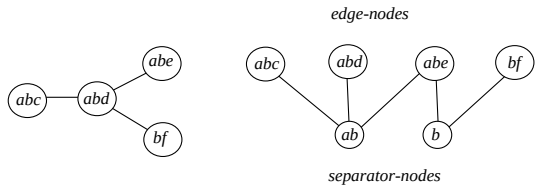


Fig. 1.

The following fact can be easily learned from a connection tree.

Fact 2.2. Let $\mathbf{A}$ be an acyclic, connected hypergraph on X. A subset Y of X is a separator of $\mathbf{A}$ if and only if, for some edge A of $\mathbf{A}$, Y equals the boundary of a connected component of $\mathbf{A} - A$.

Several characterizations of acyclic hypergraphs exist [5], one of which states that a hypergraph $\mathbf{A}$ is acyclic if and only if the following procedure reduces $\mathbf{A}$ to an empty hypergraph.

Algorithm 1. *Graham reduction of a hypergraph*

Input. A hypergraph $\mathbf{A}$.
Output: A subhypergraph of $\mathbf{A}$.

Repeatedly apply the following two operations until neither can be longer applied:

(*vertex removal*)	Delete a vertex if it is a leaf.
(*edge removal*)	Delete a set if it is empty or is a redundant edge. $\mathbf{A}$.

A linear implementation was given in [56]. Note that, if $\mathbf{A}$ is an acyclic and connected, simple hypergraph, then the separators of $\mathbf{A}$ are exactly the sets that are deleted by the edge-removal operation during the Graham reduction of $\mathbf{A}$.

2.2. The compact hypergraph

The *compact hypergraph* of hypergraph $\mathbf{A}$ [42] (also called the "compaction" of $\mathbf{A}$ in [43, 44]) is the simple hypergraph whose edges are the maximal sets of vertices that are separated by no partial edge of $\mathbf{A}$. As noted in [44], the compact hypergraph of $\mathbf{A}$ is exactly the "prime hypergraph" [29] of the clique hypergraph of $\mathbf{A}$.

Example 2.3. The compact hypergraph of the hypergraph $\mathbf{A} = \{ac, bc, abd, abe, ef\}$ has edges abc, abd, abe, ef.

We now recall some nice properties of the compact hypergraph of a hypergraph $\mathbf{A}$ [43, 44].

(K1) The separators of $\mathbf{A}$ and of the compact hypergraph of $\mathbf{A}$ are the same.

(K2) The compact hypergraph of $\mathbf{A}$ is an acyclic cover of $\mathbf{A}$ and is equivalent to $\mathbf{A}$ if and only if $\mathbf{A}$ is acyclic.

(K3) If $\mathbf{A}$ is a simple hypergraph with v vertices and e edges, the compact hypergraph of $\mathbf{A}$ has $O(e)$ edges and can be constructed in $O(ev^3)$ time.

As to (K2), it should be noted that in literature there exist efficient procedures for finding an acyclic cover of $\mathbf{A}$ that is equivalent to $\mathbf{A}$ if and only if $\mathbf{A}$ is acyclic. The most popular algorithms are *zero fill-in algorithms* [56] which are linear in the size of $\mathbf{A}$; we call the resultant acyclic covers of $\mathbf{A}$ *fill-in covers* of $\mathbf{A}$. (Note that, in general, finding a minimum fill-in cover is an NP-complete problem [60].) Using the compact hypergraph of $\mathbf{A}$ and a zero fill-in algorithm, one can find acyclic covers

of $\mathbf{A}$ that are finer than the compact hypergraph of $\mathbf{A}$ and are equivalent to $\mathbf{A}$ if and only if $\mathbf{A}$ is acyclic. Let $\mathbf{K}$ be the compact hypergraph of $\mathbf{A}$. For each $K \in \mathbf{K}$, let $\mathbf{C}^{(K)}$ be a fill-in cover of the subhypergraph $\mathbf{A}(K)$. We call the hypergraph $\mathbf{C} = \cup_{K \in \mathbf{K}} \mathbf{C}^{(K)}$ a *canonical cover* of $\mathbf{A}$. Note that, given a connection tree T for $\mathbf{K}$ and a connection tree $T^{(K)}$ for $\mathbf{C}^{(K)} (K \in \mathbf{K})$, a connection tree for $\mathbf{C}$ can be obtained by replacing each edge-node K of T by $T^{(K)}$.

2.3. Closed sets

Let $\mathbf{A}$ be a hypergraph on X. A subset Y of X is *closed* in $\mathbf{A}$ [40] if the boundary of every connected component of $\mathbf{A} - Y$ is either empty or a partial edge of $\mathbf{A}$. Note that every edge of $\mathbf{A}$ is closed in $\mathbf{A}$. It can be proven [40] that the intersection of every two closed sets is still a closed set so that, given any subset Y of X, there is exactly one minimal (with respect to set inclusion) closed superset of Y, which is called the (*closed hull*) of Y (in $\mathbf{A}$), denoted by $\bar{Y}$. Thus, the family of closed sets defines a convexity space on $\mathbf{A}$ [42]. If $\mathbf{A}$ is an acyclic hypergraph, then the simple reduction of the subhypergraph of $\mathbf{A}$ induced by the hull of Y can be constructed using the following procedure [35], of which a linear implementation was given in [56].

Algorithm 2. *Graham reduction of a hypergraph with a sacred set*

Input. A hypergraph $\mathbf{A}$ on X, and a set Y of (sacred) vertices.
Output: An induced subhypergraph of $\mathbf{A}$.

Repeatedly apply the following two operations until neither can be longer applied:

(*vertex removal*)	Delete a vertex if it is a leaf that does not belong to Y.
(*edge removal*)	Delete a set if it is empty or is a redundant edge.

More in general, if $\mathbf{A}$ is a simple hypergraph then, given the compact hypergraph of $\mathbf{A}$, the hull of Y is the output of the following algorithm which runs in $O(ev)$ time, where e is the number of edges of $\mathbf{A}$ and v is the number of its vertices [40].

Algorithm 3.

Input: A simple hypergraph $\mathbf{A}$, the compact hypergraph $\mathbf{K}$ of $\mathbf{A}$ and a vertex set Y.
Output: A superset $\bar{Y}$ of Y.

(1) Perform the Graham reduction of $\mathbf{K}$ with sacred set Y. Let $\mathbf{G}$ be the resultant hypergraph and let $\bar{Y}$ be the vertex set of $\mathbf{G}$.

(2) For each edge G of $\mathbf{G}$, if G is neither a partial edge of $\mathbf{A}$ nor an edge of $\mathbf{K}$, then set $\bar{Y} = \bar{Y} \cup K$, where K is the edge of $\mathbf{K}$ that contains G.

The following two facts easily follow from Algorithm 3.

Fact 2.4. Let $\mathbf{A}$ be a hypergraph. Every edge of the compact hypergraph of $\mathbf{A}$ is closed in $\mathbf{A}$.

The following is a stronger result than Fact 2.4.

Lemma 2.5. (Malvestuto and Moscarini [44]) The compact hypergraph of a hypergraph $\mathbf{A}$ is the minimal (with respect to refinement) acyclic cover of $\mathbf{A}$ whose edges are all closed in $\mathbf{A}$.

3. DISTRIBUTIONS

In this section we recall some more-or-less standard notions and results on probability distributions and their supports which will be used in the sequel.

Let X be a finite set of discrete variables, each of which has a finite value set. A *state* of X is a value assignment to the variables in X such that the value assigned to a variable is an element of its value set. By Ω we denote the state space of X.

A *distribution* over X is a nonnegative real-valued function p defined on Ω such that $\sum_{x\in\Omega} p(x) = 1$. Let p be a distribution over X; the *support* of p, denoted by $\|p\|$, is the set of states x of X with $p(x) > 0$, and the *entropy* of p is the nonnegative quantity

$$\mathcal{H}(p) = -\sum_{x\in\|p\|} p(x)\log p(x).$$

Let p and q be two distributions over X; p is *dominated* by q (or, equivalently, p is "absolutely continuous" with respect to q) if $\|p\|\subseteq\|q\|$. The *cross-entropy* of p with respect to q is the nonnegative functional

$$\mathcal{D}(p:q) = \begin{cases} \sum_{x\in\|p\|} p(x)\log\dfrac{p(x)}{q(x)} & \text{if } p \text{ is dominated by } q \\ +\infty & \text{else.} \end{cases}$$

Accordingly, if p is dominated by q, then one has

$$\mathcal{D}(p:q) = -\mathcal{H}(p) - \sum_{x\in\|p\|} p(x)\log q(x).$$

Note that, if q is the uniform distribution, then $\|q\| = \Omega$ so that p is definitely dominated by q and $\mathcal{D}(p:q) = \log|\Omega| - \mathcal{H}(p)$. The functional $\mathcal{D}(p:q)$ has a variety of other names (e.g., "I-divergence", "directed divergence", "Kullback-Leibler distance", "relative entropy", "discrimination information").

Let Y be a nonempty proper subset of X. The state space of Y is denoted by $\Omega[Y]$. The *restriction* to Y of a state x of X, written x_Y, is the state of Y for which the variables in Y are assigned the same values as in x. The *marginal* on Y of a distribution p over X, written $p[Y]$, is the distribution over Y defined as follows:

$$p[Y](y) = \sum_{x\in\Omega: x_Y = y} p(x) \qquad (y\in\Omega[Y]).$$

Note that $p(x) > 0$ implies $p[Y](x_Y) > 0$ for every subset Y of X so that one has

$$\|p[Y]\| = \{y\in\Omega[Y] : \exists\, x\in\|p\| \text{ with } x_Y = y\}.$$

We also admit the case $Y = \emptyset$; then, by convention $p[Y]$ is the unity.

Let X be a finite set of discrete variables and let $\mathbf{A}$ be a hypergraph on X. A *probabilistic database* (a *pdb*, for short) over X with *scheme* $\mathbf{A}$ is a set of distributions $P = \{p_A : A \in \mathbf{A}\}$, where p_A is a distribution over A. The pdb P is *simple* if the scheme $\mathbf{A}$ of P is a simple hypergraph. Let Y be a subset of X; the *subdatabase* of P induced by Y, denoted by $P(Y)$, is the set of distributions $p_A[A \cap Y]$ for all A with $A \cap Y \neq \emptyset$. Note that scheme of $P(Y)$ is the induced subhypergraph $\mathbf{A}(Y)$.

3.1. Consistency

Let $P = \{p_A : A \in \mathbf{A}\}$ be a pdb over X. An *extension* of P is a distribution p over X such that $p[A] = p_A$ for all $A \in \mathbf{A}$. The pdb P is *consistent* if there exists an extension of P. Let $\mathbf{A}'$ be the simple reduction of $\mathbf{A}$. The *simple reduction* of P is the simple pdb P' with scheme $\mathbf{A}'$ that consists of distributions in P one for each edge of $\mathbf{A}'$. Of course, the consistency of P' is a necessary but not sufficient condition for the consistency of P.

3.2. Maximum-entropy extension

It is well-known [8] that, if P is a consistent pdb, then there exists exactly one extension p of P such that $\mathcal{H}(p) \geq \mathcal{H}(p')$ for every extension p' of P. This extension p of P will be referred to as the *maximum entropy extension* of P (the *ME extension* of P, for short). As noted in [8], the ME extension of P dominates every extension of P.

Given a hypergraph $\mathbf{A}$ on X, let $I_{\mathbf{A}}$ be the operator that maps every distribution p over X to the ME extension of the (consistent) pdb $\{p[A] : A \in \mathbf{A}\}$. A distribution p over X is a *fixed point* of $I_{\mathbf{A}}$ if $I_{\mathbf{A}}(p) = p$. Of course, if p is the ME extension of a pdb with scheme $\mathbf{A}$, then p is a fixed point of $I_{\mathbf{A}}$.

Fact 3.1. For every hypergraph $\mathbf{A}$ on X, the uniform distribution over X is a fixed point of $I_{\mathbf{A}}$.

Let Y be a subset of X; a fixed point p of $I_{\mathbf{A}}$ is *collapsible* onto Y if $p[Y]$ is a fixed point of $I_{\mathbf{A}(Y)}$ [1].

We now recall three relevant hypergraph-theoretic properties of the operator $I_{\mathbf{A}}$.

Theorem 3.2. (Malvestuto [39]) Let $\mathbf{A}$ and $\mathbf{B}$ be two hypergraphs with the same vertex set. Every fixed point of $I_{\mathbf{A}}$ is also a fixed point of $I_{\mathbf{B}}$ if and only if $\mathbf{A}$ is finer than $\mathbf{B}$.

The next two properties involves collapsibility and decomposability and were proven in [32, 33, 40] and in [17, 37, 48], respectively.

Theorem 3.3. Let $\mathbf{A}$ be a hypergraph on X and let Y be a subset of X. Every fixed point of $I_{\mathbf{A}}$ is collapsible onto Y if and only if Y is closed in $\mathbf{A}$.

Theorem 3.4. Let $\mathbf{A}$ be an acyclic, simple hypergraph on X, let $\mathbf{S}$ be the separator set of $\mathbf{A}$ and, for each separator S of $\mathbf{A}$, let m_S be the multiplicity of S in $\mathbf{A}$. A distribution p is a fixed point of $I_{\mathbf{A}}$ if and only if the factorization

$$p(x) = \frac{\prod_{A\in\mathbf{A}} p[A](x_A)}{\prod_{S\in\mathbf{S}}(p[S](x_S))^{m_S}}$$

holds for every x in the support of p.

In other words, Theorem 3.4 states that, if p is a fixed point of $I_{\mathbf{A}}$, then and only then p has the *closed-form expression*:

$$\frac{\prod_{A\in\mathbf{A}} p[A]}{\prod_{S\in\mathbf{S}}(p[S])^{m_S}}.$$

Note that every connection tree for $\mathbf{A}$ provides a graphical representation of the closed-form expression generated by $\mathbf{A}$. Let T be a connection tree for an acyclic, simple hypergraph $\mathbf{A}$ and let p be a fixed point of $I_{\mathbf{A}}$. Let l be the node labelling of T defined as follows: for each edge-node A, the label of A is $l_A = p[A]$ and, for each separator-node S, the label of S is $l_S = p[S]$. We call the labelled tree (T, l) a *tree-representation* of p *generated* by $\mathbf{A}$.

3.3. Minimum cross-entropy extension

Let P be a (consistent) pdb over X, and let q be a distribution over X. The pdb P is *consistent with respect to* q (q-*consistent*, for short) [24] if there exists an extension of P that is dominated by q. Given a q-consistent pdb P, it is well-known [8] that there exists exactly one extension p of P dominated by q such that $\mathcal{D}(p : q) \leq \mathcal{D}(p' : q)$ for every extension p' of P dominated by q. This extension p of P will be referred to as the *minimum cross-entropy extension* of P with respect to q (the *q-MCE extension* of P, for short). The next theorem is a well-known result [8].

Theorem 3.5. Let $P = \{p_A : A \in \mathbf{A}\}$ be a q-consistent pdb over X. An extension p of P dominated by q is the q-MCE extension of P if and only if, for each $A \in \mathbf{A}$, there exists a real-valued function f_A defined on the state space of A such that the factorization

$$p(x) = q(x) \prod_{A\in\mathbf{A}} f_A(x_A)$$

holds for every x in the support of p

Fact 3.6. If q is strictly positive distribution over X, then every consistent pdb P over X is q-consistent; moreover, if q is the uniform distribution over X, then the q-MCE extension of P coincides with the ME extension of P

Corollary 3.7. Let $P = \{p_A : A \in \mathbf{A}\}$ be a consistent pdb over X. An extension p of P is the ME extension of P if and only if, for each $A \in \mathbf{A}$, there exists a real-valued function f_A defined on the state space of A such that the factorization

$$p(x) = \prod_{A\in\mathbf{A}} f_A(x_A)$$

holds for every x in the support of p.

3.4. Supports as relations

Supports of distributions play a key role in the relative-consistency problem (see the Introduction). In this section, we state some algebraic properties of the support of the ME extension of a consistent pdb. It is natural to view the support of a distribution over a variable set X as a *relation* over X, which is meant to be any subset of the state space Ω of X. We now recall two basic operations of relational algebra [5]: the "projection" of a relation and the "join" of two or more relations.

Let r be a relation over X, and let Y be a nonempty proper subset of X. The *projection* of r onto Y, denoted by $r[Y]$, is the relation over Y defined as follows:

$$r[Y] = \{y \in \Omega[Y] : \exists\, x \in r \text{ with } x_Y = y\}.$$

Note that, if $Y \subseteq Z \subseteq X$, then $(r[Z])[Y] = r[Y]$.

Let r and s be two relations over Y and Z, respectively, let $X = Y \cup Z$ and let Ω be the state space of X. The *(natural) join* of r and s, denoted by $r \bowtie s$, is the relation over X defined as follows:

$$\{x \in \Omega : x_Y \in r \text{ and } x_Z \in s\}.$$

Note that the join operator is associative and commutative so that the join of three or more relations is well-defined.

Let $\mathbf{A}$ be a hypergraph on X. A *relational database* (an *rdb*, for short) *over* X with *scheme* $\mathbf{A}$ is a set of relations $R = \{r_A : A \in \mathbf{A}\}$, where r_A is a relation over A. We denote the join of the relations in R by $\bowtie R$, that is,

$$\bowtie R = \bowtie_{A \in \mathbf{A}} r_A.$$

An *extension* of R is a relation r over X such that $r[A] = r_A$ for all $A \in \mathbf{A}$. An rdb is *consistent* if it admits an extension.

Fact 3.8. If an rdb R is consistent, then the relation $\bowtie R$ is the maximal (with respect to set-inclusion) extension of R.

Using the projection and join operators one can define an operator which is the relational counterpart of the operator $I_{\mathbf{A}}$ (introduced in Subsection 3.2) and, as shown below, enjoys the same hypergraph-theoretic properties as $I_{\mathbf{A}}$.

Given a hypergraph $\mathbf{A}$ on X, let $J_{\mathbf{A}}$ be the operator that maps every relation r over X to the join of the projections of r onto edges of $\mathbf{A}$; that is, $J_{\mathbf{A}}(r) = \bowtie_{A \in \mathbf{A}} r[A]$. A relation r over X is a *fixed point* of $J_{\mathbf{A}}$ if $J_{\mathbf{A}}(r) = r$. Of course, if R is a consistent rdb with scheme $\mathbf{A}$, then the relation $\bowtie R$ is a fixed point of $J_{\mathbf{A}}$.

Let Y be a subset of X; a fixed point r of $J_{\mathbf{A}}$ is *collapsible* onto Y if $r[Y]$ is a fixed point of $J_{\mathbf{A}}(Y)$. The next two results are the relational counterparts of Theorems 3.2 and 3.3 and were stated in [4].

Theorem 3.9. Let $\mathbf{A}$ and $\mathbf{B}$ be two hypergraphs with the same vertex set. Every fixed point of $J_{\mathbf{A}}$ is also a fixed point of $J_{\mathbf{B}}$ if and only if $\mathbf{B}$ is finer than $\mathbf{A}$.

Theorem 3.10. Let $\mathbf{A}$ be a hypergraph on X and Y a subset of X. Every fixed point of $I_{\mathbf{A}}$ is collapsible onto Y if and only if Y is closed in $\mathbf{A}$.

Consider now relations that are supports of distributions.

Fact 3.11. Let p be a distribution over X, and let Y be a nonempty subset of X. The support of the marginal $p[Y]$ of p coincides with the projection onto Y of the support of p, that is, $\|p[Y]\| = \|p\|[Y]$.

Let $P = \{p_A : A \in \mathbf{A}\}$ be consistent pdb. By Fact 3.11, the support of every extension of P is an extension of the rdb $\{\|p_A\|: A \in \mathbf{A}\}$ which, hence, is consistent; moreover, by Fact 3.8, the support of every extension of P (and, hence, of the ME extension of P) is contained in the relation $\bowtie_{A\in\mathbf{A}} \|p_A\|$. However, as stated below [41], if the scheme of P is an acyclic hypergraph, then the support of the ME extension of P equals the join of the supports of the distributions in P.

Theorem 3.12. Let $\mathbf{A}$ be an acyclic hypergraph. The support of every fixed point of $I_{\mathbf{A}}$ is a fixed point of $J_{\mathbf{A}}$.

4. TREE–REPRESENTATION OF A MAXIMUM–ENTROPY EXTENSION

In this section we recall some results on tree-computation of ME extensions. The ME extension of a consistent pdb P over a variable set X can be computed using the Iterative Proportional Fitting Procedure (IPFP) with input the uniform distribution over X. However, if the scheme of P is an acyclic, simple hypergraph, the ME extension of P can be directly computed using its closed-form expression generated by the scheme of P. More in general, by Theorems 3.2 and 3.4 any acyclic cover of the scheme of P generates a closed-form expression (and, hence, a tree-representation) of the ME extension of P. In this section, we make use of the tree-implementation of the IPFP given in [3] to construct tree-representations of the ME extension of P generated by an acyclic cover and by a canonical cover of the scheme of P. We begin by recalling the definition of the *fitting operator* used in the IPFP. Let f be a distribution over X and let g be a distribution over $Y \subseteq X$ that is dominated by $f[Y]$; the result of fitting f to g is the distribution h over X defined as follows

$$h(x) = \begin{cases} \dfrac{g(x_Y)}{f[Y](x_Y)} f(x) & \text{if } x \in \|f\| \\ 0 & \text{else.} \end{cases}$$

As proven in [19], h is dominated by f and $h[Y] = g$. In what follows, as in [19], we denote the result of fitting f to g by $f \triangleright g$.

Let P be a consistent pdb over X with scheme $\mathbf{A}$. Let $(A_1, \ldots, A_n)$ be any ordering of edges of $\mathbf{A}$ and let p_i be the distribution in P over $A_i (1 \leq i \leq n)$. Without loss of generality, we assume that P is a simple pdb (otherwise, we take the simple reduction of P). As recalled in the Introduction, the ME extension p of P can be computed by applying the IPFP to P with prior uniform distribution; that is, p is the limit of the sequence of distributions $p^{(0)}$, $p^{(1)}$, $p^{(2)}$, $\ldots$, where $p^{(0)}$ is the uniform distribution over X and, for $t = rn + i$ with $r \geq 0$ and $1 \leq i \leq n$, $p^{(t)} = p^{(t-1)} \triangleright p_i$.

The following result [20] in some sense generalizes the if-part of Theorem 3.2.

Lemma 4.1. Let P be a consistent pdb with scheme $\mathbf{A}$. If $\mathbf{A}$ is finer than $\mathbf{B}$, then every distribution $p^{(t)}$ is a fixed point of $I_{\mathbf{B}}$.

The following is an immediate consequence of Lemma 4.1.

Theorem 4.2. Let P be a consistent pdb with scheme $\mathbf{A}$. If $\mathbf{A}$ is finer than $\mathbf{B}$, then the ME extension of P is a fixed point of $I_{\mathbf{B}}$.

Let $\mathbf{C}$ be an acyclic cover of $\mathbf{A}$ (e.g., the compact hypergraph of $\mathbf{A}$), let $\mathbf{S}$ be the set of separators of $\mathbf{C}$ and, for each $S \in \mathbf{S}$, let m_S be the multiplicity of S in $\mathbf{C}$. By Lemma 4.1 and Theorem 3.4, every distribution $p^{(t)}$ has the following closed-form expression:

$$\frac{\prod_{C\in\mathbf{C}} p^{(t)}[C]}{\prod_{S\in\mathbf{S}} \left(p^{(t)}[S]\right)^{m_S}}.$$

Therefore, at each step of the IPFP we do not need to compute $p^{(t)}$ but only its marginals $p^{(t)}[C]$ for all $C \in \mathbf{C}$ and its marginals $p^{(t)}[S]$ for all $S \in \mathbf{S}$; moreover, when the convergence is attained, the marginals of the ME extension of P on $C \in \mathbf{C}$ and on $S \in \mathbf{S}$ are available and the ME extension of P can be explicitly computed. (Of course, no computational gain is obtained if $\mathbf{C}$ is a trivial hypergraph.) We now show how to compute the distributions $p^{(t)}[C]$ and $p^{(t)}[S]$ for every t. Of course, each $p^{(0)}[C]$ is the uniform distribution over C and each $p^{(0)}[S]$ is the uniform distribution over S. Consider the case $t > 0$, say $t = rn + i$, $r \geq 0$ and $1 \leq i \leq n$. Recall that $p^{(t)} = p^{(t-1)} \triangleright p_i$. Suppose we are given a tree-representation (T, l) of $p^{(t-1)}$ generated by $\mathbf{C}$. The following algorithm [3] performs a traversal of T and, during the traversal, updates the node labels so that the output is a tree-representation of $p^{(t)}$ generated by $\mathbf{C}$. More efficient propagation procedures can be found in [3, 14].

Markovian propagation algorithm.

1. Find a minimal (w.r.t. set inclusion) node N of T that contains A_i, and set $l_N = l_N \triangleright p_i$.
2. Start a traversal of T at the node N. During the traversal of T,

   when a separator-node S is visited using arc (C, S), set $l_S = l_C[S]$;

   when an edge-node C is visited is visited using arc (S, C), set $l_C = l_C \triangleright l_S$.

To sum up, we have the following procedure for constructing a tree-representation of the ME extension of P generated by an acyclic cover of the scheme of P.

Algorithm 4.

Input: A simple hypergraph $\mathbf{A}$, a consistent pdb $P = \{p_A : A \in \mathbf{A}\}$ over X, an acyclic cover $\mathbf{C}$ of $\mathbf{A}$ and a tree-representation (T, l) of the uniform distribution over X generated by $\mathbf{C}$.
Output: A tree-representation of the ME extension of P generated by $\mathbf{C}$.

Procedure

Until the convergence is attained, repeat:

For each $A \in \mathbf{A}$, apply the Markovian propagation algorithm with input the labelled tree (T, l) and the distribution p_A.

From a computational viewpoint, the cost of Algorithm 4 depends on the acyclic cover $\mathbf{C}$ of $\mathbf{A}$ in use through the number of sets in $\mathbf{C}$ and the maximum size of the state spaces of sets in $\mathbf{C}$. If $\mathbf{C}$ is taken to be a canonical cover of $\mathbf{A}$, we can save storage and time as follows. Recall that $\mathbf{C} = \cup_{K \in \mathbf{K}} \mathbf{C}^{(K)}$ where $\mathbf{K}$ is the compact hypergraph of $\mathbf{A}$ and $\mathbf{C}^{(K)}$ is a fill-in cover of $\mathbf{A}(K)$; moreover, if T is a connection tree for $\mathbf{K}$ and $(T^{(K)}, l^{(K)})$ is a tree-representation of the ME extension of the subdatabase $P(K)$, then a tree-representation of the ME extension of P generated by $\mathbf{C}$ can be obtained by

labelling each separator-node S of T by $p_A[S]$ where A is an edge of $\mathbf{A}$ with minimum-size state space, and replacing each edge-node K of T by $(T^{(K)}, l^{(K)})$.

The resultant labelled tree is actually a tree-representation of the ME extension of P since $\mathbf{K}$ is an acyclic cover of $\mathbf{A}$ whose separators are all partial edges of $\mathbf{A}$ and, for every edge K of $\mathbf{K}$, by Fact 2.4 and Theorem 3.3 the ME extension of P is collapsible onto each K (that is, the ME extension of $P(K)$ equals the marginal on K of the ME extension of P). To sum up, we have the following procedure for constructing a tree-representation of the ME extension of P generated by a canonical cover of the scheme of P.

Algorithm 5.

Input: A simple hypergraph $\mathbf{A}$, a consistent pdb $P = \{p_A : A \in \mathbf{A}\}$, the compact hypergraph $\mathbf{K}$ of $\mathbf{A}$, a connection tree T for $\mathbf{K}$ and, for each edge K of $\mathbf{K}$, a fill-in cover $\mathbf{C}^{(K)}$ of $\mathbf{A}(K)$ and a tree-representation $(T^{(K)}, l^{(K)})$ of the uniform distribution over K generated by $\mathbf{C}^{(K)}$.

Output: A tree-representation of the ME extension of P generated by $\mathbf{C} = \cup_{K \in \mathbf{K}} \mathbf{C}^{(K)}$.

Procedure

(1) For each separator-node S of T, find a minimum-size distribution p_A in P such that A contains S and label the node S by $p_A[S]$.

(2) For each edge-node K of T, apply Algorithm 4 with input the simple reduction of $P(K)$ and $(T^{(K)}, l^{(K)})$ and replace the node K of T by the labelled tree $(T^{(K)}, l^{(K)})$.

5. THE CONSISTENCY PROBLEM

In this section we show that, using the compact hypergraph $\mathbf{K}$ of the scheme $\mathbf{A}$ of a pdb P, the consistency problem can be decomposed into subproblems, one for each edge of $\mathbf{K}$. Moreover, in the case that $\mathbf{A}$ is acyclic, we provide an efficient algorithm to make P consistent.

First of all, observe that a necessary condition for $P = \{p_A : A \in \mathbf{A}\}$ to be consistent is that, for every two edges A and A' of $\mathbf{A}$ with $A \subseteq A'$, one has $p_{A'}[A] = p_A$. If this is the case, we say that P is *projective*. Of course, P is consistent if and only if P is projective and the simple reduction of P is consistent. The projectivity requirement can be tested by pairwise comparisons and the consistency requirement for the simple reduction of P can be tested using a linear constraint system such as system (1) in the Introduction. In what follows, we always assume that P is projective so that we shall limit our considerations to the case that P is a simple pdb. We first discuss the case that $\mathbf{A}$ is acyclic and, then, the general case.

5.1. The acyclic case

A pdb $P = \{p_A : A \in \mathbf{A}\}$ is *pairwise consistent* if, for every two distributions p_A and $p_{A'}$ in P, the distribution pair $\{p_A, p_{A'}\}$ is consistent, that is, for every two distributions p_A and $p_{A'}$ in P, either $A \cap A' = \emptyset$ or $p_A[A \cap A'] = p_{A'}[A \cap A']$. The following is a well-known result [22, 37, 58, 59].

Theorem 5.1. Every pairwise-consistent pdb with an acyclic scheme is consistent.

Corollary 5.2. Let $\mathbf{A}$ be an acyclic, simple hypergraph. A pdb $P = \{p_A : A \in \mathbf{A}\}$ is consistent if and only if, for each articulation pair $\{A, A'\}$ of $\mathbf{A}$, the distribution pair $\{p_A, p_{A'}\}$ is consistent

Proof. (*only if*). Trivial. (*if*) By Theorem 5.1, it is sufficient to prove that, for each edge pair $\{A, A'\}$ with $A \cap A' \neq \emptyset$, the distribution pair $\{p_A, p_{A'}\}$ is consistent. Let T be a connection tree for $\mathbf{A}$. The set $A \cap A'$ is contained in each node of T along the unique (even) path joining A and A', say $(A_1 = A, S_1, A_2, S_2, \ldots, A_k, S_k, A_{k+1} = A')$, and by hypothesis each distribution pair $\{p_{A_h}, p_{A_{h+1}}\}$ is consistent; that is, $p_{A_h}[S_h] = p_{A_{h+1}}[S_h] (1 \leq h \leq k)$. Therefore, since the set $A \cap A'$ is a subset of each S_h for all $(1 \leq h \leq k)$, one has

$$p_{A_h}[A \cap A'] = (p_{A_h}[S_h])[A \cap A'] = (p_{A_{h+1}}[S_h])[A \cap A'] = p_{A_{h+1}}[A \cap A']$$

and, hence,

$$p_{A_1}[A \cap A'] = p_{A_2}[A \cap A'] = \cdots = p_{A_{k+1}}[A \cap A']$$

which proves that the distribution pair $\{p_{A_1}, p_{A_{k+1}}\} = \{p_A, p_{A'}\}$ is consistent. □

By Corollary 5.2, the consistency of P can be tested using the following algorithm, which terminates with the value True of the output variable *test* if and only if P is consistent.

Algorithm 6.

Input: An acyclic, connected hypergraph $\mathbf{A}$, a pdb $P = \{p_A : A \in \mathbf{A}\}$, a connection tree T for $\mathbf{A}$, and a node labelling l of T with $l_A = p_A$ for each edge-node A of T and l_S undefined for each edge-node S.

Output: A truth value of the logical variable *test*.

Procedure

(1) Set $test$ = True.

(2) Perform a traversal of T with start-point an arbitrary edge-node:

when a separator-node S is visited using arc (A, S), set $l_S = p_A[S]$;
when an edge-node A is visited using the arc (S, A), if $l_A[S] \neq l_S$ then set $test$ = False and Exit.

5.2. The general case

In the general case, pairwise consistency is not sufficient for consistency so that, in order to test consistency, one need to check the consistency of the linear constraint system (1) in the Introduction. The following result suggests a method for reducing the size of system (1).

Theorem 5.3. Let P be a pdb with scheme $\mathbf{A}$, and let $\mathbf{C}$ be an acyclic cover of $\mathbf{A}$ whose separators are all partial edges of $\mathbf{A}$. There exists an extension of P if and only if, for every edge C of $\mathbf{C}$, there exists an extension of the subdatabase $P(C)$.

Proof. (*only if*) Let p be an extension of P. For every edge C of $\mathbf{C}$, $p[C]$ is definitely an extension of $P(C)$, which hence is consistent. ((*if*) Let $P = \{p_A : A \in \mathbf{A}\}$. By hypothesis, each subdatabase $P(C)$ is consistent. Consider the pdb $P' = \{p'_C : C \in \mathbf{C}\}$ where p'_C is any extension of $P(C)$. We shall show that

(i) P' is consistent, and

(ii) every extension of P' is also an extension of P,

which will prove that P is consistent.

Proof of (i) Consider any articulation pair $\{C, C'\}$ of $\mathbf{C}$ and let $S = C \cap C'$. Since $P(S)$ is a subdatabase of both $P(C)$ and $P(C')$, $P(S)$ is consistent and both $p'_C[S]$ and $p'_{C'}[S]$ are extensions of $P(S)$. On the other hand, by hypothesis, the separator S of $\mathbf{C}$ is a partial edge of $\mathbf{A}$ so that there exists at least one edge A of $\mathbf{A}$ that contains S and $p_A[S]$ belongs to $P(S)$. It follows that $p_A[S]$ is the only extension of $P(S)$. Since $p'_C[S]$ and $p'_{C'}[S]$ are extensions of $P(S)$, one has that $p'_C[S] = p_A[S]$ and $p'_{C'}[S] = p_A[S]$ which proves that the distribution pair $\{p'_C, p'_{C'}\}$ is consistent. Therefore, since $\mathbf{C}$ is an acyclic hypergraph, P' is consistent by Corollary 5.2.

Proof of (ii). Consider any distribution p_A in P. Since $\mathbf{C}$ is a cover of $\mathbf{A}$, A is contained in some edge of $\mathbf{C}$, say C. It follows that p_A belongs to $P(C)$ and, as p'_C is an extension of $P(C)$, one has $p'_C[A] = p_A$. Let p' be any extension of P'. Since $p'[C] = p'_C$ and $A \subseteq C$, one has $p'[A] = (p'[C])[A] = p'_C[A] = p_A$, which proves that p' is an extension of P. □

Theorem 5.3 provides an effective method to test the consistency of P provided the acyclic cover $\mathbf{C}$ of $\mathbf{A}$ is not a trivial hypergraph. But, what is a choice for $\mathbf{C}$ that reduces the computation to a minimum? By Theorem 5.3, we argue that a "good" choice for $\mathbf{C}$ is an acyclic cover of $\mathbf{A}$ whose separators are partial edges of $\mathbf{A}$ and

that an "optimal" choice is the finest of such acyclic covers of $\mathbf{A}$ (so that the sizes of the subdatabases $P(C)$ for edges C of $\mathbf{C}$ are minimized). The next lemma proves that the compact hypergraph of $\mathbf{A}$ is the optimal choice.

Lemma 5.4. The compact hypergraph of a hypergraph $\mathbf{A}$ is the minimal (with respect to refinement) acyclic cover of $\mathbf{A}$ whose separators are all partial edges of $\mathbf{A}$.

Proof. Let $\mathbf{K}$ be the compact hypergraph of $\mathbf{A}$. Recall that $\mathbf{K}$ is an acyclic cover of $\mathbf{A}$; moreover, by property (K1), every separator of $\mathbf{K}$ is a partial edge of $\mathbf{A}$. Let $\mathbf{C}$ be another acyclic cover of $\mathbf{A}$ whose separators are all partial edges of $\mathbf{A}$. In order to prove that $\mathbf{K} \leq \mathbf{C}$, it is sufficient to show that every edge of $\mathbf{C}$ is closed in $\mathbf{A}$ for, then, the statement follows from Lemma 2.5. Let C be any edge of $\mathbf{C}$. Let $\mathbf{A}'$ be any connected component of $\mathbf{A} - C$ and let Y be the boundary of $\mathbf{A}'$. Since $\mathbf{A} \leq \mathbf{C}$, $\mathbf{A}'$ is a subhypergraph of one connected component of $\mathbf{C} - C$. Let $\mathbf{C}'$ be the connected component of $\mathbf{C} - C$ that contains $\mathbf{A}'$ and let Z be the boundary of $\mathbf{C}'$. Since $\mathbf{A}' \leq \mathbf{C}'$, Y is a subset of Z. On the other hand, by Fact 2.2, Z is a separator of $\mathbf{C}$ and, since every separator of $\mathbf{C}$ is a partial edge of $\mathbf{A}$, Z is a partial edge of $\mathbf{A}$ and, since $Y \subseteq Z$, Y is a partial edge of $\mathbf{A}$ too. The closedness of C in $\mathbf{A}$ follows from the arbitrariness of $\mathbf{A}'$ and C. So, $\mathbf{C}$ is an acyclic cover of $\mathbf{A}$ and every edge of $\mathbf{C}$ is closed in $\mathbf{A}$. □

Given the compact hypergraph $\mathbf{K}$ of $\mathbf{A}$, by Lemma 5.4 and Theorem 5.3 it is sufficient to check the consistency of the subdatabase of P induced by each edge of $\mathbf{K}$, which can be done as follows. We distinguish edges of $\mathbf{K}$ between "simple edges" and "compound edges"; an edge of $\mathbf{K}$ is *simple* if it is also an edge of $\mathbf{A}$, and *compound* otherwise. For each compound edge K of $\mathbf{K}$, we first check the projectivity of $P(K)$ and, then, the consistency of the simple reduction of $P(K)$ using a linear constraint system such as (1); if $P(K)$ is not consistent for some compound edge K of $\mathbf{K}$, then we soon conclude that P is not consistent. Assume that, for each compound edge K of $\mathbf{K}$, $P(K)$ is consistent. Let K be any compound edge of $\mathbf{K}$ and let p_K be an extension of $P(K)$. By the proof of the if-part of Theorem 5.3, one has that, for every simple edge A of $\mathbf{K}$ such that $\{A, K\}$ is an articulation pair of $\mathbf{K}$, the distribution pair $\{p_A, p_K\}$ is consistent. It also follows that, for every two simple edges A and A' of $\mathbf{K}$ such that $\{A, A'\}$ is an articulation pair of $\mathbf{K}$ with $A \cap A' \subseteq K$, the distribution pair $\{p_A, p_{A'}\}$ is consistent. Therefore, what remains to do is to check the consistency of the distribution pair $\{p_A, p_{A'}\}$, for each articulation pair $\{A, A'\}$ of $\mathbf{K}$ where A and A' are simple edges of $\mathbf{K}$ whose intersection is contained in no compound edge of $\mathbf{K}$. To achieve this, we make use of a connection tree T for $\mathbf{K}$. After deleting the edge-nodes of T corresponding to compound edges of $\mathbf{K}$ and their adjacent separator-nodes, we apply Algorithm 5.1 to each connected component of the resultant forest that is not a one-point tree.

Example 5.5. Consider the pdb $P = \{p_A : A \in \mathbf{A}\}$ where $\mathbf{A} = \{ac, bc, abd, abe, ef\}$. The compact hypergraph of $\mathbf{A}$ is $\mathbf{K} = \{abc, abd, abe, ef\}$ (see Example 2.3) and the connection tree for $\mathbf{K}$ is shown in Figure 2. Since abc is the only compound edge of $\mathbf{K}$, we test the consistency of $P(abc) = \{p_{ac}, p_{bc}, p_{abd}[ab], p_{abe}[ab]\}$ as follows. First of all, we check the projectivity of $P(abc)$, that is, whether or not

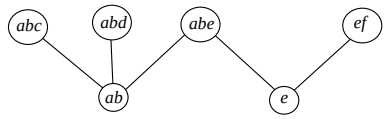


Fig. 2.

$p_{abd}[ab] = p_{abe}[ab]$. If $P(abc)$ is not projective, then we soon conclude that $P(abc)$ is not consistent; otherwise, we check the consistency of a simple reduction of $P(abc)$, e.g. $\{p_{ac}, p_{bc}, p_{abd}[ab]\}$, by finding a solution $p^{(abc)}$ to the following linear constraint system:

$$\begin{cases} p^{(abc)}[ac] & = p_{ac} \\ p^{(abc)}[bc] & = p_{bc} \\ p^{(abc)}[ab] & = p_{abd}[ab] \end{cases}$$

If this constraint system has no solutions, then we conclude that $P(abc)$ is not consistent. Otherwise, $P(abc)$ is consistent and we delete the edge-node abc and the separator-node ab from T. The resultant forest is shown in Figure 3. Finally, we

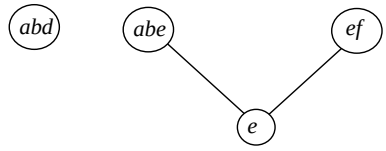


Fig. 3.

apply Algorithm 6 to the labelled tree of Figure 4 and conclude that P is consistent

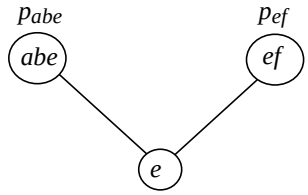


Fig. 4.

if and only if Algorithm 6 terminates with the value True of the output variable *test*.

5.3. Inconsistency

Assume that we are given a simple pdb $P = \{p_A : A \in \mathbf{A}\}$ but we do not know whether or not P is consistent. We now discuss the problem of finding a consistent pdb $\Pi = \{\pi_A : A \in \mathbf{A}\}$ which is an "approximation" to P in that

- $\mathcal{D}(\pi_A : p_A) < \infty$ for all $A \in \mathbf{A}$, and
- $\Pi = P$ if P is consistent.

A different approach to dealing with inconsistency was described in [57]. We only consider the case that the hypergraph $\mathbf{A}$ is acyclic and show that, under suitable conditions (see Theorem 5.6), an easy solution can be found using the following variant of Algorithm 6, where we make use of a connection tree T for $\mathbf{A}$, which is rooted at an edge-node, say N; moreover, as in Algorithm 6, each edge-node A of T is labelled by p_A. It should be noted that the resultant consistent pdb Π is "tuned" to p_N, and that the choice of another edge of $\mathbf{A}$ as root of T may produce another approximation to P.

Algorithm 7.

Input: A connection tree T for $\mathbf{A}$, where each edge-node A of T is labelled by p_A, and an edge-node N.

Output: A consistent pdb $\Pi = \{\pi_A : A \in \mathbf{A}\}$.

Procedure

(1) Set $\pi_N = p_N$.

(2) Perform a traversal of T with start-point N:

when a separator-node S is visited using arc (A, S), set $\pi_S = \pi_A[S]$;
when an edge-node A is visited using arc (S, A), set $\pi_A = p_A \triangleright \pi_S$.

Theorem 5.6. Let $P = \{p_A : A \in \mathbf{A}\}$ be a pdb whose scheme $\mathbf{A}$ is an acyclic hypergraph and let T be a connection tree for $\mathbf{A}$. If N is an edge of $\mathbf{A}$ such that, for every path $(N, \ldots, A', S, A)$ in T, the distribution $p_{A'}[S]$ is dominated by the distribution $p_A[S]$, then Algorithm 7 correctly finds an approximation consistent pdb $\Pi = \{\pi_A : A \in \mathbf{A}\}$ such that

(i) Π is consistent and $\mathcal{D}(\pi_A : p_A) < +\infty$ for all $A \in \mathbf{A}$, and

(ii) $\Pi = P$ if P is consistent.

P r o o f. (i) By Corollary 5.2, Π is a consistent pdb if and only if, for each articulation pair $\{A', A\}$ of $\mathbf{A}$, both $\pi_{A'}$ and π_A are distributions and the distribution pair $\{\pi_{A'}, \pi_A\}$ is consistent. Since π_N is a distribution (see Step 1), it is sufficient to prove that, for every (even) path $(N, \ldots, A', S, A)$, π_A is a distribution, $\{\pi_{A'}, \pi_A\}$ is consistent and $\mathcal{D}(\pi_A : p_A) < +\infty$. The proof is by induction.

BASIS. Consider an edge-node A at distance 2 from N and let (N, S, A) be the path from N to A. By hypothesis, $p_N[S]$ is dominated by $p_A[S]$. When A is visited using arc (S, A), one has that $\pi_N = p_N$ and $\pi_S = \pi_N[S]$. Since $p_N[S]$ is dominated by $p_A[S]$, π_S is dominated by $p_A[S]$ so that the operation $p_A \triangleright \pi_S$ is applied correctly and its result π_A is a distribution; moreover, by Lemma 4.1, the marginal of π_A on S equals π_S ($= \pi_N[S]$) so that the distribution pair $\{\pi_N, \pi_A\}$ is consistent. Finally, by Lemma 4.1, the distribution π_A is dominated by p_A so that $\mathcal{D}(\pi_A : p_A) < +\infty$.

INDUCTION. Consider an edge-node A at distance greater than 2 from N and let $(N, \ldots, A', S, A)$ be the path from N to A. By hypothesis, $p_{A'}[S]$ is dominated by $p_A[S]$. When A is visited using arc (S, A), one has that $\pi_S = \pi_{A'}[S]$ and, by the

inductive hypothesis, $\pi_{A'}$ is dominated by $p_{A'}$. Therefore, π_S is dominated by $p_A[S]$ so that the operation $p_A \triangleright \pi_S$ is applied correctly and its result π_A is a distribution; moreover, by Lemma 4.1, the marginal of π_A on S equals $\pi_S (= \pi_{A'}[S])$ so that the distribution pair $\{\pi_{A'}, \pi_A\}$ is consistent. Finally, by Lemma 4.1, the distribution π_A is dominated by p_A so that $\mathcal{D}(\pi_A : p_A) < +\infty$.

(*ii*) Let A be an edge of $\mathbf{A}$ and let $(N, \ldots, A', S, A)$ be the path from N to A. It is easily seen that $\pi_A = p_A$ if and only if $p_A[S] = \pi_S$, that is, if and only if the distribution pair $\{\pi_{A'}, p_A\}$ is consistent. Therefore, $\pi_A = p_A$ for all A if and only if, for each articulation pair $\{A', A\}$ of $\mathbf{A}$, the distribution pair $\{p_{A'}, p_A\}$ is consistent which, by Corollary 5.2, holds if and only if P is consistent. □

6. THE RELATIVE-CONSISTENCY PROBLEM

Let Q be a consistent pdb over X with scheme $\mathbf{B}$, let q denote the ME extension of Q and let P be a consistent pdb over X with scheme $\mathbf{A}$. In this section, we show that, using a suitable acyclic cover of $\mathbf{A} \cup \mathbf{B}$, the problem of the q-consistency of P can be decomposed into subproblems that can be solved locally. We begin by stating a result which, in some sense, generalizes Theorem 5.3.

Theorem 6.1. Let P and Q be two consistent pdb's over the same variable set and with schemes $\mathbf{A}$ and $\mathbf{B}$, respectively. Let $\mathbf{C}$ be an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ whose separators (if any) are all partial edges of $\mathbf{A}$. There exists an extension of P that is dominated by the ME extension of Q if and only if, for every edge C of $\mathbf{C}$, there exists an extension of the subdatabase $P(C)$ that is dominated by the marginal on C of the ME extension of Q.

Proof. Let q be the ME extension of Q.

(*only if*) Let p be an extension of P that is dominated by q; thus, $\|p\| \subseteq \|q\|$. For each $C \in \mathbf{C}$, $p[C]$ is definitely an extension of $P(C)$; moreover, since $\|p\| \subseteq \|q\|$, one has $\|p\|[C] \subseteq \|q\|[C]$. By Fact 3.11, $\|p\|[C] = \|p[C]\|$ and $\|q\|[C] = \|q[C]\|$ so that $\|p[C]\| \subseteq \|q[C]\|$ which proves that $p[C]$ is dominated by $q[C]$.

(*if*) By hypothesis, for each $C \in \mathbf{C}$ there exists an extension p'_C of $P(C)$ dominated by $q[C]$. In order to prove that P is q-consistent, we show that

(*i*) the pdb $P' = \{p'_C : C \in \mathbf{C}\}$ is consistent,

(*ii*) every extension of P' is also an extension of P, and

(*iii*) the ME extension of P' is dominated by q.

After doing that, we will have that the ME extension of P' is an extension of P that is dominated by q, which proves the statement.

The proofs of (*i*) and (*ii*) are the same as in Theorem 5.3.

Proof of (*iii*). Since q is a fixed point of $I_{\mathbf{B}}$ and $\mathbf{B} \leq \mathbf{A} \cup \mathbf{B} \leq \mathbf{C}$, q is a fixed point of $I_{\mathbf{C}}$ by Theorem 3.2. Moreover, since $\mathbf{C}$ is an acyclic hypergraph, by Theorem 3.9 the support of q is a fixed point of $J_{\mathbf{C}}$ so that, by Fact 3.11, one has

$$\|q\| = \bowtie_{C \in \mathbf{C}} \|q\|[C].$$

Since p'_C is dominated by $q[C]$ and since, by Fact 3.11, the support of $q[C]$ is equal to $\|q\|[C]$, one has

$$\|p'_C\| \subseteq \|q\|[C_j]$$

Let p' be the ME extension of P'. Since p' is a fixed point of $I_{\mathbf{C}}$ and $\mathbf{C}$ is an acyclic hypergraph, by Theorem 3.9 the support of p' is a fixed point of $J_{\mathbf{C}}$ so that, by Fact 3.11, one has

$$\|p'\| = \bowtie_{C \in \mathbf{C}} \|p'_C\|.$$

Therefore, one has

$$\|p'\| = \bowtie_{C \in \mathbf{C}} \|p'_C\| \subseteq \bowtie_{C \in \mathbf{C}} \|q\|[C] = \|q\|$$

which proves that p' is dominated by q. □

In order to apply Theorem 6.1 we need: (1) a procedure to construct an acyclic cover $\mathbf{C}$ of $\mathbf{A} \cup \mathbf{B}$ whose separators (if any) are all partial edges of $\mathbf{A}$, and (2) a procedure to compute the support of the marginal of the ME extension of Q on every edge of $\mathbf{C}$. In the next two subsections, we deal with tasks (1) and (2), respectively.

6.1. Task (1)

We now give a simple procedure to construct an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ whose separators (if any) are all partial edges of $\mathbf{A}$. We start with the compact hypergraph of $\mathbf{A} \cup \mathbf{B}$, say $\mathbf{H}$. By property (K1) of $\mathbf{H}$ (see Section 2.2), the separators of $\mathbf{H}$ are all partial edges of $\mathbf{A} \cup \mathbf{B}$, that is, they are partial edges of $\mathbf{A}$ or $\mathbf{B}$. Let T be a connection tree for $\mathbf{H}$ where the separator-nodes that are partial edges of $\mathbf{A}$ are marked (see Figure 5). Let $T_1, \ldots, T_n$ be the connected components of the

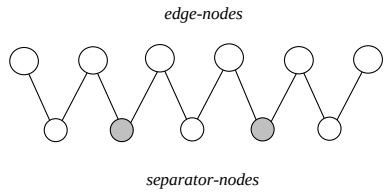


Fig. 5.

forest obtained from T by deleting all marked separator-nodes and their incident arcs (see Figure 6). Let us denote the union of nodes of T_i by $K_i (1 \leq i \leq n)$, and

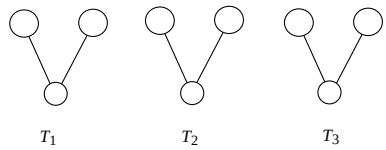


Fig. 6.

let $\mathbf{K} = \{K_1, \ldots, K_n\}$. Of course, $\mathbf{K}$ is an acyclic hypergraph and a connection tree for $\mathbf{K}$ can be obtained from T by replacing each subtree T_i by a single edge-node K_i. Moreover, as $\mathbf{A} \cup \mathbf{B} \leq \mathbf{H} \leq \mathbf{K}$, one has that $\mathbf{K}$ is a cover of $\mathbf{A} \cup \mathbf{B}$. Finally, by construction, each separator of $\mathbf{K}$ is a partial edge of $\mathbf{A}$.

Remark 6.2. The procedure above still works if $\mathbf{H}$ is any acyclic cover of $\mathbf{A} \cup \mathbf{B}$ whose edges are all partial edges of $\mathbf{A} \cup \mathbf{B}$. However, by Lemma 5.4 taking $\mathbf{H}$ to be the compact hypergraph of $\mathbf{A} \cup \mathbf{B}$ proves to be a good choice.

Of course, if the resultant hypergraph $\mathbf{K}$ is a trivial hypergraph, no computational gain is obtained.

Example 6.3. Consider the two cyclic (hyper)graphs $\mathbf{A} = \{ab, ad, bc, be, cf, de, ef\}$ and $\mathbf{B} = \{ab, ad, ae, ce, cf, de, ef\}$ (see Figure 7). The compact hypergraph of $\mathbf{A} \cup \mathbf{B}$

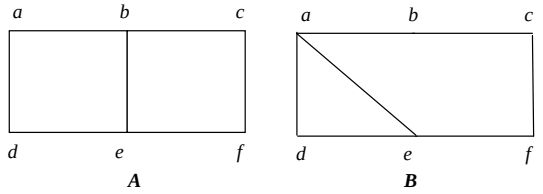


Fig. 7.

is $\mathbf{H} = \{abe, ade, bcef\}$; its separators are ae and be, which are edges of $\mathbf{B}$ and $\mathbf{A}$, respectively. Figure 8 shows the connection tree for $\mathbf{H}$ where the separator-node be is marked. The hypergraph $\mathbf{K} = \{abde, bcef\}$ is an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ and its

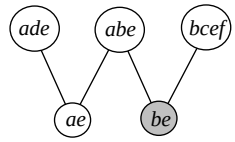


Fig. 8.

separator (be) is a (partial) edge of $\mathbf{A}$. Note that (by chance) $\mathbf{K}$ equals the compact hypergraph of $\mathbf{A}$.

6.2. Task (2)

After accomplishing task (1) with output the acyclic cover $\mathbf{K}$ of $\mathbf{A} \cup \mathbf{B}$, what remains to do is the computation of the support of $q[K]$ for every edge K of $\mathbf{K}$, where q denotes the ME extension of Q. A simple way consists in first constructing a tree-representation (T, l) of q generated by $\mathbf{K}$ using Algorithm 4 and, then, taking the supports of the labels of edge-nodes of T. As said above, the costs of Algorithm 4 depends on the maximum size of the state spaces of sets in $\mathbf{K}$. A more efficient method makes use of a canonical cover of $\mathbf{B}$, say $\mathbf{C}$. Explicitly, if $\mathbf{J}$ is the compact hypergraph of $\mathbf{B}$ and, for each edge J of $\mathbf{J}$, $\mathbf{C}^{(J)}$ is a fill-in cover of $\mathbf{B}(J)$, then

$\mathbf{C} = \cup_{J \in \mathbf{J}} \mathbf{C}^{(J)}$. Suppose we have already constructed a tree-representation (T, l) of q generated by $\mathbf{C}$ using Algorithm 5 with input $\mathbf{B}$, Q, $\mathbf{J}$ and $\mathbf{C}^{(J)}$ for each edge J of $\mathbf{J}$. Consider the rdb $R = \{r_C : C \in \mathbf{C}\}$, where r_C is the support of the label of edge-node C of T. Since $\mathbf{C}$ is an acyclic cover of $\mathbf{B}$, the support of q is the join of the relations r_C in R. By Fact 3.11, for each $K \in \mathbf{K}$, one has

$$\|q[K]\| = (\bowtie R)[K]. \tag{2}$$

It is not convenient to compute $\|q[K]\|$ using (2) because the right-hand side of (2) can be reduced as follows. Let $\bar{K}$ be the hull of K in $\mathbf{C}$. Since $\bar{K}$ is closed in $\mathbf{C}$, by Theorem 3.10 the relation $(\bowtie R)$ is collapsible onto $\bar{K}$, that is, $(\bowtie R)[\bar{K}] = \bowtie R(\bar{K})$ where $R(\bar{K})$ is the subdatabase of R induced by $\bar{K}$. Therefore, one has

$$\|q[K]\| = (\bowtie R)[K] = ((\bowtie R)[\bar{K}])[K] = (\bowtie R(\bar{K}))[K]. \tag{3}$$

Consider now the relation $\bowtie R(\bar{K})$. Note that, owing to the decomposability of $\mathbf{C}$, $\bar{K}$ equals the vertex set of the hypergraph $\mathbf{G}$ resulting from the Graham reduction of $\mathbf{C}$ with sacred set K. Recall that $\mathbf{G}$ is the simple reduction of the subhypergraph of $\mathbf{C}$ induced by $\bar{K}$. Therefore, every edge of $\mathbf{G}$ is contained in some edge of $\mathbf{C}$, and, hence, each relation in $R(\bar{K})$ can be obtained by projection of some relation r_C in R. Explicitly, for every edge G of $\mathbf{G}$, let $r_{C(G)}$ be a minimum-size relation among the relations r_C in R for which C contains G. Then, one has

$$\bowtie R(\bar{K}) = \bowtie_{G \in \mathbf{G}} r_{C(G)}[G].$$

Finally, by substituting this expression into the right-hand side of (3) one has the following relational expression:

$$\left(\bowtie_{G \in \mathbf{G}} r_{C(G)}[G]\right)[K] \tag{4}$$

which is simpler than the right-hand side of (2). Finally, we can evaluate (4) using the following algorithm [61], which consists in pruning a junction tree T for $\mathbf{G}$ rooted at an arbitrary edge-node. Initially, each edge-node G of T is labelled by the relation $r_G = r_{C(G)}[G]$.

Algorithm 8. *Pruning algorithm*

1. Until T is a one-point tree, repeat:

   Find a leaf L of T.

   Let G be the parent of L.
   Replace the label of G by the relation $(r_G \bowtie r_L)[G \cup (L \cap K)]$ (see Figure 9).
   Delete the leaf L.

2. Output the relation labelling the unique node of T.

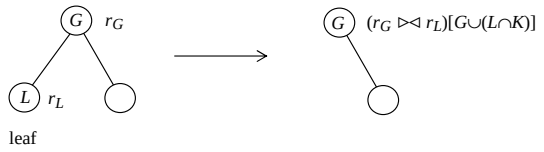


Fig. 9.

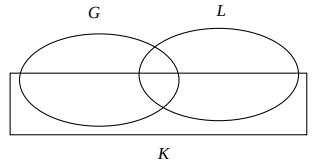


Fig. 10.

Remark 6.4. In each pruning step, one can reduce the cost of label updating by taking

$$r_G \bowtie r_L[L \cap (K \cup G)]$$

which is the same as the relation $(r_G \bowtie r_L)[G \cup (L \cap K)]$ (see Figure 10).

Example 6.5. Let $P = \{p_A : A \in \mathbf{A}\}$ and $Q = \{q_B : B \in \mathbf{B}\}$ be two consistent pdb's, where $\mathbf{A}$ and $\mathbf{B}$ are the hypergraphs of Example 6.3. Let q denote the ME extension of Q. In order to test the q-consistency of P we make use of the acyclic cover $\mathbf{K} = \{abde, bcef\}$ of $\mathbf{A} \cup \mathbf{B}$ from task (1) (see Example 6.3). In order to apply Theorem 6.1, we need to compute the supports of the marginals of q on the two edges of $\mathbf{K}$. To achieve this, suppose we make use of the canonical cover $\mathbf{C} = \{abf, ade, aef, bcf\}$ of $\mathbf{B}$ and that, after applying Algorithm 5, we have the relations $r_{abf} = \|q[abf]\|$, $r_{ade} = \|q[ade]\|$, $r_{aef} = \|q[aef]\|$ and $r_{bcf} = \|q[bcf]\|$. The supports of the marginals of q on the edges $abde$ and $bcef$ of $\mathbf{K}$ are computed as follows.

When we perform the Graham reduction of $\mathbf{C}$ with sacred set $abde$, we obtain $G = \{abf, ade, aef\}$. The junction tree for G rooted at the node ade is shown in Figure 12 (a). Using the pruning algorithm and Remark 6.4, we find that the support

Fig. 12.

of $q[abde]$ is given by

$$r_{ade} \bowtie ((r_{aef} \bowtie r_{abf})[abe]).$$

When we perform the Graham reduction of $\mathbf{C}$ with sacred set $bcef$, we obtain $\mathbf{G} = \{abf, aef, bcf\}$. The junction tree for $\mathbf{G}$ rooted at the node bcf is shown in Figure 12 (b). Using the pruning algorithm and Remark 6.4, we find that the support of $q[bcef]$ is given by

$$r_{bcf} \bowtie ((r_{abf} \bowtie r_{aef})[bef]).$$

7. THE MINIMUM CROSS-ENTROPY EXTENSION PROBLEM

Let Q be a consistent pdb with scheme $\mathbf{B}$, let q denote the ME extension of Q, let P be a pdb with scheme $\mathbf{A}$ that is q-consistent, let $(A_1, \ldots, A_n)$ be any ordering of edges of $\mathbf{A}$ and let p_i be the distribution in P over $A_i (1 \leq i \leq n)$. As recalled in the Introduction, the q-MCE extension p of P can be computed by applying the IPFP to P with prior q; that is, p is the limit of the sequence of distributions $p^{(0)}$, $p^{(1)}$, $p^{(2)}$, $\ldots$, where $p^{(0)} = q$ and, for $t = rn + i$ ($r \geq 0$ and $1 \leq i \leq n$), $p^{(t)} = p^{(t-1)} \triangleright p_i$. In Subsection 7.1 we prove that p has a tree-representation generated by any acyclic cover $\mathbf{C}$ of the hypergraph $\mathbf{A} \cup \mathbf{B}$, and show how to construct such a tree-representation of p. In Subsection 7.2, we show that, under suitable conditions, the tree-representation generated by $\mathbf{C}$ can be constructed by local computation.

7.1. Tree-representation of an MCE extension

The following generalizes Lemma 4.1.

Lemma 7.1. Let P be a q-consistent pdb with scheme $\mathbf{A}$. If q is a fixed point of $I_{\mathbf{B}}$ and $\mathbf{A} \cup \mathbf{B}$ is finer than $\mathbf{C}$, then every distribution $p^{(t)}$ is a fixed point of $I_{\mathbf{C}}$.

P r o o f. Let $P = \{p_A : A \in \mathbf{A}\}$. We prove the statement by induction on t.

BASIS ($t = 0$). Since $p^{(0)} = q$ and q is a fixed point of $I_{\mathbf{B}}$ and $\mathbf{B} \leq \mathbf{A} \cup \mathbf{B} \leq \mathbf{C}$, by Theorem 3.2 $p^{(0)}$ is a fixed point of $I_{\mathbf{C}}$.

INDUCTIVE STEP Assume that $p^{(t-1)}$ is a fixed point of $I_{\mathbf{C}}$ for $t > 0$. By Corollary 3.7, for every $C \in \mathbf{C}$ there exists a function f_C such that the factorization

$$p^{(t-1)}(x) = \prod_{C \in \mathbf{C}} f_C(x_C)$$

holds for every $x \in \|p^{(t-1)}\|$. Since $p^{(t)} = p^{(t-1)} \triangleright p_A$, the following factorization

$$p^{(t)}(x) = \frac{p_A(x_A)}{p^{(t-1)}[A](x_A)} \prod_{C \in \mathbf{C}} f_C(x_C)$$

holds for every $x \in \|p^{(t-1)}\|$ and, since $\|p^{(t)}\| \subseteq \|p^{(t-1)}\|$, it holds for every $x \in \|p^{(t)}\|$. Moreover, since $\mathbf{A} \leq \mathbf{A} \cup \mathbf{B} \leq \mathbf{C}$, A is a partial edge of $\mathbf{C}$. Let C be an edge of $\mathbf{C}$ that contains A, and let

$$f'_C(x_C) = \frac{p_A(x_A)}{p^{(t-1)}[A](x_A)} f_C(x_C)$$

Then, the above factorization of $p^{(t)}$ can be re-written as

$$p^{(t)}(x) = f'_C(x_C) \prod_{C' \in \mathbf{C} \setminus \{C\}} f_{C'}(x_{C'})$$

so that, by Corollary 3.7, $p^{(t)}$ is a fixed point of $I_{\mathbf{C}}$. □

The following is an immediate consequence of Lemma 7.1.

Theorem 7.2. Let P be a q-consistent pdb with scheme $\mathbf{A}$. If q is a fixed point of $I_{\mathbf{B}}$ and $\mathbf{A} \cup \mathbf{B}$ is finer than $\mathbf{C}$, then the q-MCE of P is a fixed point of $I_{\mathbf{C}}$.

By Theorem 7.2, if $\mathbf{C}$ is an acyclic cover of $\mathbf{A} \cup \mathbf{B}$, then $\mathbf{C}$ can be used to construct a tree-representation the q-MCE of P as follows. Since $\mathbf{B} \leq \mathbf{A} \cup \mathbf{B} \leq \mathbf{C}$, $\mathbf{C}$ generates a tree-representation of the ME extension q of Q which can be constructed using Algorithm 4 with input Q and a tree-representation (T, l) of the uniform distribution generated by $\mathbf{C}$. By Theorem 7.2, a tree-representation of the q-MCE of P generated by $\mathbf{C}$ can be obtained using Algorithm 4 with input P and the output (T, l) of the previous application of Algorithm 4.

Algorithm 9.

Input: A consistent pdb $Q = \{q_B : B \in \mathbf{B}\}$ over X, a q-consistent pdb $P = \{p_A : A \in \mathbf{A}\}$ over X where q denotes the ME extension of Q, an acyclic cover $\mathbf{C}$ of $\mathbf{A} \cup \mathbf{B}$ and a tree-representation (T, l) of the uniform distribution over X generated by $\mathbf{C}$.

Output: A tree-representation tree of the q-MCE extension of P generated by $\mathbf{C}$.

Procedure

(1) Apply Algorithm 4 to Q.

(2) Apply Algorithm 4 to P.

Example 7.3. Let $P = \{p_A : A \in \mathbf{A}\}$ and $Q = \{q_B : B \in \mathbf{B}\}$ be two consistent pdb's, where $\mathbf{A} = \{ab, ad, bc, be, cf, de, ef\}$ and $\mathbf{B} = \{ab, ae, bc, de, ef\}$ (see Figure 13). Let q denote the ME extension of Q and let p denote the the q-MCE of P.

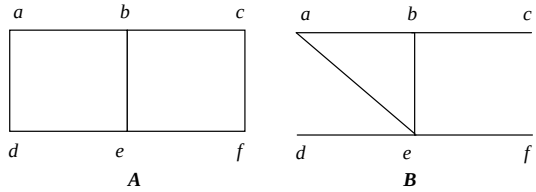


Fig. 13.

An acyclic cover of $\mathbf{A} \cup \mathbf{B}$ is $\mathbf{C} = \{abe, ade, bce, cef\}$. Using Algorithm 4 with input Q and a tree-representation (T, l) of the uniform distribution generated by $\mathbf{C}$ we can construct a tree-representation of q generated by $\mathbf{C}$. At this point, using Algorithm 4 with input P and (T, l) we can construct a tree-representation of p generated by $\mathbf{C}$.

7.2. Local computation

In this subsection, we show that the problem of finding a tree-representation of the MCE extension can be decomposed into subproblems that can be solved locally.

Lemma 7.4. Let P and Q be two consistent pdb's over the same variable set and with schemes $\mathbf{A}$ and $\mathbf{B}$, respectively. Assume that P is q-consistent, where q denotes the ME extension of Q, and let p denote the q-MCE extension of P. Let $\mathbf{K}$ be an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ whose separators are all partial edges of $\mathbf{A}$. For every edge K of $\mathbf{K}$, $p[K]$ is the $q[K]$-MCE extension of $P(K)$.

Proof. First of all, observe that, since every separator of $\mathbf{K}$ is a partial edge of $\mathbf{A}$, every separator of $\mathbf{K}$ is also a partial edge of $\mathbf{A} \cup \mathbf{B}$. So, $\mathbf{K}$ is an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ and every separator of $\mathbf{K}$ is a partial edge of $\mathbf{A} \cup \mathbf{B}$. Without loss of generality, we can assume that $\mathbf{K}$ is a connected hypergraph. Let K be any edge of $\mathbf{K}$. Of course, $p[K]$ is an extension of $P(K)$. In order to prove that $p[K]$ is the $q[K]$-MCE extension of $P(K)$, by Theorem 3.5 it is sufficient to show that, for each $A' \in \mathbf{A}(K)$ there exists a real-valued function $g_{A'}$ defined on the state space of A' such that

$$p[K] = q[K] \prod_{A' \in \mathbf{A}(K)} g_{A'}.$$

To achieve this, we introduce a suitable acyclic cover $\mathbf{H}$ of $\mathbf{K}$ such that K is an edge of $\mathbf{H}$ and each separator of $\mathbf{H}$ is a separator of $\mathbf{K}$. The hypergraph $\mathbf{H}$ is constructed as follows. Let T be a connection tree for $\mathbf{K}$. Assume that the edge-node K has n neighbours, say the separator-nodes $S_1, \ldots, S_n$. Let $T_1, \ldots, T_n$ be the connected components of the forest obtained from T by deleting the edge-node K and the arcs $(K, S_1), \ldots, (K, S_n)$ (see Figure 14). Let H_i be the union of the labels of nodes of

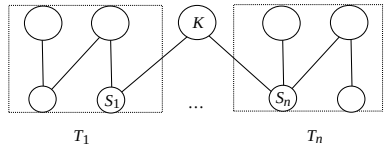


Fig. 14.

T_i $(1 \leq i \leq n)$, and let $\mathbf{H} = \{K, H_1, \ldots, H_n\}$. Of course, $\mathbf{H}$ is an acyclic hypergraph, the separators of $\mathbf{H}$ are $S_1, \ldots, S_n$ and the multiplicity of each S_i in $\mathbf{H}$ is one. A connection tree for $\mathbf{H}$ is shown in Figure 15. Since $\mathbf{B} \leq \mathbf{A} \cup \mathbf{B} \leq \mathbf{K} \leq \mathbf{H}$ and $\mathbf{H}$ is

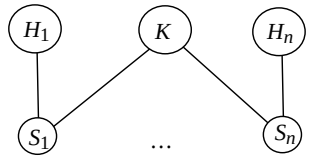


Fig. 15.

an acyclic, simple hypergraph, by Theorem 3.4 one has

$$q = q[K] \prod_{1 \le i \le n} \frac{q[H_i]}{q[S_i]}.$$

On the other hand, by Theorem 3.5 one has that, for each $A \in \mathbf{A}$, there exists a real-valued function f_A defined on the state space of A such that

$$p = q \prod_{A \in \mathbf{A}} f_A$$

so that

$$p = q[K] \prod_{1 \le i \le n} \frac{q[H_i]}{q[S_i]} \left(\prod_{A \in \mathbf{A}} f_A \right).$$

Since $\mathbf{A} \le \mathbf{A} \cup \mathbf{B} \le \mathbf{K} \le \mathbf{H}$, there is a many-to-one mapping α from $\mathbf{A}$ to $\mathbf{H}$ such that $A \subseteq \alpha(A)$. Using the mapping α, the above expression of p can be re-written as:

$$p = q[K] \left(\prod_{1 \le i \le n} \frac{q[H_i] \prod_{A:\alpha(A)=H_i} f_A}{q[S_i]} \right) \left(\prod_{A \subseteq K} f_A \right).$$

At this point, we marginalize p on K by summing out the variables $a \notin K$. We now prove that $a \notin K$ if and only if there exists exactly one H_i that contains a so that $a \in H_i \setminus S_i$. Of course, if $a \in H_i \setminus S_i$ then $a \notin K$. On the other hand, let A be an edge of $\mathbf{A}$ containing a and let i be such that $\alpha(A) = H_i$. Since $a \notin K$ and each S_i is contained in K, one has that $a \notin S_i$ for all i and, hence, H_i is the only edge of $\mathbf{H}$ containing a so that $a \in H_i \setminus S_i$. It follows that the marginal p on K has the following expression:

$$p = q[K] \left(\prod_{1 \le i \le n} \frac{\sum_{a \in H_i \setminus S_i} \left(q[H_i] \prod_{A:\alpha(A)=H_i} f_A \right)}{q[S_i]} \right) \left(\prod_{A \subseteq K} f_A \right)$$

which can be re-written as

$$p = q[K] \left(\prod_{1 \le i \le n} g_i \right) \left(\prod_{A \subseteq K} f_A \right)$$

where

$$g_i = \frac{\sum_{a \in H_i \setminus S_i} \left(q[H_i] \prod_{A:\alpha(A)=H_i} f_A \right)}{q[S_i]}$$

is a function defined on the state space of S_i. Let $\mathbf{A}' = \{A \in \mathbf{A} : A \subseteq K\} \cup \{S_1, \ldots, S_n\}$. At this point, it is sufficient to show that the hypergraph $\mathbf{A}'$ is equivalent to $\mathbf{A}(K)$. We first show $\mathbf{A}' \le \mathbf{A}(K)$ and, then, $\mathbf{A}(K) \le \mathbf{A}'$.

Proof of $\mathbf{A}' \le \mathbf{A}(K)$. Of course, each edge A of $\mathbf{A}$ such that $A \subseteq K$ is also an edge of $\mathbf{A}(K)$. Consider now any $S_i (1 \le i \le n)$. Since, by construction, S_i is also a

separator of $\mathbf{K}$ and, by hypothesis, each separator of $\mathbf{K}$ is a partial edge of $\mathbf{A}$, one has that S_i is a partial edge of $\mathbf{A}$. Let A be an edge of $\mathbf{A}$ that contains S_i. Since $S_i \subseteq A$ and $S_i \subseteq K$, one also has $S_i \subseteq A \cap K$; but, the set $A \cap K$ is an edge of $\mathbf{A}(K)$ so that S_i is a partial edge of $\mathbf{A}(K)$. Therefore, one has $\mathbf{A}' \leq \mathbf{A}(K)$.

Proof of $\mathbf{A}(K) \leq \mathbf{A}'$. Consider any edge A' of $\mathbf{A}(K)$. Then, there exists and edge A of $\mathbf{A}$ such that $A' = A \cap K$. Now, if $A \subseteq K$, then $A' = A$ and, hence, A' is also an edge of $\mathbf{A}'$. Otherwise, let i be such that $\alpha(A) = H_i$. Then, one has $A' = A \cap K \subseteq H_i \cap K = S_i$ and, hence, A' is a partial edge of $\mathbf{A}'$. Therefore, one has $\mathbf{A}(K) \leq \mathbf{A}'$.

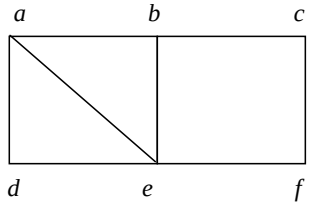
To sum up, the hypergraphs $\mathbf{A}(K)$ and $\mathbf{A}'$ are equivalent, which completes the proof. □

Theorem 7.5. Let P and Q be two consistent pdb's over the same variable set and with schemes $\mathbf{A}$ and $\mathbf{B}$, respectively. Assume that P is q-consistent, where q denotes the ME extension of Q, and let p denote the q-MCE extension of P. Let $\mathbf{K}$ be an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ whose separators are all partial edges of $\mathbf{A}$ and whose edges are closed in $\mathbf{B}$. For every edge K of $\mathbf{K}$, $p[K]$ is the MCE extension of $P(K)$ with respect to the ME extension of $Q(K)$.

Proof. Since each edge of $\mathbf{K}$ is closed in $\mathbf{B}$, by Theorem 3.3 the marginal $q[K]$ coincides with the ME extension of $Q(K)$ and the statement follows from Lemma 7.4. □

A simple procedure to construct an acyclic cover $\mathbf{K}$ of $\mathbf{A} \cup \mathbf{B}$ whose separators are all partial edges of $\mathbf{A}$ and whose edges are all closed in $\mathbf{B}$ is a variant of the procedure given in Subsection 6.1 with the only difference that in the connection tree for the compact hypergraph of $\mathbf{A} \cup \mathbf{B}$ a separator-node is marked if it is a partial edge of both $\mathbf{A}$ and $\mathbf{B}$. The resultant hypergraph is then an acyclic cover $\mathbf{K}$ of $\mathbf{A} \cup \mathbf{B}$ whose separators are all partial edges of both $\mathbf{A}$ and $\mathbf{B}$ and, by the proof of Lemma 5.4, each edge of $\mathbf{K}$ is closed in $\mathbf{B}$.

Example 7.6. Consider the hypergraphs $\mathbf{A}$ and $\mathbf{B}$ of Example 7.3. The hypergraph $\mathbf{A} \cup \mathbf{B}$ is shown in Figure 16. The compact hypergraph of $\mathbf{A} \cup \mathbf{B}$ is the hypergraph



$\boldsymbol{A \cup B}$

Fig. 16.

$\{abe, ade, bcef\}$ and its separators are ae and be. Only be is a (partial) edge of both $\mathbf{A}$ and $\mathbf{B}$. The hypergraph $\mathbf{K} = \{abde, bcef\}$ is an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ whose separator is a (partial) edge of $\mathbf{A}$ and whose edges are both closed in $\mathbf{B}$.

Let $\mathbf{K}$ be an acyclic cover of $\mathbf{A} \cup \mathbf{B}$ constructed as above, and let $\mathbf{C} = \cup_{K \in \mathbf{K}} \mathbf{C}^{(K)}$ where $\mathbf{C}^{(K)}$ is a canonical cover of $\mathbf{A}(K) \cup \mathbf{B}(K)$. Let T be a connection tree for $\mathbf{K}$ and let $T^{(K)}$ be a connection tree for $\mathbf{C}^{(K)} (K \in \mathbf{K})$. Note that a connection tree for $\mathbf{C}$ can be obtained from replacing each edge-node K of T by $T^{(K)}$. The following algorithm, which echoes Algorithm 5, constructs a tree-representation of the q-MCE extension p of P generated by $\mathbf{C}$ by first labelling each separator-node of T and, then, by replacing each edge-node K of T by a tree-representation $(T^{(K)}, l^{(K)})$ of $p[K]$ which, by Theorem 7.5, is constructed using Algorithm 9.

Algorithm 10.

Input: A consistent pdb $Q = \{q_B : B \in \mathbf{B}\}$ over X, a q-consistent pdb $P = \{p_A : A \in \mathbf{A}\}$ over X where q denotes the ME extension of Q, an acyclic cover $\mathbf{K}$ of $\mathbf{A} \cup \mathbf{B}$ whose separators are all partial edges of $\mathbf{A}$ and whose edges are closed in $\mathbf{B}$, a connection tree T for $\mathbf{K}$ and, for each edge K of $\mathbf{K}$, a fill-in cover $\mathbf{C}^{(K)}$ of $\mathbf{A}(K)$ and a tree-representation $(T^{(K)}, l^{(K)})$ of the uniform distribution over K generated by $\mathbf{C}^{(K)}$.

Output: A tree-representation of the q-MCE extension of P generated by $\mathbf{C} = \cup_{K \in \mathbf{K}} \mathbf{C}^{(K)}$.

Procedure

(1) For each separator-node S of T do:

Find a minimum-size distribution p_A in P such that A contains S and label the node S by $p_A[S]$.

(2) For each edge-node K of T do:

Apply Algorithm 9 with input $Q(K)$, $P(K)$ and $(T^{(K)}, l^{(K)})$;
replace the node K of T by the labelled tree $(T^{(K)}, l^{(K)})$.

Example 7.7. Consider the hypergraphs $\mathbf{A}$ and $\mathbf{B}$ of Example 7.3, and the hypergraph $\mathbf{K}$ of Example 7.6. The connection tree T for $\mathbf{K} = \{abde, bcef\}$ is shown in Figure 17. A canonical cover of $\mathbf{A}(abde) \cup \mathbf{B}(abde)$ is $\mathbf{C}^{(abde)} = \{abe, ade\}$ and

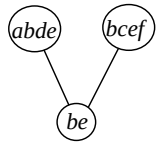


Fig. 17.

a canonical cover of $\mathbf{A}(bcef) \cup \mathbf{B}(bcef)$ is $\mathbf{C}^{(bcef)} = \{bce, cef\}$. The connection trees $T^{(abde)}$ and $T^{(bcef)}$ for $\mathbf{C}^{(abde)}$ and $\mathbf{C}^{(bcef)}$ are shown in Figure 18. At Step 1 of Algorithm 10 we label the separator-node of T by p_{be}. At Step 2 of Algorithm 10, the two edges of $\mathbf{K}$ are processed as follows. ($abde$) We first construct a tree-representation $(T^{(abde)}, l^{(abde)})$ of $p[abde]$ using Algorithm 9 with input $Q(abde)$, $P(abde)$ and a tree-representation of the uniform distribution generated by $\mathbf{C}^{(abde)}$

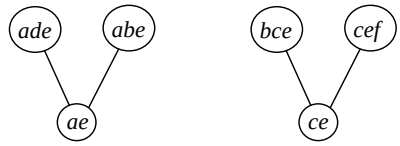


Fig. 18.

and, then, replace the edge-node $abde$ of T by $(T^{(abde)}, l(abde))$. $(bcef)$ We first construct a tree-representation $(T(bcef), l(bcef))$ of $p[bcef]$ using Algorithm 9 with input $Q(bcef)$, $P(bcef)$ and a tree-representation of the uniform distribution generated by $\mathbf{C}^{(bcef)}$ and, then, replace the edge-node $bcef$ of T by $(T^{(bcef)}, l^{(bcef)})$.

8. CONCLUSIONS

We have examined three problems (consistency, relative consistency and numeric computation) related to cross-entropy minimization under marginality constraints and, in order to reduce the computational effort, for each of them we have provided a decomposition into subproblems that can be solved locally and whose solutions can be merged together to get the global solution. To achieve this, we have exploited graphical properties of the schemes of the involved pdb's, which make themselves conspicuous when such schemes are viewed as hypergraphs as proven by existing work on entropy maximization and graphical models. We intend to try our techniques on realistic examples borrowed from probabilistic databases or expert systems. An open problem is how to extend them to more general cross-entropy minimization problems where constraints do not consist of marginality only.

(Received August 19, 2009)

REFERENCES

[1] S. Asmussen and D. Edwards: Collapsibility and response variables in contingency tables. Biometrika *70* (1983), 367–378.

[2] M. Bacharach: Biproportional Matrices and Input-Output Change. Cambridge University Press, Cambridge 1970.

[3] J.-H. Badsberg and F. M. Malvestuto: An implementation of the iterative proportional fitting procedure by propagation trees. Comput. Statist. Data Analysis *37* (2001), 297–322.

[4] C. Beeri and M. Vardi: On the Properties of Full Join Dependencies. Adv. Database Theory I, Plenum Press, New York 1981.

[5] C. Beeri, R. Fagin, D. Maier, and M. Yannakakis: On the desirability of acyclic database schemes. J. Assoc. Comput. Mach. *30* (1983), 479–513.

[6] C. Berge: Hypergraphs. North-Holland, Amsterdam 1989.

[7] C. Berge: Discrete Multivariate Analysis. MIT Press, Cambridge 1975.

[8] I. Csiszár: I-divergence geometry of probability distributions and minimization problems. Ann. Probab. *3* (1975), 146–158.

[9] I. Csiszár: Maxent, mathematics, and information theory. In: Proc. Internat. Workshop on "Maximum entropy and Bayesian methods", 1995, pp. 35–50.

[10] G. Dall'Aglio, K. Kotz, and G. Salinetti (eds.): Advances in Probability Distributions with Given Marginals. Kluwer Academic Pub., Dordrecht, Boston, London 1991.

[11] J. N. Darroch and D. Ratcliff: Generalized iterative scaling for log-linear models. Ann. Math. Statist. *43* (1972), 1470–1480.

[12] W. E. Deming: Statistical Adjustment of Data. Dover Pub., New York 1943.

[13] W. E. Deming and F. F. Stephan: On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. Ann. Math. Statist. *11* (1940), 427–444.

[14] Y. Endo and A. I. Takemura: Iterative proportional scaling via decomposable submodels for contingency tables. Comput. Statist. Data Analysis *53* (2009), 966–978.

[15] S. E. Fienberg: An iterative procedure for estimation in contingency tables. Ann. Math. Statist. *41* (1970), 907–917.

[16] S. E. Fienberg and M. M. Meyer: Iterative proportional fitting. In: Encyclopedia of Statistical Sciences (S. Kotz, N. L. Johnson, and C. B. Read, eds.), *4*, John Wiley and Sons, New York, pp. 275–279.

[17] S. J. Haberman: Log-linear Models for Contingency Tables. University of Chicago Press, Chicago 1974.

[18] C. T. Ireland and S. Kullback: Contingency tables with given marginals. Biometrika *55* (1968), 179–188.

[19] R. Jiroušek: Composition of probability measures on finite spaces. In: Proc. XIII Internat. Conf. Uncertainty in Artificial Intelligence 1997, pp. 274–281.

[20] R. Jiroušek and S. Přeučil: On the effective implementation of the iterative proportional fitting procedure. Comput. Statist. Data Analysis *19* (1995), 177–189.

[21] R. W. Johnson: Axiomatic characterization of the directed divergences and their linear combinations. IEEE Trans. Inform. Theory *25* (1979), 709–716.

[22] H. G. Kellerer: Verteilungsfunktionen mit gegebenen marginalverteilungen. Zeitschrift Wahrscheinlichkeitstheorie und Verw. Gebiete *3* (1964), 247–270.

[23] H. G. Kellerer: Masstheoretische marginalprobleme. Math. Annalen *153* (1964), 168–198.

[24] G. Kern-Isberner: Characterizing the principle of minimum-cross entropy within a conditional-logical framework. Artificial Intelligence *98* (1998), 169–208.

[25] H. H. Ku and S. Kullback: Interaction in multidimensional contingency tables: an information-theoretic approach. J. Res. Nat. Bur. Standards - Math. Sci. *72 B* (1968), 159–199.

[26] S. L. Lauritzen: Graphical Models. Oxford Science Pub., Clarendom Press, Oxford 1996.

[27] S. L. Lauritzen, M. P. Speed and K. Vijayan: Decomposable graphs and hypergraphs. J. Austral. Math. Soc. Ser. *A 36* (1984), 12–29.

[28] S. L. Lauritzen and D. J. Spiegelhalter: Local computations with probabilities on graphical structures and their application to expert systems. J. Roy. Stat. Soc. Ser. *B 50* (1988), 157–224.

[29] G. Leimer: Optimal decomposition by clique separators. Discrete Math. *113* (1993), 99–123.

[30] W. W. Leontief: The Structure of American Economy 1919–1929. Oxford University Press, New York 1941.

[31] W. W. Leontief and A. Strout: Multiregional input-output analysis. In: Structural Interdependence and Economic Development, 1963, pp. 119–169.

[32] D. Madigan and K. Mosurski: An extension of the results of Asmussen and Edwards on collapsibility in contingency tables. Biometrika *77* (1990), 315–319.

[33] D. Madigan and K. Mosurski: Errata: An extension of the results of Asmussen and Edwards on collapsibility in contingency tables. Biometrika *86* (1999) 973.

[34] D. Maier: The Theory of Relational Databases. Computer Science Press, 1983. (http://web.cecs.pdx.edu/ maier/TheoryBook/TRD.html)

[35] D. Maier and J. D. Ullman: Connections in acyclic hypergraphs. Theoret. Comput. Sci. *32* (1984), 185–199.

[36] F. M. Malvestuto: Answering queries in categorical data bases. In: Proc. VI ACM Symp. Principles of Database Systems 1987, pp. 87–96.

[37] F. M. Malvestuto: Existence of extensions and product extensions for discrete probability distributions. Discrete Math. *69* (1988), 61–77.

[38] F. M. Malvestuto: Computing the maximum-entropy extension of given discrete probability distributions. Computat. Statist. Data Anal. *8* (1989), 299–311.

[39] F. M. Malvestuto: Testing implication of hierarchical loglinear models for discrete probability distributions. Statist. Computing *6* (1996), 169–176.

[40] F. M. Malvestuto: A hypergraph-theoretic analysis of collapsibility and decomposability for extended loglinear models. Statist. Computing *11* (2001), 155–169.

[41] F. M. Malvestuto: From conditional independences to factorization constraints with discrete random variables. Ann. Math. Artificial Intelligence *35* (2002), 253–285.

[42] F. M. Malvestuto: Canonical and monophonic convexities in hypergraphs. Discrete Math. *309* (2009), 4287–4298.

[43] F. M. Malvestuto and M. Moscarini: A fast algorithm for query optimization in universal-relation databases. J. Comput. System Sci. *56* (1998), 299–309.

[44] F. M. Malvestuto and M. Moscarini: Decomposition of a hypergraph by partial-edge separators. Theoret. Comput. Sci. *237* (2000), 57–79.

[45] F. M. Malvestuto and E. Pourabbas: Customized answers to summary queries via aggregate views. In: Proc. XVI Intl. Conf. Scientific & Statistical Database Management 2004, pp. 193–202.

[46] F. M. Malvestuto and E. Pourabbas: Local computation of answers to table queries on summary databases. In: Proc. XVII Intl. Conf. Scientific & Statistical Database Management 2005, pp. 263–272.

[47] F. Matúš: Discrete marginal problem for complex measures. Kybernetika *24* (1988), 36–46.

[48] F. Matúš: On the maximum-entropy extensions of probability measures over undirected graphs. In: Proc. III Workshop Uncertainty Processing in Expert Systems 1994, pp. 181–198.

[49] F. Matúš and J. Flusser: Image representations via a finite Radon transform. IEEE Trans. Pattern Analysis and Machine Intelligence *15* (1993), 996–1006.

[50] N. J. Purcell and L. Kish: Estimation for small domains. Biometrics *35* (1979), 365–384.

[51] N. J. Purcell and L. Kish: Postcensal estimates for local areas (or domains). Internat. Statist. Rev. *48* (1980), 3–18.

[52] L. Rüschendorf: Convergence of the iterative proportional fitting procedure. Ann. Statist. *23* (1995), 1160–1174.

[53] J. E. Shore and R. W. Johnson: Properties of cross-entropy minimization. IEEE Trans. Inform. Theory *27* (1981), 472–482.

[54] F. F. Stephan: An iterative method of adjusting sample frequencies tables when expected marginal totals are known. Ann. Math. Statist. *13* (1942), 166–178.

[55] R. Stone and A. Brown: A Computable Model for Economic Growth: A Programme for Growth, No. 1. Chapman Hall, London 1962.

[56] R. E. Tarjan and M. Yannakakis: Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce hypergraphs. SIAM J. Comput. *13* (1984), 566–579.

[57] J. Vomlel: Integrating inconsistent data in a probabilistic model. J. Appl. Non-Classical Logics *14* (2004), 365–386.

[58] N. N. Vorob'ev: Consistent families of measures and their extensions. Theor. Prob. Appl. *7* (1962), 147–163.

[59] N. N. Vorob'ev: Markov measures and Markov extensions. Theor. Prob. Appl. *8* (1963), 420–429.

[60] M. Yannakakis: Computing the minimum fill-in is NP-complete. SIAM J. Algebraic Discrete Mathematics *2* (1981), 77–79.

[61] M. Yannakakis: Algorithms for acyclic database schemes. In: Proc. VII Internat. Conf. Very Large Data Bases 1981, pp. 82–94.

Francesco M. Malvestuto, Department of Informatics, Faculty of Math. Phys. and Nat. Sciences, Sapienza University of Rome, Via Salaria 113, 00198 Rome. Italy.

e-mail: malvestuto@di.uniroma1.it