ELSEVIER

Artificial Intelligence 102 (1998) 21–38

Artificial Intelligence

Approximating MAPs for belief networks is NP-hard and other theorems

Ashraf M. Abdelbar [a,*], Sandra M. Hedetniemi [b]

[a] *Department of Computer Science, American University in Cairo, 113 Kasr El Aini Street, Cairo, Egypt*

[b] *Department of Computer Science, Clemson University, Clemson, SC 29634-1906, USA*

Received 9 May 1997

Abstract

Finding *maximum a posteriori* (MAP) assignments, also called Most Probable Explanations, is an important problem on Bayesian belief networks. Shimony has shown that finding MAPs is NP-hard. In this paper, we show that approximating MAPs with a constant ratio bound is also NP-hard. In addition, we examine the complexity of two related problems which have been mentioned in the literature. We show that given the MAP for a belief network and evidence set, or the family of MAPs if the optimal is not unique, it remains NP-hard to find, or approximate, alternative next-best explanations. Furthermore, we show that given the MAP, or MAPs, for a belief network and an initial evidence set, it is also NP-hard to find, or approximate, the MAP assignment for the same belief network with a modified evidence set that differs from the initial set by the addition or removal of even a single node assignment. Finally, we show that our main result applies to networks with constrained in-degree and out-degree, applies to randomized approximation, and even still applies if the ratio bound, instead of being constant, is allowed to be a polynomial function of various aspects of the network topology.

Keywords: Bayesian belief networks; Dynamic abduction; Next-best explanation; Probabilistic reasoning; Uncertainty; Complexity; Satisfiability

1. Introduction

Bayesian belief networks [13–15] are an important graphical knowledge representation useful for reasoning under (probabilistic) uncertainty. Two important abductive reasoning, also known as diagnostic reasoning or explanation, problems on (Bayesian) belief networks are probabilistic inference and MAP explanation. In probabilistic inference, the objective is

* Corresponding author. Email: abdelbar@acs.auc.eun.eg.

PII: S0004-3702(98)00043-5

to compute the probability of a given set of events conditioned on a given set of observances called the evidence. In MAP explanation, the objective is to find the most probable explanation for the evidence. Probabilistic inference was shown to be NP-hard in 1990 by Cooper [5], and in 1993 Dagum and Luby [6] showed that approximating probabilistic inference is also NP-hard. In 1994, Shimony [23] showed that MAP explanation is NP-hard. Here, we extend Shimony's proof to show that approximating MAP explanations is also NP-hard.

We also examine the problem of finding second-best explanations given the MAP explanation. It has been suggested [21, p. 20] that in some applications, "having alternative explanations is often useful and sometimes necessary. Having the second best, third best, and so on, can provide a useful gauge on the quality of the best explanation." In the literature, some algorithms have been described as more easily finding alternative explanations after the best explanation is found [4,19]. While this may be true in some cases or even in the average case, we show here that in the worst case, finding alternative explanations given the best explanation is no easier than finding the best explanation; they are both NP-hard. We define the following problem: Given a belief network and evidence set B and $\mathcal{E}$ and given all the optimal explanations (if the optimal explanation is not unique), find the second-best explanation for B and $\mathcal{E}$. We show that approximating this problem is NP-hard. We then show that this result can be extended to the general case where we are given the best, second-best, third-best, through the Kth-best explanation and required to find a $(K+1)$st-best solution.

We then look at the complexity of using the best explanation for a given belief network B and evidence set $\mathcal{E}_1$ to find the best explanation for B with a new evidence set $\mathcal{E}_2$. This is useful for applications where the evidence accumulates gradually over time; Charniak and Santos [3] have called this "dynamic abduction" and have indicated its usefulness in domains such as medical diagnosis, vision, and story understanding. Unfortunately, we prove that given a belief network B and two evidence sets $\mathcal{E}_1$ and $\mathcal{E}_2$, and given all the optimal explanations for B and $\mathcal{E}_1$, the problem of finding, or approximating, the best explanation for B and $\mathcal{E}_2$ is NP-hard, even if $\mathcal{E}_2$ differs from $\mathcal{E}_1$ by the addition or removal of a single node assignment.

2. Bayesian belief networks

Let $V = v_1, \ldots, v_n$ be a finite set of random variables. Let P be some probability distribution over V and let X, Y, and Z be three disjoint subsets of V. X is said to be *conditionally independent of* Y *given* Z if and only if for any given instantiations x, y, and z, of X, Y, and Z, respectively,

$$P(x \mid y, z) = P(x \mid z) \quad \text{whenever} \quad P(y, z) > 0. \tag{1}$$

Let $D = (V, E)$ be a directed acyclic graph (DAG) whose nodes are identified with the random variables $v_1, \ldots, v_n$. D is said to be a *minimal independency map* of P if and only if every $v \in V$ is conditionally independent, given its parents $\pi(v)$, of all its non-descendents [14, p. 120]. Given a set V of random variables and a probability distribution P over V, a *Bayesian belief network* representation of P is a dag (V, E), such that (V, E) is a minimal independency map of P, augmented with a set of conditional probability distributions $\{P_v\colon v \in V\}$ where each P_v is a local probability distribution which specifies the probability of each possible instantiation of v given every possible instantiation of its parents. For example, if node v is identified with a binary-valued random variable and has two parents, say u and w, which are identified with binary-valued variables, then P_v must specify for each of the four possible instantiations of $\pi(v)$, (i.e., $\{u \leftarrow \mathbf{T}, w \leftarrow \mathbf{T}\}$, $\{u \leftarrow \mathbf{F}, w \leftarrow \mathbf{T}\}$, $\{u \leftarrow \mathbf{T}, w \leftarrow \mathbf{F}\}$, and $\{u \leftarrow \mathbf{F}, w \leftarrow \mathbf{F}\}$) the two probabilities $P(v = \mathbf{T} \mid \pi(v))$ and $P(v = \mathbf{F} \mid \pi(v))$. However, since for any given instantiation of $\pi(v)$, the probability of $v = \mathbf{T}$ and the probability of $v = \mathbf{F}$ must sum to 1, it is sufficient for P_v to specify one probability, conventionally the probability of true for binary-valued variables, for each possible instantiation of $\pi(v)$. In general, if a variable is n-valued, P_v must specify $n - 1$ probabilities for every possible instantiation of $\pi(v)$.

An instantiation, or full assignment, $\mathcal{A}$ of a binary-valued belief network defined over a set of variables V is a mapping which assigns a truth value to each member of V. A partial assignment $\mathcal{E}$ of a subset of the nodes of a belief network assigns truth values to the members of the subset. Where there is no confusion, we will use the same symbol, e.g., $\mathcal{E}$, to refer to both the partial assignment and to the subset of V to which it assigns values. If $\mathcal{E}$ is a partial assignment of a belief network and $\mathcal{A}$ is a full assignment, then we will use the notation $\mathcal{E} \subseteq \mathcal{A}$ to indicate that $\mathcal{A}$ assigns to the set of variables which are assigned values by $\mathcal{E}$, the same values they are assigned by $\mathcal{E}$.

Based on the assumption that a belief network's underlying graph is an independency map of the network variables, Pearl [14] has shown that the joint probability of any given full instantiation, $\mathcal{A}\colon V \mapsto v_1, \ldots, v_n$, of the network variables can be computed according to

$$P(v_1, v_2, \ldots, v_n) = \prod_{i=1}^{n} P\big(v_i \mid \pi(v_i)\big). \tag{2}$$

The MAP problem is an optimization problem where we are given a belief network B and partial assignment $\mathcal{E}$ of B which represents "real-world" observations, or evidence, for which we seek an explanation. The problem is undefined if $P(\mathcal{E}) = 0$, i.e., if the evidence is impossible. The objective is to find the full assignment which is most probable given the evidence at hand. In other words, it is required to find the instantiation $\mathcal{A}$ with *maximum a posteriori* (MAP) probability $P(\mathcal{A} \mid \mathcal{E})$. This is equivalent to maximizing $P(\mathcal{A}, \mathcal{E})$ or maximizing $P(\mathcal{A})$ under the constraint that $\mathcal{E} \subseteq \mathcal{A}$. If the evidence set is empty, then the objective is to find the network assignment $\mathcal{A}$ with highest unconditional probability $P(\mathcal{A})$.

For the special case of singly-connected networks, which are networks in which the underlying undirected graph is also acyclic, an efficient polynomial-time (linear-time, in fact) algorithm exists [14]. However, for general multiply-connected networks, the problem has been shown to be NP-hard [23] and current methods [4,10,13,14,20,22] all have worst-case exponential-time complexity.

3. Complexity of approximating MAP explanations

In general, when a problem is proven to be NP-hard, this means that a polynomial-time algorithm which is guaranteed to find the optimal solution is unlikely to be found. However, it may still be possible to find polynomial-time algorithms which always find solutions which are guaranteed to be close to the optimal solution.

A *constant ratio-bounded approximation algorithm*, with constant bound ρ, for a maximization problem, is an algorithm which returns a solution whose quality s^{approx} is guaranteed to be within a ratio ρ of the quality s^{opt} of the optimal solution,

$$\frac{s^{\text{opt}}}{s^{\text{approx}}} \leqslant \rho. \tag{3}$$

In this paper, we prove that approximating the MAP problem with a constant ratio bound is at least as hard as finding an exact solution, i.e., they are both NP-hard.

Theorem 1. *Approximating the MAP assignment problem for belief networks with a constant ratio bound ρ is* NP-*hard for any $\rho \geqslant 1$.*

We begin by describing the "known NP-complete problem" we will use in our transformation; note that our transformation makes use of some of the constructs developed in [23].

3.1. Definition of no-negation ONE-IN-THREE 3SAT

An instance of the ONE-IN-THREE 3SAT problem consists of a set U of variables, and a collection C of clauses over U such that each clause $c \in C$ has $|c| = 3$. The question is whether there exists a truth assignment for U such that each clause has exactly one true literal. The ONE-IN-THREE 3SAT problem is NP-complete and remains NP-complete if it is restricted such that no $c \in C$ contains a negated literal [7, p. 259]. It is this restricted no-negation ONE-IN-THREE 3SAT problem (which we will abbreviate as NN3SAT for the remainder of the paper), that we will use in our proofs.

An example of this problem is the set of variables $\{u_1, u_2, u_3, u_4, u_5\}$ and the set of clauses $\{\{u_1, u_3, u_5\}, \{u_2, u_3, u_4\}, \{u_1, u_4, u_5\}, \{u_2, u_4, u_5\}, \{u_2, u_3, u_5\}\}$, i.e.,

$$\begin{aligned}\phi = {} & (u_1 \vee u_3 \vee u_5) \wedge (u_2 \vee u_3 \vee u_4) \wedge (u_1 \vee u_4 \vee u_5) \\ & \wedge (u_2 \vee u_4 \vee u_5) \wedge (u_2 \vee u_3 \vee u_5). \end{aligned} \tag{4}$$

This problem instance happens to be satisfiable: one satisfying assignment is $\{u_1 = \mathbf{T}, u_2 = \mathbf{T}, u_3 = \mathbf{F}, u_4 = \mathbf{F}, u_5 = \mathbf{F}\}$.

3.2. Transformation

Let (U, C) be an arbitrary instance of NN3SAT and let $\rho \geqslant 1$ be an arbitrary constant. Let $n = |U|$, $m = |C|$, and

$$d = \lceil \log \rho \rceil + 1, \tag{5}$$

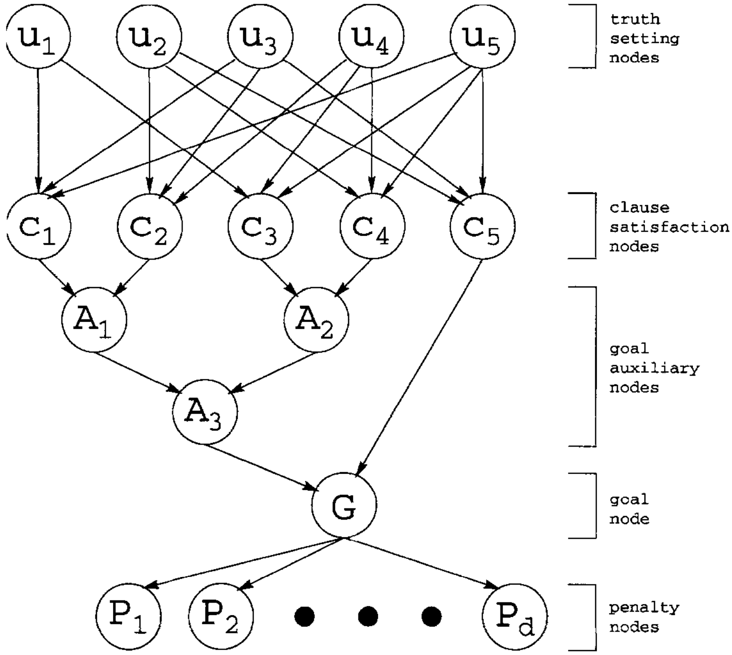

Fig. 1. Illustration of Construction 2.

where, for the remainder of the paper, log will be assumed to denote $\log_2$ unless another base is explicitly specified. We will present a polynomial-time (in the size of (U, C)) procedure for constructing a belief network B and evidence $\mathcal{E}$; the construction is illustrated in Fig. 1.

Construction 2.
Input: An instance (U, C) of NN3SAT and a constant $\rho \geqslant 1$.
Output: A belief network B and evidence $\mathcal{E}$.

(i) For each $u \in U$, construct a node $v_u \in V$ which will be called a truth setting node. Let $P_{\text{truth}}(\mathcal{A})$ represent the joint probability of the truth setting nodes under an instantiation $\mathcal{A}$,

$$P_{\text{truth}}(\mathcal{A}) = \prod_{u \in U} P(v_u). \tag{6}$$

(ii) For each $c \in C$, construct a node $v_c \in V$ which will be called a clause satisfaction node. Let $P_{\text{clause}}(\mathcal{A})$ represent the conditional joint probability of the clause satisfaction nodes under an instantiation $\mathcal{A}$,

$$P_{\text{clause}}(\mathcal{A}) = \prod_{c \in C} P\big(v_c \mid \pi(v_c)\big). \tag{7}$$

(iii) Construct a single node $G \in V$ which will be called the goal node.

(iv) Construct d nodes, $p_1, \ldots, p_d$, which will be called penalty nodes [1]. Let $P_{\text{penalty}}(\mathcal{A})$ represent the conditional joint probability of the d penalty nodes under an instantiation $\mathcal{A}$.

(v) All the truth setting nodes will be root nodes. For each truth setting node v_u, set v_u's probability distribution to be

$$P(v_u = \mathbf{T}) = \tfrac{1}{2}. \tag{8}$$

(Note that $P(v_u)$ under an instantiation $\mathcal{A}$ will be one-half regardless of how $\mathcal{A}$ assigns v_u. Therefore,

$$P_{\text{truth}}(\mathcal{A}) = \left(\tfrac{1}{2}\right)^n, \tag{9}$$

for any $\mathcal{A}$.)

(vi) For each clause $c \in C$, construct an edge, for each $u \in c$, from v_u to v_c. Therefore, each clause satisfaction node will have an in-degree of exactly three. Set the conditional distribution for each clause satisfaction node v as follows:

$$P(v = \mathbf{T} \mid u_1, u_2, u_3) = \begin{cases} 1, & \text{if exactly one of } u_1, u_2, \text{ and } u_3 \text{ is } \mathbf{T}, \\ 0, & \text{otherwise.} \end{cases}$$

(Note that this distribution, which will be called an *XOR*[2] *distribution*, is designed so that the probability $P(v \mid \pi(v))$ is 1 if and only if the network is instantiated in such a way that v is the xor-ing of its parents—in which case we say that the instantiation *respects* node v's XOR distribution. An instantiation $\mathcal{A}$ will have $P_{\text{clause}}(\mathcal{A})$ equal to 1 if it respects all XOR distributions and equal to 0 if $\mathcal{A}$ disrespects the XOR distribution of even a single clause satisfaction node.)

(vii) Let the following distribution for a node v with parents $\pi(v)$ be called an AND distribution.

$$P\big(v = \mathbf{T} \mid \pi(v)\big) = \begin{cases} 1, & \text{if all } \pi(v) \text{ are } \mathbf{T}, \\ 0, & \text{otherwise.} \end{cases}$$

(Note that the probability $P(v \mid \pi(v))$ under an instantiation $\mathcal{A}$ will be 1 if $\mathcal{A}$ assigns v to a truth value equal to the and-ing of the truth values of its parents; otherwise, $P(v \mid \pi(v))$ under $\mathcal{A}$ will be 0. In the former case, we say that $\mathcal{A}$ respects v's AND distribution, and, in the latter, that it disrespects it.)

(viii) Construct a binary tree of nodes, such that the clause satisfaction nodes are the leaf nodes of the tree, G is the root of the tree, all edges in the tree are directed towards the root, and all interior nodes, including the root node G, have the AND distribution. The interior nodes of the tree, excluding G, will be called goal auxiliary nodes. Let $P_{\text{goal}}(\mathcal{A})$ represent the conditional joint probability of G and the goal auxiliary nodes.

---

[1] The penalty nodes were called drain nodes in [23].

[2] The XOR function is commonly generalized to n variables $x_1, \ldots, x_n$ in one of two ways: either as the "one-of" function, defined to be true if and only if exactly one of $x_1, \ldots, x_n$ is true, or as the parity function, defined to be true if and only if exactly k of $x_1, \ldots, x_n$ are true where k is an odd number. Throughout this paper, we adopt the former convention.

(Note that $P_{\text{goal}}(\mathcal{A})$ under an instantiation $\mathcal{A}$ will be 1 if $\mathcal{A}$ respects all AND distributions and will be 0 otherwise. Further, note that if $\mathcal{A}$ respects all AND distributions, it must assign G to a value equal to the and-ing of the clause satisfaction nodes. Therefore, if $P(\mathcal{A}) \neq 0$, then $\mathcal{A}$ can only assign G to true if all clauses $c \in C$ are satisfied by the truth values $\mathcal{A}$ has assigned to the truth setting nodes.)

(ix) Construct an edge from G to each of the d penalty nodes and construct the conditional probability distribution for each penalty node v as follows:

$$P(v = \mathbf{T} \mid G) = \begin{cases} 1, & \text{if } G = \mathbf{T}, \\ \frac{1}{2}, & \text{if } G = \mathbf{F}. \end{cases}$$

(Note that if the goal node is false, then $P_{\text{penalty}}(\mathcal{A})$ will be $(1/2)^d$, while if G is true, $P_{\text{penalty}}(\mathcal{A})$ will be 1 if $\mathcal{A}$ assigns all the penalty nodes to true and will be 0 otherwise.)

(x) Make the evidence set $\mathcal{E}$ empty.

Proof. Let A be a polynomial-time approximation algorithm for the MAP problem with constant ratio-bound ρ. For a given arbitrary instance (U, C) of NN3SAT, we apply Construction 2 to produce an instance $(B, \mathcal{E})$ of the MAP problem and run algorithm A. Let $\mathcal{A}'$ be the solution that is returned by algorithm A and let $\hat{\mathcal{A}}$ be the theoretically optimal solution. We claim that (U, C) is satisfiable if and only if

$$P(\mathcal{A}') = \left(\tfrac{1}{2}\right)^n. \tag{10}$$

Let $\mathcal{A}$ be an arbitrary instantiation of B. If $\mathcal{A}$ disrespects any AND or XOR distribution or assigns any of the penalty nodes to false while assigning G to true, then

$$P(\mathcal{A}) = 0. \tag{11}$$

If $P(\mathcal{A}) \neq 0$, then $\mathcal{A}$ must respect all AND and XOR distributions. This means that $\mathcal{A}$ can only assign G to true if the clause satisfaction nodes are all true; $\mathcal{A}$ can assign all the clause satisfaction nodes to true only if it has assigned the truth setting nodes in a manner that corresponds to a satisfying assignment for (U, C), i.e., such that exactly one parent per clause satisfaction node is assigned to true. Therefore, if $P(\mathcal{A}) \neq 0$, there are exactly two cases. If $\mathcal{A}$ assigns G to true, then

$$P(\mathcal{A}) = P_{\text{truth}}(\mathcal{A}) = \left(\tfrac{1}{2}\right)^n; \tag{12}$$

if $\mathcal{A}$ assigns G to false, then

$$P(\mathcal{A}) = P_{\text{truth}}(\mathcal{A}) P_{\text{penalty}}(\mathcal{A}) = \left(\tfrac{1}{2}\right)^n \left(\tfrac{1}{2}\right)^d. \tag{13}$$

Therefore, for an arbitrary $\mathcal{A}$, exactly one of equations (11)–(13) will hold.

If (U, C) is satisfiable, then

$$P(\hat{\mathcal{A}}) = \left(\tfrac{1}{2}\right)^n. \tag{14}$$

It cannot be the case that

$$P(\mathcal{A}') = \left(\tfrac{1}{2}\right)^n \left(\tfrac{1}{2}\right)^d, \tag{15}$$

because

$$\frac{P(\hat{\mathcal{A}})}{P(\mathcal{A}')} = \frac{\left(\frac{1}{2}\right)^n}{\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d} = 2^d > \rho. \tag{16}$$

Therefore,

$$P(\mathcal{A}') = P(\hat{\mathcal{A}}) = \left(\tfrac{1}{2}\right)^n, \tag{17}$$

if and only if (U, C) is satisfiable.

Our transformation can clearly be carried out in polynomial time. One truth setting node is constructed for each $u \in U$, and one clause satisfaction node is constructed for each $c \in C$. The number of goal auxiliary nodes created in step (viii) is linear in the number of clause satisfaction nodes. Since ρ is constant, the number of penalty nodes is also constant. No node in the network has an in-degree greater than three, therefore the size of the local probability distributions P_v is bounded by a constant.

4. Complexity of next-best explanations

4.1. Second-best explanations

In this section, we are interested in the following problem. Given a belief network B, evidence set $\mathcal{E}$, and the set $\mathcal{F}$ of optimal assignments for B and $\mathcal{E}$, the objective is to find the assignment $\mathcal{A} \notin \mathcal{F}$ with highest conditional probability $P(\mathcal{A} \mid \mathcal{E})$. We call this the SECOND-BEST-MAP problem.

Theorem 3. *Approximating the SECOND-BEST-MAP problem with a constant ratio bound ρ is* NP-*hard for any $\rho \geqslant 1$.*

Given an arbitrary instance (U, C) of NN3SAT, we create in polynomial time a belief network B, an evidence set $\mathcal{E}$, and a set of assignments $\mathcal{F}$. We then prove that the set $\mathcal{F}$ we have constructed is exactly the set of optimal MAP assignment for B and $\mathcal{E}$. Finally, we show that (U, C) can be decided by approximating the second-best MAP assignment.

To construct B, we build on Construction 2. We add a new "dummy" truth setting node z, as illustrated in Fig. 2, which does not correspond to any variable in U and construct edges from it to all the clause satisfaction nodes. P_{truth} will now denote the joint probability of the truth setting nodes plus the new dummy z. Observe that setting z to true and setting all the "real" truth setting nodes to false is a trivial way to satisfy all the clauses.

We construct the distribution for z as follows:

$$P(z = \mathbf{T}) = \tfrac{6}{10}. \tag{18}$$

Let $\hat{\mathcal{A}}$ be a network assignment for B which assigns z to true, all other truth setting nodes to false, all clause satisfaction nodes to true, the goal node and all goal auxiliary nodes to true, and all penalty nodes to true. Then,

$$P_{\text{clause}}(\hat{\mathcal{A}}) = P_{\text{goal}}(\hat{\mathcal{A}}) = P_{\text{penalty}}(\hat{\mathcal{A}}) = 1, \tag{19}$$

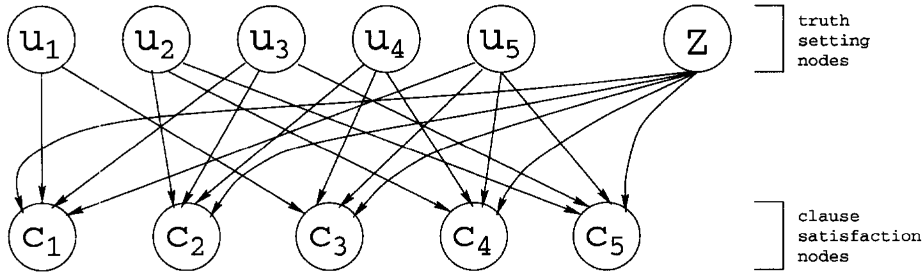

Fig. 2. Adding a dummy truth setting node z.

and

$$P(\hat{\mathcal{A}}) = P_{\text{truth}}(\hat{\mathcal{A}}) = \tfrac{6}{10}\left(\tfrac{1}{2}\right)^n. \tag{20}$$

Any assignment $\mathcal{A}$ different from $\hat{\mathcal{A}}$ will have $P(\mathcal{A}) < P(\hat{\mathcal{A}})$. Therefore, the family $\mathcal{F}$ of optimal assignments for B and $\mathcal{E} = \Phi$ consists of the single assignment $\hat{\mathcal{A}}$ and can be constructed in polynomial time.

We now prove that to approximate the second-best MAP assignment, an algorithm would have to find a satisfying truth assignment for (U, C) if one exists. Before the addition of z, there were exactly three possible values for $P(\mathcal{A})$ for an arbitrary assignment $\mathcal{A}$. With the addition of z, there are now two possibilities for $P_{\text{truth}}(\mathcal{A})$ and exactly five possible values for $P(\mathcal{A})$. These are summarized in Table 1, in decreasing order of probability.

The first case in Table 1 corresponds to $\hat{\mathcal{A}}$. If (U, C) is satisfiable, then the optimal solution $\hat{\mathcal{A}}_2$ for SECOND-BEST-MAP has the probability shown in case 2 of the table. The next-best solution for SECOND-BEST-MAP has the probability shown in the third case of the table. However, the ratio between the two probabilities is greater than ρ. Therefore, if $\mathcal{A}_2'$ is the assignment returned by an approximation algorithm A with ratio bound ρ, then

$$P(\mathcal{A}_2') = \tfrac{4}{10}\left(\tfrac{1}{2}\right)^n, \tag{21}$$

if and only if (U, C) is satisfiable.

4.2. *K*th-best explanation

We have shown that, in general, finding the second-best solution given the best solution is just as difficult as finding the best solution from "scratch". Now, let us go one step further and consider the following problem. Given a belief network B, evidence set $\mathcal{E}$, and a collection $\hat{\mathcal{F}}$ of sets $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_K$, where $\mathcal{F}_i$ represents the set of ith-best solutions, for $i = 1, \ldots, K$, the objective is to find the assignment $\mathcal{A}$ with highest conditional probability $P(\mathcal{A} \mid \mathcal{E})$, such that $\mathcal{A} \notin \mathcal{F}_i$, for $i = 1, \ldots, K$.

We can make some modifications to our earlier discussion to show that approximating this problem is NP-hard for any constant K. Instead of a single node z, we add K "dummy" truth setting nodes $z_1, \ldots, z_K$. Each node z_i is connected to all the clause satisfaction nodes. This means that setting exactly one dummy z_i to true and setting all the "real" truth setting nodes to false will trivially satisfy all the clauses.

Table 1
The five possible values for $P(\mathcal{A})$ in the SECOND-BEST-MAP transformation

Case	$P(\mathcal{A})$	Description of $\mathcal{A}$
1	$P(\mathcal{A}) = \frac{6}{10}\left(\frac{1}{2}\right)^n$	Assigns z to true, all truth setting nodes to false, respects all XOR and AND distributions, and assigns all penalty nodes to true.
2	$P(\mathcal{A}) = \frac{4}{10}\left(\frac{1}{2}\right)^n$	Assigns z to false, assigns the truth setting nodes in a manner that corresponds to a satisfying assignment to (U, C), respects all AND and XOR distributions, and assigns all penalty nodes to true.
3	$P(\mathcal{A}) = \frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d$	Assigns z to true, assigns any of the truth setting nodes to true, and respects all XOR and AND distributions (which means it must assign the goal node to false).
4	$P(\mathcal{A}) = \frac{4}{10}\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d$	Assigns z to false, assigns the truth setting nodes in a manner that corresponds to a non-satisfying assignment to (U, C), and respects all XOR and AND distributions.
5	$P(\mathcal{A}) = 0$	Disrespects any of the AND or XOR distributions or assigns any of the penalty nodes to false while assigning the goal node to true.

We also add K new penalty nodes $p_1^z, \ldots, p_K^z$, which will have only G as their parent and will have the penalty distribution,

$$P(p_i^z = \mathbf{T} \mid G) = \begin{cases} 1, & \text{if } G = \mathbf{T}, \\ \frac{1}{2}, & \text{if } G = \mathbf{F}. \end{cases}$$

The probability distribution of each z_i node, for $i = 1, \ldots, K$, is set as follows:

$$P(z_i = \mathbf{T}) = \frac{1}{2} + \frac{1}{10}\frac{K - i + 1}{K}. \tag{22}$$

Let P_z denote the joint probability of the z nodes. Note that for $i_1 < i_2$,

$$P(z_{i_1} = \mathbf{T}) > P(z_{i_2} = \mathbf{T}), \quad P(z_{i_1} = \mathbf{F}) < P(z_{i_2} = \mathbf{F}), \tag{23}$$

and for any $i = 1, \ldots, K$,

$$\frac{P(z_i = \mathbf{F})}{P(z_i = \mathbf{T})} \geqslant \frac{\left(\frac{4}{10}\right)}{\left(\frac{6}{10}\right)} > \frac{1}{2}. \tag{24}$$

Therefore, the minimum possible value of P_z will occur for an assignment which assigns all the z_i nodes to false; let this minimum value be denoted $P_z^{\min}$. The maximum possible value will occur for an assignment which assigns all the z_i nodes to true; let us denote this value as $P_z^{\max}$. Now, let $\hat{\mathcal{A}}_i$ be an assignment which assigns z_i to true, all other truth setting nodes to false, respects all AND and XOR distributions (which implies that it must assign G to true), and assigns the penalty nodes to true. Then, for any $i_1 < i_2$,

$$P_z^{\max} > P_z(\hat{\mathcal{A}}_{i_1}) > P_z(\hat{\mathcal{A}}_{i_2}) > P_z^{\min}. \tag{25}$$

Further, let P_{penalty}^z denote the joint probability of the K newly added penalty nodes, $p_1^z, \dots, p_K^z$, for an assignment which assigns G to false,

$$P_{\text{penalty}}^z = \left(\tfrac{1}{2}\right)^K. \tag{26}$$

From (24),

$$\frac{P_z^{\min}}{P_z^{\max}} > \left(\tfrac{1}{2}\right)^K, \tag{27}$$

therefore,

$$P_{\text{penalty}}^z < \frac{P_z^{\min}}{P_z^{\max}}. \tag{28}$$

Let $\mathcal{A}_G$ be an arbitrary assignment which respects all AND and XOR distributions and assigns G (and all penalty nodes) to true; let $\mathcal{A}_{\bar{G}}$ be an arbitrary assignment which respects all AND and XOR distributions and assigns G to false. Then,

$$\begin{aligned}
\frac{P(\mathcal{A}_G)}{P(\mathcal{A}_{\bar{G}})} &\geqslant \frac{P_z^{\min}\left(\frac{1}{2}\right)^n}{P_z^{\max}\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d P_{\text{penalty}}^z}, && (29)\\
&\geqslant 2^d \frac{P_z^{\min}}{P_z^{\max} P_{\text{penalty}}^z}, && (30)\\
&> \rho. && (31)
\end{aligned}$$

In other words, not only does any assignment which assigns G to true still have higher probability than any assignment which assigns G to false, assuming both assignments respect all AND and XOR distributions, the ratio between them is greater than ρ.

This means that the network assignment with highest unconditional probability is $\hat{\mathcal{A}}_1$, the network assignment with second-highest probability is $\hat{\mathcal{A}}_2$, and in general for $i = 1, \dots, K$, the assignment with ith highest probability is $\hat{\mathcal{A}}_i$. If (U, C) is satisfiable, the next-best assignment after $\hat{\mathcal{A}}_K$ will have to assign G to true by assigning the "non-dummy" truth setting nodes in a manner that corresponds to a satisfying assignment to (U, C). An assignment which assigns G to false cannot be returned as an approximation to the optimal next-best assignment because, from (31), the ratio between them would be greater than ρ.

5. Complexity of dynamic abduction

In this and the following section, we are interested in determining the complexity of finding, or approximating, the optimal MAP assignment for a belief network B and evidence set $\mathcal{E}_2$ given all the optimal assignments for the same network B and an initial evidence set $\mathcal{E}_1$, such that $\mathcal{E}_1$ differs from $\mathcal{E}_2$ by the addition or removal of one or more node assignments.

5.1. Dynamic abduction with expanding evidence

The NP-hardness of the case where $\mathcal{E}_2$ involves adding node assignments to $\mathcal{E}_1$ is a fairly straightforward extension of Theorem 3. We again, as in Section 4.1, make use of a single "dummy" truth-setting node z and we set $\mathcal{E}_1$ to the empty set. This makes the unique optimal assignment for B and $\mathcal{E}_1$ the trivial assignment which assigns z to true and all other truth setting nodes to false. We then set $\mathcal{E}_2$ to the single assignment $\{z \leftarrow \mathbf{F}\}$; the optimal assignment for B and $\mathcal{E}_2$ will thus not be able to set z to true in order to assign G to true without disrespecting any AND or XOR distributions. If (and only if) (U, C) is satisfiable, the optimal assignment $\hat{\mathcal{A}}$ for B and $\mathcal{E}_2$ will correspond to the second case of Table 1. The next-best solution will correspond to the fourth case of the table and its probability will not be within a ratio ρ of the optimal,

$$\frac{\frac{4}{10}\left(\frac{1}{2}\right)^n}{\frac{4}{10}\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d} > \rho. \tag{32}$$

Theorem 4. *Given a Bayesian belief network B, two evidence sets $\mathcal{E}_1$ and $\mathcal{E}_2$ such that $\mathcal{E}_1 \subset \mathcal{E}_2$, and the set $\mathcal{F}$ of optimal MAPs for B and $\mathcal{E}_1$, the problem of approximating the optimal MAP explanation for B and $\mathcal{E}_2$ with a constant ratio bound ρ is* NP*-hard for any $\rho \geqslant 1$.*

5.2. Dynamic abduction with diminishing evidence

In this section, our concern is with dynamic abduction in the case where the new evidence set $\mathcal{E}_2$ differs from $\mathcal{E}_1$ by the removal of one or more node assignments. Proving this case is a little more involved than in the case where the evidence is expanding.

Theorem 5. *Given a Bayesian belief network B, two evidence sets $\mathcal{E}_1$ and $\mathcal{E}_2$ such that $\mathcal{E}_2 \subset \mathcal{E}_1$, and the set $\mathcal{F}$ of optimal MAPs for B and $\mathcal{E}_1$, the problem of approximating the optimal MAP explanation for B and $\mathcal{E}_2$ with a constant ratio bound ρ is* NP*-hard for any $\rho \geqslant 1$.*

Given an instance of (U, C), we construct a belief network B as described in Section 4.1. Let

$$k = \left\lceil \log_{\frac{10}{6}} \left(\lceil \rho \rceil\right) \right\rceil + 1. \tag{33}$$

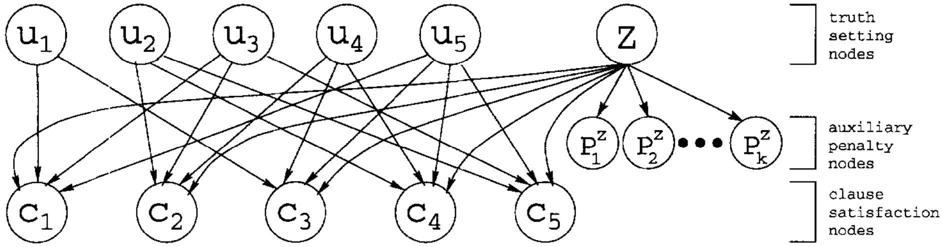

Fig. 3. Attaching penalty nodes to z.

We add k additional nodes $p_1^z, \ldots, p_k^z$, which will be called auxiliary penalty nodes, and we construct an edge from z to each of the auxiliary penalty nodes, as illustrated in Fig. 3. The distribution for each auxiliary penalty node will be as follows:

$$P(p_i^z = \mathbf{T} \mid z) = \begin{cases} \frac{6}{10}, & \text{if } z = \mathbf{T}, \\ 1, & \text{if } z = \mathbf{F}. \end{cases}$$

Let P^z_{penalty} denote the conditional joint probability of the k auxiliary penalty nodes.

Let $\mathcal{A}$ be an arbitrary assignment for B; there are four possible cases for $P(\mathcal{A})$, presented in Table 2.

Let $\mathcal{E}_1 = \{z \to \mathbf{T}\}$ and $\mathcal{E}_2 = \Phi$. The family $\mathcal{F}$ of optimal MAP assignments for B and $\mathcal{E}_1$ will consist of the unique assignment, let us denote it $\hat{\mathcal{A}}_{\mathcal{E}_1}$, corresponding to case 2 of the table, and can be constructed in polynomial time. For B and $\mathcal{E}_2$, the optimal assignment $\hat{\mathcal{A}}_{\mathcal{E}_2}$ will correspond to the first case of the table and will have probability

$$P(\hat{\mathcal{A}}_{\mathcal{E}_2}) = \tfrac{4}{10}\left(\tfrac{1}{2}\right)^n. \tag{34}$$

Let $\mathcal{A}'_{\mathcal{E}_2}$ be the assignment returned by an approximation algorithm with ratio bound ρ. If (U, C) is satisfiable, then $\mathcal{A}'_{\mathcal{E}_2}$ cannot correspond to case 2 or 3 of the table because

$$\frac{\frac{4}{10}\left(\frac{1}{2}\right)^n}{\frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{6}{10}\right)^k} > \rho, \tag{35}$$

and cannot correspond to case 4 of the table because

$$\frac{\frac{4}{10}\left(\frac{1}{2}\right)^n}{\frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d} > \rho. \tag{36}$$

Therefore,

$$P(\mathcal{A}'_{\mathcal{E}_2}) = \tfrac{4}{10}\left(\tfrac{1}{2}\right)^n, \tag{37}$$

if and only if (U, C) is satisfiable.

Table 2
The four possible values for $P(\mathcal{A})$ in the dynamic abduction with diminishing evidence transformation

Case	$P(\mathcal{A})$	Description of $\mathcal{A}$
1	$P(\mathcal{A}) = \frac{4}{10}\left(\frac{1}{2}\right)^n$	Assigns z to false and all penalty and auxiliary penalty nodes to true, includes a satisfying assignment to the non-dummy truth setting nodes, and respects all AND and XOR distributions.
2	$P(\mathcal{A}) = \frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{6}{10}\right)^k$	Assigns z and all penalty and auxiliary penalty nodes to true, and all other truth setting nodes to false, and respects all AND and XOR distributions.
3	$\frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{4}{10}\right)^k \leqslant P(\mathcal{A}) < \frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{6}{10}\right)^k$	Assigns z and all penalty nodes to true, and all other truth setting nodes to false, does not assign all auxiliary penalty nodes to true, and respects all AND and XOR distributions.
4	$P(\mathcal{A}) \leqslant \frac{6}{10}\left(\frac{1}{2}\right)^n\left(\frac{1}{2}\right)^d$	Assigns G to false or does not respect all AND and XOR distributions.

6. Additional results and concluding remarks

6.1. Constrained topology

It is easy to extend Theorem 1 to networks with a maximum in-degree of 2 and a maximum out-degree of 2. The method described by Shimony [23] for reducing the out-degree of the goal node can be used in our constructions to reduce the out-degree of the goal node and the truth setting nodes. In networks created by Construction 2, the clause satisfaction nodes have an in-degree of 3 and all other nodes have a maximum in-degree of 2. The three-variable XOR distribution of the clause satisfaction nodes can, however, easily be avoided using the well-known identity

$$XOR(a,b,c) \equiv [(\neg a \wedge \neg b) \wedge c] \vee [(\neg a \wedge b) \wedge \neg c] \vee [(a \wedge \neg b) \wedge \neg c].$$

We replace each XOR node by a constant number of two-input AND, two-input OR, and one-input NOT nodes.

6.2. Randomized approximation

A number of randomized algorithms have been developed for belief networks. They are based on using the probability distributions residing in network nodes to generate network instantiations and using the instantiation that is generated most frequently as an approximation of the optimal MAP assignment. Unfortunately, we will show it is not possible, unless RP = NP, to guarantee that the solution returned by a polynomial-time algorithm has a fixed probability δ of being the optimal, or even within a constant ratio bound ρ of the optimal.

A *Monte Carlo algorithm A* for a yes-no decision problem Π is an algorithm which always terminates with an affirmative or negative answer such that

- The probability that A answers affirmatively and the correct answer to Π is "no" is 0;
- The probability that A answers negatively and the correct answer to Π is "yes" is less than $1/2$.

A problem Π belongs to the complexity class RP if and only if there exists a polynomial-time Monte Carlo algorithm for it [12]. While it is known that $\mathrm{P} \subseteq \mathrm{RP} \subseteq \mathrm{NP}$, it remains open whether $\mathrm{P} \subset \mathrm{RP}$ and whether $\mathrm{RP} \subset \mathrm{NP}$.

Theorem 6. *If* $\mathrm{RP} \neq \mathrm{NP}$, *a polynomial-time algorithm which is guaranteed to find a solution which has a fixed probability* δ *of being within a ratio* ρ *of the optimal solution does not exist for the MAP problem for any* $\rho \geqslant 1$ *and* $\delta \in (0, 1)$.

Let A be such an algorithm; we use A to construct a Monte Carlo algorithm A' in the following way. Given an instance of NN3SAT, apply Construction 2 to produce a belief network and evidence set, run algorithm A k times, and answer affirmatively if any of the k runs return a positive answer, where

$$k = \left\lceil \left(-\frac{1}{\log(1-\delta)} \right) \right\rceil + 1. \tag{38}$$

The probability of k "false negatives" is

$$(1-\delta)^k < \tfrac{1}{2}. \tag{39}$$

Therefore, A' is a Monte Carlo algorithm for NN3SAT and since k is constant, it runs in polynomial time.

6.3. Approximation with a polynomial ratio bound

Thus far, we have shown that a polynomial-time ratio-bounded approximation algorithm with a constant ρ cannot be found unless P = NP. However, this leaves open the possibility of a ratio-bounded algorithm with a ρ that is not a constant but perhaps a simple function of the problem size. For example, for the set covering problem, where it is required to cover a set of elements X with the smallest subset of a family $\mathcal{F}$ of subsets of X, there exists a polynomial-time algorithm which guarantees a ratio bound of $(\ln|X| + 1)$ [11]. Such a ratio bound is not a constant and would not be precluded even though, because the ln function grows so slowly, it is almost as useful as a constant ratio-bounded algorithm.

Therefore, it would be significant to prove for the MAP problem not only the NP-hardness of approximation with a constant ratio bound but also of approximation with a ratio bound that is an arbitrary positive polynomial function of the size of the network. However, we will settle for proving this for a ratio bound that is an arbitrary positive polynomial function of the number of nodes in the network minus the number of single-parent leaf nodes, i.e., nodes with an in-degree of 1 and an out-degree of 0. This will exclude, if P $\neq$ NP, ratio bounds which are polynomial functions of, for example,

- the number of cycles in the network,
- the length of the longest cycle, and
- the depth of the network.

These are excluded because single-parent leaves do not participate in cycles and can only increase the depth by one.

Definition 7. Let B be a belief network defined over a set of nodes V. The set $\Upsilon(B)$ will be defined as

$$\Upsilon(B) = V - \{v \in V\colon \mathit{indegree}(v) = 1,\ \mathit{outdegree}(v) = 0\}. \tag{40}$$

Theorem 8. *Let $\rho(\cdot)$ be an arbitrary positive polynomial function. If* P $\neq$ NP, *a polynomial-time approximation algorithm with ratio-bound $\rho(|\Upsilon(B)|)$ does not exist for the MAP problem.*

Given an instance (U, C) of NN3SAT, we apply Construction 2 with one modification; instead of (5), set

$$d = 1, \tag{41}$$

and let B' and $\mathcal{E}'$ be the belief network and evidence set that are produced. This can be carried out in polynomial time since d is constant. Note that B' will have a single penalty node and this penalty node will be a single-parent leaf node.

Then, apply Construction 2 again to (U, C), this time with

$$d = \left\lceil \log \rho\big(\left|\Upsilon(B')\right|\big)\right\rceil + 1, \tag{42}$$

to produce B and $\mathcal{E}$. Since ρ is a polynomial function, and $|\Upsilon(B')|$ is polynomial in the size of (U, C), this second application of Construction 2 can also be carried out in polynomial time. Because B' differs from B only in the number of penalty nodes, which are all single-parent leaf nodes,

$$\rho\big(\left|\Upsilon(B')\right|\big) = \rho\big(\left|\Upsilon(B)\right|\big). \tag{43}$$

Therefore, if $\mathcal{A}'$ is the assignment returned by an approximation algorithm with ratio bound $\rho(|\Upsilon(B)|)$, then (U, C) is satisfiable if and only if

$$P(\mathcal{A}') = \left(\tfrac{1}{2}\right)^n. \tag{44}$$

6.4. Concluding remarks

The results of this paper exclude, unless P = NP, the possibility of algorithms whose run-time behavior is guaranteed to be polynomial and who guarantee a certain quality of solution. However, they do leave open the possibility of methods, such as best-first search [4] and integer programming [20], which guarantee solution quality but not worst-case run-time (even though they may have good run-time in practice), and algorithms whose solution quality is not guaranteed even though they may produce good solutions in practice. Genetic algorithms [8], simulated annealing [9], and neural network [24] methods all fit in the latter category; there has been some work on applying these methods to belief networks [1,2,16–18] and they seem good candidates for further exploration.

Acknowledgement

The first author would like to thank his graduate student Mohamed Helal for helping with the proofreading of, and with drawing figures for, this paper, and for useful discussions.

References

[1] A.M. Abdelbar, S.M. Hedetniemi, A parallel hybrid genetic algorithm simulated annealing approach to finding most probable explanations on Bayesian belief networks, in: Proceedings IEEE International Conference on Neural Networks, Vol. I, 1997, pp. 450–455.

[2] A.M. Abdelbar, S.M. Hedetniemi, A recursive cubic-order neural network for generating probable explanations, in: Proceedings IMACS Congress on Computational Engineering in Systems Applications, 1998.

[3] E. Charniak, E. Santos Jr, Dynamic MAP calculations for abduction, in: Proceedings AAAI-92, San Jose, CA, 1992, pp. 552–557.

[4] E. Charniak, S. Shimony, Cost-based abduction and MAP explanation, Artificial Intelligence 66 (1994) 345–374.

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

[6] P. Dagum, M. Luby, Approximating probabilistic inference in Bayesian belief networks is NP-hard, Artificial Intelligence 60 (1993) 141–153.

[7] M. Garey, D. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, New York, 1979.

[8] D. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[9] S. Kirkpatrick, D. Gelatt, M. Vecchi, Optimization by simulated annealing, Science 220 (1983) 621–630.

[10] S. Lauritzen, D. Spiegelhalter, Local computations with probabilities on graphical structures and their applications to expert systems, J. Roy. Statist. Soc. Ser. B 50 (2) (1988) 154–227.

[11] L. Lovasz, On the ratio of optimal integral and fractional covers, Discrete Math. 13 (1975) 383–390.

[12] C. Papadimitriou, Computational Complexity, Addison-Wesley, Reading, MA, 1994.

[13] J. Pearl, Fusion, propagation, and structuring in belief networks, Artificial Intelligence 29 (1986) 241–288.

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

[15] J. Pearl, Belief networks revisited, Artificial Intelligence 59 (1993) 49–56.

[16] Y. Peng, J. Reggia, Plausibility of diagnostic hypotheses, in: Proceedings of Fifth National Conference on Artificial Intelligence (AAAI-86), Philadelphia, PA, 1986, pp. 140–145.

[17] C. Rojas-Guzman, M.A. Kramer, GALGO: a Genetic ALGOrithm decision support tool for complex uncertain systems modeled with Bayesian belief networks, in: Proceedings Ninth Conference on Uncertainty in AI (UAI-93), Washington, DC, 1993, pp. 368–375.

[18] C. Rojas-Guzman, M.A. Kramer, Remote diagnosis and monitoring of complex industrial systems using a genetic algorithms approach, in: Proceedings IEEE International Symposium on Industrial Electronics, 1994, pp. 363–367.

[19] E. Santos Jr, On the generation of alternative explanations with implications for belief revision, in: Proceedings Seventh Conference on Uncertainty in AI (UAI-91), Los Angeles, CA, 1991, pp. 339–347.
[20] E. Santos Jr, A fast hill-climbing approach without an energy function for probabilistic reasoning, in: Proceedings IEEE International Conference on Tools with AI, 1993, pp. 170–179.
[21] E. Santos Jr, A linear constraint satisfaction approach to cost-based abduction, Artificial Intelligence 65 (1994) 1–27.
[22] R.D. Shachter, Evaluating inference diagrams, Oper. Res. 34 (1986) 871–882.
[23] S.E. Shimony, Finding MAPs for belief networks is NP-hard, Artificial Intelligence 68 (1994) 399–410.
[24] J. Zurada, Introduction to Artificial Neural Systems, West Publishing, St. Paul, MN, 1992.