Binarization Algorithms for Approximate Updating in Credal Nets

Alessandro Antonucci [a,1], Marco Zaffalon [a], Jaime S. Ide [b] and Fabio G. Cozman [c]

[a] *Istituto Dalle Molle di Studi sull'Intelligenza Artificiale*
Galleria 2 - 6928 Manno (Lugano), Switzerland
[b] *Escola de Economia de São Paulo, Fundaçao Getulio Vargas*
Rua Itapeva, 474 - São Paulo, SP - Brazil
[c] *Escola Politecnica, Universidade de São Paulo*
Av. Prof. Mello Moraes, 2231 - São Paulo, SP - Brazil

Abstract. Credal networks generalize Bayesian networks relaxing numerical parameters. This considerably expands expressivity, but makes belief updating a hard task even on polytrees. Nevertheless, if all the variables are binary, polytree-shaped credal networks can be efficiently updated by the 2U algorithm. In this paper we present a *binarization algorithm*, that makes it possible to approximate an updating problem in a credal net by a corresponding problem in a credal net over binary variables. The procedure leads to outer bounds for the original problem. The binarized nets are in general multiply connected, but can be updated by the *loopy* variant of 2U. The quality of the overall approximation is investigated by promising numerical experiments.

Keywords. Belief updating, credal networks, 2U algorithm, loopy belief propagation.

1. Introduction

Bayesian networks (Section 2.1) are probabilistic graphical models based on precise assessments for the conditional probability mass functions of the network variables given the values of their parents. As a relaxation of such precise assessments, *credal networks* (Section 2.2) only require the conditional probability mass functions to belong to convex sets of mass functions, i.e., *credal sets*. This considerably expands expressivity, but makes also considerably more difficult to *update* beliefs about a queried variable given evidential information about some other variables: while in the case of Bayesian network, efficient algorithms can update polytree-shaped models [10], in the case of credal networks updating is NP-hard even on polytrees [4]. The only known exception to this situation is 2U [5], an algorithm providing exact posterior beliefs on *binary* (i.e., such that all the variables are binary) polytree-shaped credal networks in linear time. The topology of the network, which is assumed to be singly connected, and the number of possible states for the variables, which is limited to two for any variable, are therefore

[1]Correspondence to: Alessandro Antonucci, IDSIA, Galleria 2 - CH-6928 Manno (Lugano), Switzerland. Tel.: +41 58 666 66 69; Fax +41 58 666 66 61; E-mail: alessandro@idsia.ch.

the main limitations faced by 2U. The limitation about topology is partially overcome: the *loopy* variant of 2U (L2U) can be employed to update multiply connected credal networks [6] (Section 3). The algorithm typically converges after few iterations, providing an approximate but accurate method to update binary credal nets of arbitrary topology.

The goal of this paper is to overcome also the limitation of 2U about the number of possible states. To this extent, a map is defined to transform a generic updating problem on a credal net into a second updating problem on a corresponding binary credal net (Section 4). The transformation can be implemented in efficient time and the posterior probabilities in the *binarized* network are shown to be an outer approximation of those of the initial problem. The binarized network, which is multiply connected in general, is then updated by L2U. The quality of the approximation is tested by numerical simulations, for which good approximations are obtained (Section 5). Conclusions and outlooks are in Section 6, while some technical parts conclude the paper in Appendix A.

2. Bayesian and Credal Networks

In this section we review the basics of Bayesian networks (BNs) and their extension to convex sets of probabilities, i.e., credal networks (CNs). Both the models are based on a collection of random variables, structured as an array $\mathbf{X} = (X_1, \ldots, X_n)$, and a directed acyclic graph (DAG) $\mathcal{G}$, whose nodes are associated with the variables of $\mathbf{X}$. In our assumptions the variables in $\mathbf{X}$ take values in finite sets. For both the models, we assume the *Markov condition* to make $\mathcal{G}$ to represent probabilistic independence relations between the variables in $\mathbf{X}$: every variable is independent of its non-descendant non-parents conditional on its parents. What makes BNs and CNs different is a different notion of independence and a different characterization of the conditional mass functions for each variable given the values of the parents, which will be detailed next.

Regarding notation, for each $X_i \in \mathbf{X}$, $\Omega_{X_i} = \{x_{i0}, x_{i1}, \ldots, x_{i(d_i-1)}\}$ denotes the set of the possible states of X_i, $P(X_i)$ is a mass function for X_i and $P(x_i)$ the probability that $X_i = x_i$, where x_i is a generic element of Ω_{X_i}. A similar notation with uppercase subscripts (e.g., X_E) denotes arrays (and sets) of variables in $\mathbf{X}$. Finally, the parents of X_i, according to $\mathcal{G}$, are denoted by Π_i, while for each $\pi_i \in \Omega_{\Pi_i}$, $P(X_i|\pi_i)$ is the mass function for X_i conditional on $\Pi_i = \pi_i$.

2.1. Bayesian Networks

In the case of BNs, a conditional mass function $P(X_i|\pi_i)$ for each $X_i \in \mathbf{X}$ and $\pi_i \in \Omega_{\Pi_i}$ should be defined; and the standard notion of probabilistic independence is assumed in the Markov condition. A BN can therefore be regarded as a joint probability mass function over $\mathbf{X}$, that, according the Markov condition, factorizes as follows:

$$P(\mathbf{x}) = \prod_{i=1}^{n} P(x_i|\pi_i), \tag{1}$$

for all the possible values of $\mathbf{x} \in \Omega_{\mathbf{X}}$, with the values of x_i and π_i consistent with $\mathbf{x}$. In the following, we represent a BN as a pair $\langle \mathcal{G}, P(\mathbf{X}) \rangle$. Concerning updating, posterior beliefs about a queried variable X_q, given evidence $X_E = x_E$, are computed as follows:

$$P(x_q|x_E) = \frac{\sum_{x_M} \prod_{i=1}^{n} P(x_i|\pi_i)}{\sum_{x_M,x_q} \prod_{i=1}^{n} P(x_i|\pi_i)}, \tag{2}$$

where $X_M \equiv \mathbf{X} \setminus (\{X_q\} \cup X_E)$, the domains of the arguments of the sums are left implicit and the values of x_i and π_i are consistent with $\mathbf{x} = (x_q, x_M, x_E)$. The evaluation of Equation (2) is a NP-hard task [1], but in the special case of polytree-shaped BNs, Pearl's propagation scheme based on propagated local messages allows for efficient update [10].

2.2. Credal Sets and Credal Networks

CNs relax BNs by allowing for *imprecise probability* statements: in our assumptions, the conditional mass functions of a CN are just required to belong to a finitely generated *credal set*, i.e., the convex hull of a finite number of mass functions over a variable. Geometrically, a credal set is a *polytope*. A credal set contains an infinite number of mass functions, but only a finite number of *extreme mass functions*: those corresponding to the *vertices* of the polytope, which are, in general, a subset of the generating mass functions. It is possible to show that updating based on a credal set is equivalent to that based only on its vertices [11]. A credal set over X will be denoted as $K(X)$.

In order to specify a CN over the variables in $\mathbf{X}$ based on $\mathcal{G}$, a collection of conditional credal sets $K(X_i|\pi_i)$, one for each $\pi_i \in \Omega_{\Pi_i}$, should be provided separately for each $X_i \in \mathbf{X}$; while, regarding Markov condition, we assume *strong independence* [2]. A CN associated to these local specifications is said to be with *separately specified* credal sets. In this paper, we consider only CNs with separately specified credal sets. The specification becomes global considering the *strong extension* of the CN, i.e.,

$$K(\mathbf{X}) \equiv \mathrm{CH}\left\{ \prod_{i=1}^{n} P(X_i|\Pi_i) : P(X_i|\pi_i) \in K(X_i|\pi_i) \quad \begin{array}{l} \forall \pi_i \in \Omega_{\Pi_i}, \\ \forall i = 1, \ldots, n \end{array} \right\}, \tag{3}$$

where CH denotes the convex hull of a set of functions. In the following, we represent a CN as a pair $\langle \mathcal{G}, \mathbf{P}(\mathbf{X}) \rangle$, where $\mathbf{P}(\mathbf{X}) = \{P_k(\mathbf{X})\}_{k=1}^{n_v}$ denotes the set of the vertices of $K(\mathbf{X})$, whose number is assumed to be n_v. It is an obvious remark that, for each $k = 1, \ldots, n_v$, $\langle \mathcal{G}, P_k(\mathbf{X}) \rangle$ is a BN. For this reason a CN can be regarded as a finite set of BNs. In the case of CNs, updating is intended as the computation of tight bounds of the probabilities of a queried variable, given some evidences, i.e., Equation (2) generalizes as:

$$\underline{P}(x_q|x_E) = \min_{k=1,\ldots,n_v} \frac{\sum_{x_M} \prod_{i=1}^{n} P_k(x_i|\pi_i)}{\sum_{x_M,x_q} \prod_{i=1}^{n} P_k(x_i|\pi_i)}, \tag{4}$$

and similarly with a maximum replacing the minimum for upper probabilities $\overline{P}(x_q|x_E)$. Exact updating in CNs displays high complexity: updating in polytree-shaped CNs is NP-complete, and $\mathrm{NP}^{\mathrm{PP}}$-complete in general CNs [4]. The only known exact linear-time algorithm for updating a specific class of CNs is the 2U algorithm, which we review in the following section.

3. The 2U Algorithm and its Loopy Extension

The extension to CNs of Pearl's algorithm for efficient updating on polytree-shaped BNs faced serious computational problems. To solve Equation (2), Pearl's propagation scheme computes the joint probabilities $P(x_q, x_E)$ for each $x_q \in \Omega_{X_q}$; the conditional probabilities associated to $P(X_q|x_E)$ are then obtained using the normalization of this mass function. Such approach cannot be easily extended to Equation (4), because $\underline{P}(X_q|x_E)$ and $\overline{P}(X_q|x_E)$ are not normalized in general.

A remarkable exception to this situation is the case of *binary* CNs, i.e., models for which all the variables are binary. The reason is that a credal set over a binary variable has at most two vertices and can therefore be identified with an interval. This makes possible an efficient extension of Pearl's propagation scheme. The result is an exact algorithm for polytree-shaped binary CNs, called *2-Updating* (2U), whose computational complexity is linear in the input size.

Loosely speaking, 2U computes lower and upper messages for each node according to the same propagation scheme of Pearl's algorithm but with different combination rules. Any node produces a local computation and the global computation is concluded updating all the nodes in sequence. See [5] for a detailed description of 2U.

Loopy propagation is a popular technique that applies Pearl's propagation to multiply connected BNs [9]: propagation is iterated until probabilities converge or for a fixed number of iterations. In a recent paper [6], Ide and Cozman extend these ideas to belief updating on CNs, by developing a loopy variant of 2U (L2U) that makes 2U usable for multiply connected binary CNs.

Initialization of variables and messages follows the same steps used in the 2U algorithm. Then nodes are repeatedly updated following a given sequence. Updates are repeated until convergence of probabilities is observed or until a maximum number of iterations is reached. Concerning computational complexity, L2U is basically an iteration of 2U and its complexity is therefore linear in the number input size and in the number of iterations. Overall, the L2U algorithm is fast and returns good results, with low errors after a small number of iterations [6, Sect. 6]. However, at the present moment, there are no theoretical guarantees about convergence.

Briefly, L2U overcomes 2U limitations about topology, at the cost of an approximation; and in the next section we show how to make it bypass also the limitations about the number of possible states.

4. Binarization Algorithms

In this section, we define a procedure to map updating problems in CNs into corresponding problems in binary CNs. To this extent, we first show how to represent a random variable as a collection of binary variables (Section 4.1). Secondly, we employ this idea to represent a BN as an equivalent binary BN (Section 4.3) with an appropriate graphical structure (Section 4.2). Finally, we extend this binarization procedure to the case of CNs (Section 4.4).

4.1. Binarization of Variables

Assume d_i, which is the number of states for X_i, to be an integer power of two, i.e., $\Omega_{X_i} = \{x_{i0}, \ldots, x_{i(d_i-1)}\}$, with $d_i = 2^{m_i}$ and m_i integer. An obvious one-to-one correspondence between the states of X_i and the joint states of an array of m_i binary variables $(B_{i(m_i-1)}, \ldots, B_{i1}, B_{i0})$ can be established: we assume that the joint state $(b_{i(m_i-1)}, \ldots, b_{i0}) \in \{0,1\}^{m_i}$ is associated to $x_{il} \in \Omega_{X_i}$, where l is the integer whose m_i-bit binary representation is the sequence $b_{i(m_i-1)} \cdots b_1 b_0$. We refer to this procedure as the *binarization* of X_i and the binary variable B_{ij} is said to be the *j-th order bit* of X_i. As an example, the state x_{i6} of X_i, assuming for X_i eight possible values, i.e., $m_i = 3$, would be represented by the joint state $(1,1,0)$ for the three binary variables (B_{i2}, B_{i1}, B_{i0}).

If the number of states of X_i is not an integer power of two, the variable is said to be *not binarizable*. In this case we can make X_i binarizable simply adding to Ω_{X_i} a number of *impossible states*[1] up the the nearest power of two. For example we can make binarizable a variable with six possible values by adding two impossible states. Clearly, once the variables of $\mathbf{X}$ have been made binarizable, there is an obvious one-to-one correspondence between the joint states of $\mathbf{X}$ and those of the array of the binary variables returned by the binarization of $\mathbf{X}$, say $\tilde{\mathbf{X}} = (B_{1(m_1-1)}, \ldots, B_{10}, B_{2(m_2-1)}, \ldots, B_{n(m_n-1)}, \ldots, B_{n0})$. Regarding notation, for each $\mathbf{x} \in \Omega_{\mathbf{X}}$, $\tilde{\mathbf{x}}$ is assumed to denote the corresponding element of $\Omega_{\tilde{\mathbf{X}}}$ and *vice versa*. Similarly, $\tilde{x}_E$ denotes the joint state for the bits of the nodes in X_E corresponding to x_E.

4.2. Graph Binarization

Let $\mathcal{G}$ be a DAG associated to a set of binarizable variables $\mathbf{X}$. We call the *binarization* of $\mathcal{G}$ with respect to $\mathbf{X}$, a second DAG $\tilde{\mathcal{G}}$ associated to the variables $\tilde{\mathbf{X}}$ returned by the binarization of $\mathbf{X}$, obtained with the following prescriptions: (i) two nodes of $\tilde{\mathcal{G}}$ corresponding to bits of different variables in $\mathbf{X}$ are connected by an arc if and only if there is an arc with the same orientation between the relative variables in $\mathbf{X}$; (ii) an arc connects two nodes of $\tilde{\mathcal{G}}$ corresponding to bits of the same variable of $\mathbf{X}$ if and only if the order of the bit associated to the node from which the arc departs is lower than the order of the bit associated to the remaining node.

Figure 1 reports a multiply connected DAG $\mathcal{G}$ and its binarization $\tilde{\mathcal{G}}$. As an example of Prescription (i) for $\tilde{\mathcal{G}}$, note the arcs connecting all the three bits of X_0 with all the two bits of X_2, while, considering the bits of X_0, the arcs between the bit of order zero and those of order one and two, as well as that between the bit of order one and that of order two, are drawn because of Prescription (ii).

4.3. Bayesian Networks Binarization

The notion of binarizability extends to BNs as follows: $\langle \mathcal{G}, P(\mathbf{X}) \rangle$ is *binarizable* if and only if $\mathbf{X}$ is a set of binarizable variables. A non-binarizable BN can be made binarizable by the following procedure: (i) make the variables in $\mathbf{X}$ binarizable; (ii) specify zero values for the conditional probabilities of the impossible states, i.e., $P(x_{ij}|\pi_i) = 0$ for

[1] This denomination is justified by the fact that, in the following sections, we will set the probabilities for these states equal to zero.

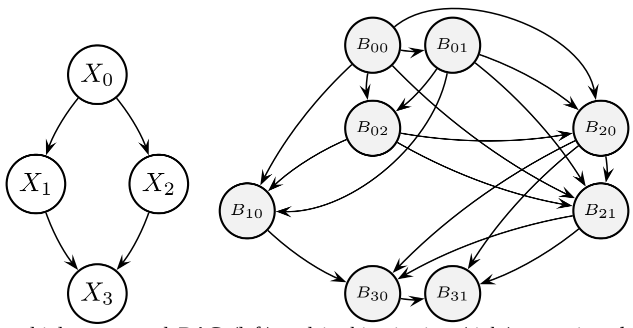


Figure 1. A multiply connected DAG (left) and its binarization (right) assuming $d_0 = 8$, $d_1 = 2$ and $d_2 = d_3 = 4$.

each $j \geq d_i$, for each $\pi_i \in \Omega_{\pi_i}$ and for each $i = 1, \ldots, n$; (iii) arbitrarily specify the mass function $P(X_i|\pi_i)$ for each π_i such that at least one of the states of the parents Π_i corresponding to π_i is an impossible state, for $i = 1, \ldots, n$. Considering Equation (1) and Prescription (ii), it is easy to note that, if the joint state $\mathbf{x} = (x_1, \ldots, x_n)$ of $\mathbf{X}$ is such that at least one of the states x_i, with $i = 1, \ldots, n$, is an impossible state, then $P(\mathbf{x}) = 0$, irrespectively of the values of the mass functions specified as in Prescription (iii). Thus, given a non-binarizable BN, the procedure described in this paragraph returns a binarizable BN that preserves the original probabilities. This makes possible to focus on the case of binarizable BNs without loss of generality, as in the following:

Definition 1. *Let $\langle \mathcal{G}, P(\mathbf{X}) \rangle$ be a binarizable BN. The binarization of $\langle \mathcal{G}, P(\mathbf{X}) \rangle$, is a binary BN $\langle \tilde{\mathcal{G}}, \tilde{P}(\tilde{\mathbf{X}}) \rangle$ obtained as follows: (i) $\tilde{\mathcal{G}}$ is the binarization of $\mathcal{G}$ with respect to $\mathbf{X}$ (ii) $\tilde{P}(\tilde{\mathbf{X}})$ corresponds to the following specifications of the conditional probabilities for the variables in $\tilde{\mathbf{X}}$ given their parents:* [2]

$$\tilde{P}(b_{ij}|b_{i(j-1)}, \ldots, b_{i0}, \tilde{\pi}_i) \propto \sum\nolimits_l^* P(x_{il}|\pi_i) \quad \begin{array}{l} i = 1, \ldots, n \\ j = 0, \ldots, m_i - 1 \\ \pi_i \in \Omega_{\Pi_i}, \end{array} \tag{5}$$

where the sum $\sum^*$ is restricted to the states $x_{il} \in \Omega_{X_i}$ such that the first $j+1$ bits of the binary representation of l are $b_{i0}, \ldots, b_{ij}$, π_i is the joint state of the parents of X_i corresponding to the joint state $\tilde{\pi}_i$ for the bits of the parents of X_i, and the symbol $\propto$ denotes proportionality.

In the following, to emphasize the fact that the variables $(B_{i(j-1)}, \ldots, B_{i0}, \tilde{\Pi}_i)$ are the parents of B_{ij} according to $\tilde{\mathcal{G}}$, we denote the joint state $(b_{i(j-1)}, \ldots, b_{i0}, \tilde{\pi}_i)$ as $\pi_{B_{ij}}$.

As an example of the procedure described in Definition 1, let X_0 be a variable with four states associated to a parentless node of a BN. Assuming for the corresponding mass function $[P(x_{00}), P(x_{01}), P(x_{02}), P(x_{03})] = (.2, .3, .4, .1)$, we can use Equation (5) to obtain the mass functions associated to the two bits of X_0 in the binarized BN. This leads to: $\tilde{P}(B_{00}) = (.6, .4)$, $\tilde{P}(B_{01}|B_{00} = 0) = (\frac{1}{3}, \frac{2}{3})$, $\tilde{P}(B_{01}|B_{00} = 1) = (\frac{3}{4}, \frac{1}{4})$, where the mass function of a binary variable B is denoted as an array $[P(B = 0), P(B = 1)]$.

[2]If the sum on the right-hand side of Equation (5) is zero for both the values of B_{ij}, the corresponding conditional mass function is arbitrary specified.

A BN and its binarization are basically the same probabilistic model and we can represent any updated belief in the original BN as a corresponding belief in the binarized BN, according to the following:

Theorem 1. *Let $\langle \mathcal{G}, P(\mathbf{X})\rangle$ be a binarizable BN and $\langle \tilde{\mathcal{G}}, \tilde{P}(\tilde{\mathbf{X}})\rangle$ its binarization. Then, given a queried variable $X_q \in \mathbf{X}$ and an evidence $X_E = x_E$:*

$$P(x_q|x_E) = \tilde{P}(b_{q(m_q-1)} \dots b_{q0}|\tilde{x}_E), \tag{6}$$

where $(b_{q(m_q-1)}, \dots, b_{q0})$ is the joint state of the bits of X_q corresponding to x_q.

4.4. Extension to Credal Networks

In order to generalize the binarization from BNs to CNs, we first extend the notion of binarizability: a CN $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$ is said to be binarizable if and only if $\mathbf{X}$ is binarizable. A non-binarizable CN can be made binarizable by the following procedure: (i) make the variables in $\mathbf{X}$ binarizable; (ii) specify zero upper (and lower) probabilities for conditional probabilities of the impossible states: $\underline{P}(x_{ij}|\pi_i) = \overline{P}(x_{ij}|\pi_i) = 0$ for each $j \geq d_i$, for each $\pi_i \in \Omega_{\Pi_i}$, and for each $i = 1, \dots, n$; (iii) arbitrarily specify the conditional credal sets $K(X_i|\pi_i)$ for each π_i such that at least one of the states of the parents Π_i corresponding to π_i is an impossible state, for $i = 1, \dots, n$. According to Equation (3) and Prescription (i), it is easy to check that, if the joint state $\mathbf{x} = (x_1, \dots, x_n)$ of $\mathbf{X}$ is such that at least one of the states x_i, with $i = 1, \dots, n$, is an impossible state, then $P(\mathbf{x}) = 0$, irrespectively of the conditional credal sets specified as in the Prescription (iii), and for each $P(\mathbf{X}) \in K(\mathbf{X})$. Thus, given a non-binarizable CN, the procedure described in this paragraph returns a binarizable CN, that preserves the original probabilities. This makes possible to focus on the case of binarizable CNs without loss of generality, as in the following:

Definition 2. *Let $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$ be a binarizable CN. The binarization of $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$ is a binary CN $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$, with $\tilde{\mathcal{G}}$ binarization of $\mathcal{G}$ with respect to $\mathbf{X}$ and the following separate specifications of the extreme probabilities:*[3]

$$\underline{\tilde{P}}(b_{ij}|\pi_{B_{ij}}) \equiv \min_{k=1,\dots,n_v} \tilde{P}_k(b_{ij}|\pi_{B_{ij}}), \tag{7}$$

where $\langle \tilde{\mathcal{G}}, \tilde{P}_k(\tilde{\mathbf{X}})\rangle$ is the binarization of $\langle \mathcal{G}, P_k(\mathbf{X})\rangle$ for each $k = 1, \dots, n_v$.

Definition 2 implicitly requires the binarization of all the BNs $\langle \mathcal{G}, P_k(\mathbf{X})\rangle$ associated to $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$, but the right-hand side of Equation (7) is not a minimum over all the BNs associated to a $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$, being in general $\tilde{P}(\tilde{\mathbf{X}}) \neq \{\tilde{P}_k(\tilde{\mathbf{X}})\}_{k=1}^{n_v}$. This means that it is not possible to represent an updating problem in a CN as a corresponding updating problem in the binarization of the CN, and we should therefore regard $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$ as an approximate description of $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$.

Remarkably, according to Equation (5), the conditional mass functions for the bits of X_i relative to the value $\tilde{\pi}_i$, can be obtained from the single mass function $P(X_i|\pi_i)$.

---

[3] Note that in the case of a binary variables a specification of the extreme probabilities as in Equation (7) is equivalent to the explicit specification of the (two) vertices of the conditional credal set $K(B_{ij}|\pi_{B_{ij}})$: if B is a binary variable and we specify $\underline{P}(B = 0) = s$ and $\underline{P}(B = 1) = t$, then the credal set $K(B)$ is the convex hull of the mass functions $P_1(B) = (s, 1 - s)$ and $P_2(B) = (1 - t, t)$.

Therefore, if we use Equation (5) with $P_k(\mathbf{X})$ in place of $P(\mathbf{X})$ for each $k = 1, \ldots, n_v$ to compute the probabilities $\tilde{P}_k(b_{ij}|\pi_{B_{ij}})$ in Equation (7), the only mass function required to do such calculations is $P_k(X_i|\pi_i)$. Thus, instead of considering all the joint mass functions $P_k(\mathbf{X})$, with $k = 1, \ldots, n_v$, we can restrict our attention to the conditional mass functions $P(X_i|\pi_i)$ associated to the elements of the conditional credal set $K(X_i|\pi_i)$ and take the minimum, i.e.,

$$\underline{\tilde{P}}(b_{ij}|\pi_{B_{ij}}) = \min_{P(X_i|\pi_i)\in K(X_i|\pi_i)} \tilde{P}(b_{ij}|\pi_{B_{ij}}), \tag{8}$$

where $\tilde{P}(b_{ij}|\pi_{b_{ij}})$ is obtained from $P(X_i|\pi_i)$ using Equation (5) and the minimization on the right-hand side of Equation (8) can be clearly restricted to the vertices of $K(X_i|\pi_i)$. The procedure is therefore linear in the input size.

As an example, let X_0 be a variable with four possible states associated to a parentless node of a CN. Assuming the corresponding credal set $K(X_0)$ to be the convex hull of the mass functions $(.2, .3, .4, .1)$, $(.25, .25, .25, .25)$, and $(.4, .2, .3, .1)$, we can use Equation (5) to compute the mass functions associated to the two bits of X_0 for each vertex of $K(X_0)$ and then consider the minima as in Equation (8), obtaining: $\underline{\tilde{P}}(B_{00}) = (.5, .3)$, $\underline{\tilde{P}}(B_{01}|B_{00} = 0) = (\frac{1}{3}, \frac{3}{7})$, $\underline{\tilde{P}}(B_{01}|B_{00} = 1) = (\frac{1}{2}, \frac{1}{4})$.

The equivalence between an updating problem in a BN and in its binarization as stated by Theorem 1 is generalizable in an approximate way to the case of CNs, as stated by the following:

Theorem 2. *Let $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$ be a binarizable CN and $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$ its binarization. Then, given a queried variable $X_q \in \mathbf{X}$ and an evidence $X_E = x_E$:*

$$\underline{P}(x_q|x_E) \geq \underline{\tilde{P}}(b_{q(m_q-1)}, \ldots, b_{q0}|\tilde{x}_E), \tag{9}$$

where $(b_{q(m_q-1)}, \ldots, b_{q0})$ is the joint state of the bits of X_q corresponding to x_q.

The inequality in Equation (9) together with its analogous for the upper probabilities provides an outer bound for the posterior interval associated to a generic updating problem in a CN. Such approximation is the posterior interval for the corresponding problem on the binarized CN.

Note that L2U cannot update joint states of two or more variables: this means that we can compute the right-hand side of Equation (9) by a direct application of L2U only in the case $m_q = 1$, i.e, if the queried variable X_q is binary.

If X_q has more than two possible states, a simple transformation of the binarized CN is necessary to apply L2U. The idea is simply to define an additional binary random variable, which is true if and only if $(B_{q(m_q-1)}, \ldots, B_{q0}) = (b_{q(m_q-1)}, \ldots, b_{q0})$. This variable is a deterministic function of some of the variables in $\tilde{\mathbf{X}}$, and can therefore be easily embedded in the CN $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$. We simply add to $\tilde{\mathcal{G}}$ a binary node, say $C_{b_{q(m_q-1)},\ldots,b_{q0}}$, with no children and whose parents are $B_{q(m_q-1)}, \ldots, B_{q0}$, and specify the probabilities for the state 1 (true) of $C_{b_{q(m_q-1)},\ldots,b_{q0}}$, conditional on the values of its parents $B_{q(m_q-1)}, \ldots, B_{q0}$, equal to one only for the joint value of the parents $(b_{q(m_q-1)}, \ldots, b_{q0})$ and zero otherwise. Then, it is straightforward to check that:

$$\underline{\tilde{P}}(b_{q(m_q-1)}, \ldots, b_{q0}|\tilde{x}_E) = \underline{\tilde{P}}'(C_{b_{q(m_q-1)},\ldots,b_{q0}} = 1|\tilde{x}_E), \tag{10}$$

where $\underline{P}'$ denotes the lower probability in the CN with the additional node. Thus, according to Equation (10), if X_q has more than two possible values, we simply add the node $C_{b_{q(m_q-1)},\ldots,b_{q0}}$ and run L2U on the modified CN.

Overall, the joint use of the binarization techniques described in this section, with the L2U algorithm represents a general procedure for efficient approximate updating in CNs. Clearly, the lack of a theoretical quantification of the outer approximation provided by the binarization as in Theorem 2, together with the fact that the posterior probabilities computed by L2U can be lower as well as upper approximations, suggests the opportunity of a numerical investigation of the quality of the overall approximation, which is the argument of the next section.

5. Tests and Results

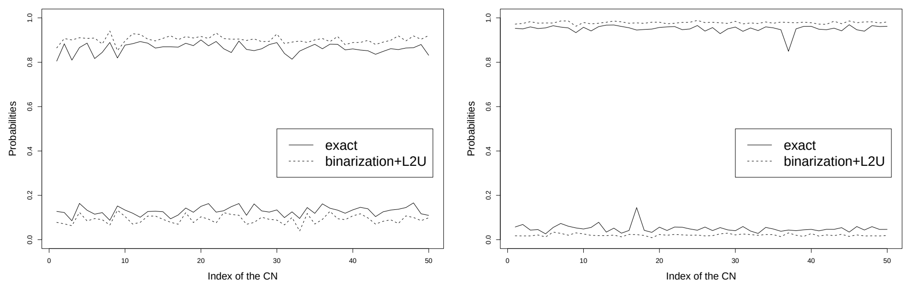
We have implemented a *binarization algorithm* to binarize CNs as in Definition 2 and run experiments for two sets of 50 random CNs based on the topology of the ALARM network and generated using *BNGenerator* [7]. The binarized networks were updated by an implementation of L2U, choosing the node "VentLung", which is a binary node, as target variable, and assuming no evidences. The L2U algorithm converges after 3 iterations and the overall computational time is quick: posterior beliefs for the networks were produced in less than one second in a Pentium computer, while the exact calculations used for the comparisons, based on branch-and-bound techniques [3], took a computational time between 10 and 25 seconds for each simulation. Results can be viewed in Figure 2.

As a comment, we note a good accuracy of the approximations with a mean square error around 3% and very small deviations. Remarkably the quality of the approximation is nearly the same for both the sets of simulations. Furthermore, we observe that the posterior intervals returned by the approximate method always include the corresponding exact intervals. This seems to suggest that the approximation due to the binarization dominates that due to L2U. It should also be pointed out that the actual difference between the computational time required by the two approaches would dramatically increase for larger networks: the computational complexity of the branch-and-bound method used for exact updating is exponential in the input size, while both our binarization algorithm and L2U (assuming that it converges) take a linear time; of course both the approaches have an exponential increase with an increase in the number of categories for the variables.

6. Conclusions

This paper describes an efficient algorithm for approximate updating on credal nets. This task is achieved transforming the credal net in a corresponding credal net over binary variables, and updating such binary credal net by the loopy version of 2U. Remarkably, the procedure can be applied to any credal net, without restrictions related to the network topology or to the number of possible states of the variables.

The posterior probability intervals in the binarized network are shown to contain the exact intervals requested by the updating problem (Theorem 2). Our numerical tests show that the quality of the approximation is satisfactory (few percents), remaining an outer



(a) Conditional credal sets with 4 vertices (b) Conditional credal sets with 10 vertices

Figure 2. A comparison between the exact results and approximations returned by the "binarization+L2U" procedure for the upper and lower values of $P(\mathrm{VentLung} = 1)$ on two sets of 50 randomly generated CNs based on the ALARM, with a fixed number of vertices for each conditional credal set.

approximation also after the approximate updating by L2U. Thus, considering also the efficiency of the algorithm, we can regard the "binarization+L2U" approach as a viable and accurate approximate method for fast updating on large CNs.

As a future research, we intend to explore the possibility of a theoretical characterization of the quality of the approximation associated to the binarization, as well as the identification of particular specifications of the conditional credal sets for which binarization provides high-quality approximations or exact results. Also the possibility of a formal proof of convergence for L2U, based on similar existing results for loopy belief propagation on binary networks [8] will be investigated in a future study.

Acknowledgements

The first author was partially supported by the Swiss NSF grant 200020-109295/1. The third author was partially supported by the FAPESP grant 05/57451-8. the fourth author was partially supported by CNPq grant 3000183/98-4. We thank Cassio Polpo de Campos for providing the exact simulations.

A. Proofs

Proof of Theorem 1. *With some algebra it is easy to check that the inverse of Equation (5) is:*

$$P(x_{il}|\pi_i) = \prod_{j=0}^{m_i-1} \tilde{P}(b_{ij}|b_{i(j-1)}, \ldots, b_{i0}, \tilde{\pi}_i), \tag{11}$$

where $(b_{i(m_i-1)}, \ldots, b_{i0})$ *is the* m_i*-bit binary representation of* l*. Thus,* $\forall \mathbf{x} \in \Omega_{\mathbf{X}}$*:*

$$P(\mathbf{x}) = \prod_{i=1}^{n} P(x_i|\pi_i) = \prod_{i=1}^{n} \prod_{j=0}^{m_i-1} \tilde{P}(b_{ij}|b_{i(j-1)}, \ldots, b_{i0}, \tilde{\pi}_i) = \tilde{P}(\tilde{\mathbf{x}}), \tag{12}$$

where the first passage is because of Equation (1), the second because of Equation (11) and the third because of the Markov condition for the binarized BN. Thus:

$$P(x_q|x_E) = \frac{P(x_q, x_E)}{P(x_E)} = \frac{\tilde{P}(b_{q(m_q-1)}, \ldots, b_{q0}, \tilde{x}_E)}{\tilde{P}(\tilde{x}_E)}, \tag{13}$$

that proves the thesis as in Equation (6). □

Lemma 1. *Let $\{\langle \mathcal{G}, P_k(\mathbf{X})\rangle\}_{k=1}^{n_v}$ be the BNs associated to a CN $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$. Let also $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$ be the binarization of $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$. Then, the BN $\langle \tilde{\mathcal{G}}, \tilde{P}_k(\tilde{\mathbf{X}})\rangle$, which is the binarization of $\langle \mathcal{G}, P_k(\mathbf{X})\rangle$, specifies a joint mass function that belongs to the strong extension of $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$, i.e.,*

$$\tilde{P}_k(\tilde{\mathbf{X}}) \in \tilde{K}(\tilde{\mathbf{X}}), \tag{14}$$

for each $k = 1, \ldots, n_v$, with $\tilde{K}(\tilde{\mathbf{X}})$ denoting the strong extension of $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$.

Proof. According to Equation (3), the strong extension of $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$ is:

$$\tilde{K}(\tilde{\mathbf{X}}) \equiv \mathrm{CH}\left\{ \prod_{B_{ij} \in \tilde{\mathbf{X}}} \tilde{P}(B_{ij}|\Pi_{B_{ij}}) : \tilde{P}(B_{ij}|\pi_{B_{ij}}) \in \tilde{K}(B_{ij}|\pi_{B_{ij}}) \quad \begin{array}{l} \forall \pi_{B_{ij}} \in \Omega_{\Pi_{B_{ij}}} \\ \forall B_{ij} \in \tilde{\mathbf{X}} \end{array} \right\}. \tag{15}$$

On the other side, considering the Markov condition for $\langle \tilde{\mathcal{G}}, \tilde{P}_k(\tilde{\mathbf{X}})\rangle$, we have:

$$\tilde{P}_k(\mathbf{X}) = \prod_{B_{ij} \in \tilde{\mathbf{X}}} \tilde{P}_k(B_{ij}|\Pi_{B_{ij}}). \tag{16}$$

But, for each $\pi_{B_{ij}} \in \Omega_{\Pi_{ij}}$ and $B_{ij} \in \tilde{\mathbf{X}}$, the conditional mass function $\tilde{P}_k(B_{ij}|\pi_{B_{ij}})$ belongs to the conditional credal set $\tilde{K}(B_{ij}|\pi_{B_{ij}})$ because of Equation (7). Thus, the joint mass function in Equation (16) belongs to the set in Equation (15), and that holds for each $k = 1, \ldots, n_v$. □

Lemma 1 basically states an inclusion relation between the strong extension of $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}})\rangle$ and the set of joint mass functions $\{P_k(\mathbf{X})\}_{k=1}^{n_v}$, which, according to the equivalence in Equation (12), is just an equivalent representation of $\langle \mathcal{G}, \mathbf{P}(\mathbf{X})\rangle$. This will be used to establish a relation between inferences in a CN and in its binarization, as detailed in the following:

Proof of Theorem 2. *We have:*

$$\underline{P}(x_q|x_E) = \min_{k=1,\ldots,n_v} P_k(x_q|x_E) = \min_{k=1,\ldots,n_v} \tilde{P}_k(b_{q(m_q-1)}, \ldots, b_{q0}|\tilde{x}_E), \tag{17}$$

where the first passage is because of Equations (4) and (2), and the second because of Theorem 1 referred to the BN $\langle \mathcal{G}, P_k(\mathbf{X})\rangle$, for each $k = 1, \ldots, n_v$.

On the other side, the lower posterior probability probability on the right-hand side of Equation (9) can be equivalently expressed as:

$$\underline{P}(b_{q(m_q-1)}, \ldots, b_{q0}|\tilde{x}_E) = \min_{\tilde{P}(\tilde{\mathbf{X}}) \in \tilde{K}(\tilde{\mathbf{X}})} \tilde{P}(b_{q(m_q-1)}, \ldots, b_{q0}|\tilde{x}_E), \tag{18}$$

where $\tilde{K}(\tilde{\mathbf{X}})$ *is the strong extension of* $\langle \tilde{\mathcal{G}}, \tilde{\mathbf{P}}(\tilde{\mathbf{X}}) \rangle$. *Considering the minima on the right-hand sides of Equations (17) and (18), we observe that they refer to the same function and the first minimum is over a domain that is included in that of the second because of Lemma 1. Thus, the lower probability in Equation (17) cannot be less than that on Equation (18), that is the thesis.* $\square$

References

[1] G. F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42:393–405, 1990.

[2] F. G. Cozman. Graphical models for imprecise probabilities. *Int. J. Approx. Reasoning*, 39(2-3):167–184, 2005.

[3] C. P. de Campos and F. G. Cozman. Inference in credal networks using multilinear programming. In *Proceedings of the Second Starting AI Researcher Symposium*, pages 50–61, Amsterdam, 2004. IOS Press.

[4] C. P. de Campos and F. G. Cozman. The inferential complexity of Bayesian and credal networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1313–1318, Edinburgh, 2005.

[5] E. Fagiuoli and M. Zaffalon. 2U: an exact interval propagation algorithm for polytrees with binary variables. *Artificial Intelligence*, 106(1):77–107, 1998.

[6] J. S. Ide and F. G. Cozman. IPE and L2U: Approximate algorithms for credal networks. In *Proceedings of the Second Starting AI Researcher Symposium*, pages 118–127, Amsterdam, 2004. IOS Press.

[7] J. S. Ide, F. G. Cozman, and F. T. Ramos. Generating random Bayesian networks with constraints on induced width. In IOS Press, editor, *Proceedings of the 16th European Conference on Artificial Intelligence*, pages 323–327, Amsterdam, 2004.

[8] J. M. Mooij and H. J. Kappen. Validity estimates for loopy belief propagation on binary real-world networks. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 945–952. MIT Press, Cambridge, MA, 2005.

[9] K. Murphy, Y. Weiss, and M. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 467–475, San Francisco, CA, 1999. Morgan Kaufmann.

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

[11] P. Walley. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, New York, 1991.