International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems
Vol. 9, No. 4 (2001) 461–489

SIMPLIFYING EXPLANATIONS IN BAYESIAN BELIEF NETWORKS

LUIS M. de CAMPOS, JOSE A. GÁMEZ* and SERAFÍN MORAL
Departamento de Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada, 18071, Granada, Spain
E-mail: {lci,smc}@decsai.ugr.es
**Departamento de Informática*
Universidad de Castilla-La Mancha
02071, Albacete, Spain
E-mail: jgamez@info-ab.uclm.es

Received September 1999
Revised March 2001

Abductive inference in Bayesian belief networks is intended as the process of generating the K most probable configurations given an observed evidence. These configurations are called *explanations* and in most of the approaches found in the literature, all the explanations have the same number of literals. In this paper we propose some criteria to simplify the explanations in such a way that the resulting configurations are still accounting for the observed facts. Computational methods to perform the simplification task are also presented. Finally the algorithms are experimentally tested using a set of experiments which involves three different Bayesian belief networks.

Keywords: Abductive inference, Bayesian belief networks, most probable explanation, probabilistic reasoning, simplicity criteria.

1. Introduction

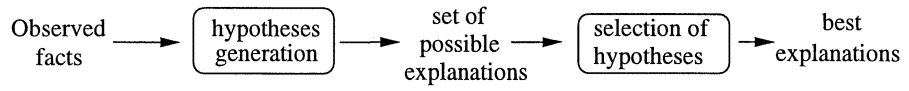
In the last years artificial intelligence researchers have devoted increasing attention to the development of abductive reasoning methods in a wide range of applications. Probably the most clear application of abductive reasoning is in the field of diagnosis, [23,24,27,28] although other applications exist in natural language understanding,[5,35] vision,[16] legal reasoning,[37] plan recognition,[2,15] planning,[25] and learning.[21]

Abduction is defined as the process of generating a plausible explanation for a given set of observations or facts.[26] This kind of reasoning can be represented by the following inference rule:

$$\frac{\psi \rightarrow \omega, \omega}{\psi},$$

i.e., if we observe ω and we have the rule $\psi \rightarrow \omega$, then we can infer that ψ is a *plausible* hypothesis (or explanation) for the occurrence of ω.

In general, there are several possible abductive hypotheses and it is necessary to choose among them. Therefore, we can divide the abductive task in two phases:



In order to select the best explanations from the generated set, two kinds of criteria are used: (1) metric based criteria (probability, weight, ...) and (2) simplicity criteria (the preferred explanation is the simplest available hypothesis). Usually *simplicity* is interpreted as logical simplicity, which means that those hypotheses with less different predicates are preferred (*Occam's razor*).

We think that the important role played by *simplicity* in the framework of abductive inference in logic has not been taken into account in the framework of abductive inference in Bayesian belief networks. In this paper simplicity criteria that can be used in the context of Bayesian belief networks are studied.

The paper is organized as follows: In the second section we introduce abductive inference in the framework of Bayesian belief networks (BBN). In the third section, we propose two kinds of simplification criteria. In the fourth section computation issues are studied. In the fifth section we study how to take advantage of the relations presented in the graph in order to optimize the simplification process. The experimental evaluation is presented in the sixth section. Finally, in the seventh section, we consider the conclusions.

2. Abductive Inference and Bayesian Belief Networks

A *Bayesian belief network* (Pearl,[22]) is a directed acyclic graph (DAG) where each node represents a random variable, and the topology of the graph shows the (in)dependence relations among the variables. The quantitative part of the model is given by a probability distribution for each node conditioned to its parents. If $X_{\mathcal{U}} = \{X_1, \ldots, X_n\}$ is the set of variables in the network, then the joint probability can be calculated as:

$$P(X_{\mathcal{U}}) = \prod_{X_i \in X_{\mathcal{U}}} P(X_i | pa(X_i)), \tag{1}$$

where $pa(X_i)$ contains the parents of X_i.

Before we continue, we define the following notation. A lower case subscript indicates a single variable (e.g., X_i). An upper case subscript indicates a set of variables (e.g., X_I). For some particular problems, the propositional variables are denoted by capital letters without subscript $A, B, C, \ldots$. The state taken by a variable X_i will be denoted by x_i, and the configuration of states taken by a set of variables X_D will be denoted by x_D. That is, capital letters are reserved for variables and set of variables, and lower case letters are reserved for states and configurations of states.

Given a set of observations[1] x_O for a set of variables X_O, propagation algorithms allow us to calculate $P(X_i|x_O)$ for every $X_i \in X_\mathcal{U} \setminus X_O$. The calculations are carried out in a secondary structure (obtained from the original BBN) called *junction tree*[2], where the evidence x_O has been entered. The propagation method[3] is based on the use of two operations: *marginalization* (addition) and *combination* (multiplication); and it is divided into two phases: *collectEvidence* (messages are passed from leaves to root) and *distributeEvidence* (messages are passed from root to leaves). See the books of Jensen and Shafer for details.[12,30]

In the context of BBNs an explanation for a set of observations $X_O = x_O$ is a configuration of states for the network variables, $x_\mathcal{U}$, such that, $x_\mathcal{U}$ is consistent with x_O, that is, $x_\mathcal{U}^{\downarrow X_O} = x_O$ (by $x_\mathcal{U}^{\downarrow X_O}$ we are denoting the configuration obtained from $x_\mathcal{U}$ by removing the literals not in X_O). In fact, the explanation is $x_\mathcal{U}^{\downarrow X_\mathcal{U} \setminus X_O}$, because the values taken by the variables in X_O are previously known. Given the large number of possible explanations and since we are interested in the best explanation, our goal will be to obtain the *most probable explanation.* Thus, abductive inference in BBNs (Pearl,[22]) corresponds to finding the maximum a posteriori probability state of the network, given the observed variables (the evidence). In a more formal way: if X_O is the set of observed variables and X_U is the set of unobserved variables, then we aim to obtain the configuration x_U^* of X_U such that:

$$x_U^* = \arg\max_{x_U} P(x_U|x_O), \tag{2}$$

where $X_O = x_O$ is the observed evidence. Usually, x_U^* is known as the *most probable explanation* (MPE), and in general we are interested in the K most probable explanations (K MPEs).

As in the case of computing marginals, finding the most probable explanation is an NP-hard problem [33].

Sometimes we are interested in obtaining the K MPEs only for a subset of the network's variables called *explanation set.*[19] This problem is known as *Partial Abductive Inference* and we think that in practical applications is more interesting than the classical abductive inference problem, because we can select as the explanation set those variables representing diseases in a medical diagnosis problem, variables representing critical components (starter, battery, alternator, ...) in a car diagnosis problem, etc.

Now, if we denote by $X_E \subset X_U$ the explanation set, then we aim to obtain the configuration x_E^* of X_E such that:

$$x_E^* = \arg\max_{x_E} P(x_E|x_O) = \arg\max_{x_E} \sum_{x_R} P(x_E, x_R|x_O), \tag{3}$$

[1] $X_O = x_O$ is known as *evidence*

[2] The nodes of a junction tree are known as *cliques* and contain more than one variable. A *separator* exists between two cliques and is obtained as the intersection between the variables of the two cliques.

[3] This probabilities propagation algorithm is known as HUGIN architecture.[14]

where $X_R = X_U \setminus X_E$. In general, x_E^* is not equal to the projection of configuration x_U^* over the variables of X_E. Therefore, we need to obtain x_E^* directly (eq. 3).

The MPE x_U^* can be found using the *probabilities propagation* method replacing addition by maximum in the marginalization operator.[6] To obtain the K MPEs, more complex methods must be used.[18,20,29]

The process of finding the MPE x_E^* is more complex than that of finding x_U^* because not all junction trees obtained from the original BBN are valid. In fact, because summation and maximum have to be used simultaneously and these operations do not show a commutative behaviour, the variables of X_E must form a sub-tree of the complete junction tree. We can deal with this problem in two ways:

- *Adapting a* standard *junction tree.* Xu[38] has proposed a method for transforming the initial junction tree into another one containing a node in which the variables of X_E are included. The problem is that if X_E contains many variables, then the size of the probability table associated with the node containing X_E will be too large. Nilsson[20] outlines how to slightly modify Xu's algorithm in order to allow (when possible) that variables in X_E constitute a sub-tree and not a single node.

- *Building a junction tree specific for* X_E. The construction of a junction tree is based on the triangulation of an undirected graph. In partial abductive inference, to obtain a valid junction tree, instead of searching for arbitrary deletion sequences, we can only consider sequences in which the variables in X_R come before the variables in X_E[9,8].

De Campos et al.[8] have shown that in both cases the size of the obtained junction tree grows significantly[4] in relation with the size of the junction tree obtained without restrictions, and so the propagation algorithm for partial abductive inference will be less efficient than propagation algorithms for (*total*) abductive inference. An approximate method based on genetic algorithms has been proposed by de Campos et al.[7] This procedure has shown quite good results when applied to the problem of searching for the K most probable partial explanations.

3. Simplicity Criteria

As we have seen in the previous section, when abductive inference is carried out in BBNs a *metric* is used to select the best explanations, concretely the explanations are ranked by their *a posteriori probability.* However, no simplicity criterion is applied. An immediate consequence is the fact that all explanations[5] have the same

[4] As an *extreme* example, consider a BBN with eleven variables $\{X_1, \ldots, X_{10}, Y\}$, such that there is a link $Y \rightarrow X_i$ for each variable X_i. If all the variables can take 10 different states, then the size of the *optimum* junction tree obtained in order to apply probabilities propagation or (total) abductive inference is $1,000$, while the size of the junction tree obtain for partial abductive inference taking $X_E = \{X_1, \ldots, X_{10}\}$ is 10^{11}.

[5] As (complete) abductive inference is a particular case of partial abductive inference taking $X_E = X_U$, in the rest of the paper we are not going to do any distinction.

number of literals ($|X_E|$). Thus, if we have observed, for example, *car does not start* and our explanation set contains critical components of the car, we can get

battery=dead, alternator=ok, starter=ok, engine cranks=ok, etc

as a MPE. However, the explanation *battery=dead* could be enough to account for the observation and it is simpler than the previous one. Therefore, our goal is to simplify the explanations obtained by the application of the methods cited in Section 2, by removing from the explanation those literals that are not important given the evidence. The process can be represented as follows

and can be stated in a more formal way as:

> Let $expl(x_O) = \{x_E^1, x_E^2, \ldots, x_E^K\}$ be the K MPEs obtained for the evidence $X_O = x_O$. Then, for all $x_E \in expl(x_O)$ we are looking for a sub-configuration x'_E ($X'_E \subset X_E$), such that x'_E is still accounting for the observed evidence.

It is important to remark that our procedure has two differentiated steps: first we generate complete explanations with values for all the variables of the explanation set and ordered by their 'a posteriori' probability given the observations; in a second stage each explanation is simplified by removing *unimportant* literals.

In the literature, we can find an alternative approach to this problem.[32,34] In these papers Shimony works with partial abduction without taking an explanation set, but trying to identify the relevant nodes directly, without previously determining an explanation set. The relevant nodes include the evidence nodes in the network and only ancestors of evidence nodes can be relevant (see [32,34] for details). In our opinion the advantage of Shimony's method is that it does not need two steps in the inference process, i.e., the method directly generates simplified explanations. This approach favours simple explanations as, if $X'_A = x'_A$ is a sub-configuration of $X_A = x_A$, then the probability of the sub-configuration is greater than the probability of the initial configuration (it contains less variables with the same assigned values). In order to prevent that the empty explanation is always obtained, a necessary condition is given for a simplification to be possible. In most of the cases, given that an assignation $X_i = x_i$ is included in the explanation set, then it requires that all the parents of this variable $pa(X_i)$ are included in the explanation set, with their corresponding assignations, except for variables X_j in $pa(X_i)$ such that known the value of the parents in the explanation set, the event $X_i = x_i$ is independent of X_j. This is a really strong condition and, as Chajewska and Halpern, point out in their paper,[4] the explanations obtained by this method can have too many variables, because it is not difficult to see that for each evidence node X_i, the explanation must include an assignment to all the nodes in at least one path from X_i to a root in

the DAG (see Shimony,[32,34] for details). But, in most of the cases all the ancestors are included in the explanation set. Another limitation of Shimony's procedure is that it always considers that the set of relevant variables (in which the reduced explanation will be searched) is the set of ancestors of the observed variables. We think that this can be reasonable in some applications, but there are situations in which it is not. Assume, for example that arcs have not a causal interpretation and that the graph is representing independences. In this situation, we can have equivalent graphs representing the same problem with different orientations of the arcs. In these equivalent graphs the relevant variables will be different. Another question is that the result will depend of considering or not intermediate variables. For example, assume that we have a link from X_i to X_j and both variables are in the relevant set. If we refine the model by considering a variable X_k with a link from X_i to X_k and a link from X_k to X_j, then X_k will be in the relevant set. This inclusion can produce that everything change, even the values of the rest of the variables. For these reasons, we have preferred that the user selects the explanation set or the relevant variables. These should be the diagnostic variables in which she is interested. The problem of discovering this set or helping the user to determine it is really interesting, but we do not have a simple answer to it and will not be considered in this paper.

The next subsections are devoted to introduce two criteria that can be used to decide when $x'_E \subset x_E \in expl(x_O)$ is still an explanation for x_O. We will denote by $x'_E \subset x_E$, the fact that x'_E is obtained from x_E by removing one or more literals; and $|x_E|$ will denote the number of literals in x_E.

3.1. *Independence Based Criteria*

Suppose we can divide our initial explanation x_E in two parts, x_D and x_I ($X_D \cup X_I = X_E$), such that, if we know x_D then adding x_I to our knowledge does not modify our belief on the presence of the evidence (x_O). Thus, x_D explains the presence of x_O as well as x_E, so we can say that x_I is irrelevant for the observed evidence and therefore its literals can be removed from the explanation. From the probabilistic point of view, we can express this idea as follows:

$$P(x_O|x_D) = P(x_O|x_D, x_I), \tag{4}$$

and this means to interpret irrelevance as statistical independence. Following this idea we can give the next definition of simplification:

Definition 1. (I-simplification)
We say that $x'_E \subset x_E$ is an *Independence based simplification* (I-simplification) of $x_E \in expl(x_O)$, if and only if, $P(x_O|x'_E) = P(x_O|x_E)$.

In order to relax the previous definition the term *equal* can be replaced by *almost equal.* The term *almost equal* can be made precise by means of a threshold

$\epsilon \in [0.01, 0.05]$. Thus, we have the following definition:

Definition 2. (I~simplification)
We say that $x'_E \subset x_E$ is an *Independence based simplification* (I~simplification) of $x_E \in expl(x_O)$, if and only if,

$$(1-\epsilon)P(x_O|x_E) \leq P(x_O|x'_E) \leq (1+\epsilon)P(x_O|x_E) \tag{5}$$

Before going on with our study, it is convenient to say that definitions similar to the previously formulated can be found in works devoted to *sensitivity analysis in Bayesian networks.*[13,36] However, our idea should be interpreted in the opposite sense, because sensitivity analysis studies how sensitive is the conclusion (hypothesis) with respect to the set of observations, analyzing which items of evidence are *in favor of/against/irrelevant for* the conclusion. In our case, the evidence is the only thing we consider previously fixed, and our goal is to analyze which subsets of the hypothesis are still an explanation for the given evidence.

Taking up again our definition of I~simplification, we can see that some explanations can not be simplified, but others can have more than one simplification. In the last case, we aim to obtain the *best* possible simplification, i.e., the simplification with the smallest number of literals. The following definition formalizes this idea and uses probability to break ties:

Definition 3. (Best I~simplification)
We say that $x'_E \subset x_E$ is the *best independence based simplification* (I~simplification) of $x_E \in expl(x_O)$, if and only if, the following conditions hold:

1. x'_E is an I~simplification of x_E.
2. $\nexists\ x''_E \subset x_E$, such that x''_E is an I~simplification of x_E and $|x''_E| < |x'_E|$.
3. If x''_E is an I~simplification of x_E and $|x''_E| = |x'_E|$ then the following expression is true: $abs(P(x_O|x''_E) - P(x_O|x_E)) \geq abs(P(x_O|x'_E) - P(x_O|x_E))$.

where *abs* denotes the absolute value function.

In Section 4 we will talk about computational aspects of this criterion, but now we are going to give another independence based criterion.

3.2. *Relevance Based Criteria*

Now, our idea is the following: Let x'_E be an I~simplification of x_E, then we have removed the literals in $X_E \setminus X'_E$ because they were (almost) irrelevant for the observed evidence. But, what happen when $P(x_O|x_E)$ is not (almost) equals to $P(x_O|x'_E)$?. We can distinguish two cases:

1. $P(x_O|x'_E) < P(x_O|x_E)$. This can be interpreted as the sub-configuration x'_E accounts for x_O in smaller degree than the original explanation.

2. $P(x_O|x'_E) > P(x_O|x_E)$. This can be interpreted as the sub-configuration x'_E account for x_O in greater degree than the original explanation.

Therefore, in the second case we think that x'_E would be accepted as a simplification of x_E, because it accounts for the observed evidence at least as x_E, and it has a smaller number of literals. This idea can be formalized by the following definition:

Definition 4. (R~simplification)
We say that $x'_E \subset x_E$ is a *Relevance based simplification* (R~simplification) of $x_E \in expl(x_O)$, if and only if,

$$(1-\epsilon)P(x_O|x_E) \leq P(x_O|x'_E) \tag{6}$$

We have called this criterion *relevance* based simplification because the removed literals are not irrelevant to the evidence, against what happens when using I~simplification. The *best* R~simplification can be defined similarly to definition 3, breaking ties in favor of the sub-configuration x'_E with greatest probability $P(x_O|x'_E)$. Furthermore, it is clear that if x'_E is an I~simplification of x_E then it is also a R~simplification of x_E.

3.3. *Examples*

In this subsection we give two simple examples in order to show the intuition of the simplifications provided by the previous definitions. They will show two different situations in which the behaviour of the simplifications has some particularities. The first one is a case in which we have two alternative causes for an unusual finding. The second one will present a situation in which we have two consecutive causes for an observation, being one of the causes independent of the observations given the other cause.

Example 1. Let us assume the network given by the graph $G = (\{A, B, C\}, \{A \rightarrow C, B \rightarrow C\})$, where each variable is bivalued, and the following conditional probabilities:

$P(\mathtt{a}) = 0.1$	$P(\mathtt{c}\|\mathtt{ab}) = 0.95$	$P(\overline{\mathtt{c}}\|\mathtt{ab}) = 0.05$
$P(\overline{\mathtt{a}}) = 0.9$	$P(\mathtt{c}\|\mathtt{a}\overline{\mathtt{b}}) = 0.9$	$P(\overline{\mathtt{c}}\|\mathtt{a}\overline{\mathtt{b}}) = 0.1$
$P(\mathtt{b}) = 0.01$	$P(\mathtt{c}\|\overline{\mathtt{a}}\mathtt{b}) = 0.92$	$P(\overline{\mathtt{c}}\|\overline{\mathtt{a}}\mathtt{b}) = 0.08$
$P(\overline{\mathtt{b}}) = 0.99$	$P(\mathtt{c}\|\overline{\mathtt{a}}\overline{\mathtt{b}}) = 0.001$	$P(\overline{\mathtt{c}}\|\overline{\mathtt{a}}\overline{\mathtt{b}}) = 0.999$

This is a very common situation in which we have two competing causes A and B for an unusual effect C. This effect is rarely present if none of the causes is active. The 'a priori' probability of a cause is low too (0.1 for a and 0.01 for b).

Assume now that we have observed c and that the explanation set is $X_E = \{A, B\}$. In these conditions the three best explanations are (in this order): $\mathtt{a}\overline{\mathtt{b}}, \overline{\mathtt{a}}\mathtt{b}, \mathtt{ab}$, with probabilities $P(\mathtt{a}\overline{\mathtt{b}}|\mathtt{c}) = 0.8980, P(\overline{\mathtt{a}}\mathtt{b}|\mathtt{c}) = 0.0835, P(\mathtt{ab}|\mathtt{c}) = 0.0096$. Of

course, there is a big difference between the first and the rest of more probable configurations (which is mainly due to its higher 'a priori' probability), so that, $\mathtt{a}\overline{\mathtt{b}}$ is the true best explanation. However, let us proceed to simplify the three more probable configurations. Taking into account that $P(\mathtt{c}|\mathtt{a}\overline{\mathtt{b}}) = 0.9$ and $P(\mathtt{c}|\mathtt{a}) = 0.9005$, $\mathtt{a}\overline{\mathtt{b}}$ is simplified to $\mathtt{a}$. In fact, $\overline{\mathtt{b}}$ does not add anything to $\mathtt{a}$ as an explanation to $\mathtt{c}$. In an analogous way, $\overline{\mathtt{a}}\mathtt{b}$ is simplified to $\mathtt{b}$, though in this case a higher ϵ (0.02) is necessary for the I~simplification. With respect to $\mathtt{ab}$, a higher value of ϵ (0.05) is required in order to simplify it to $\mathtt{b}$; in this case we can see how removing one of the causes decreases the probability of the observations. □

Example 2. Let us assume the network given by the graph $G = (\{A, B, C\}, \{A \rightarrow B, B \rightarrow C\})$, where C is conditionally independent of A given B. Consider that each variable is bivalued with the following conditional probabilities:

$P(\mathtt{a}) = 0.1$	$P(\mathtt{b}\|\mathtt{a}) = 0.95$	$P(\overline{\mathtt{b}}\|\mathtt{a}) = 0.05$
$P(\overline{\mathtt{a}}) = 0.9$	$P(\mathtt{b}\|\overline{\mathtt{a}}) = 0.05$	$P(\overline{\mathtt{b}}\|\overline{\mathtt{a}}) = 0.95$
	$P(\mathtt{c}\|\mathtt{b}) = 0.99$	$P(\overline{\mathtt{c}}\|\mathtt{b}) = 0.01$
	$P(\mathtt{c}\|\overline{\mathtt{b}}) = 0.01$	$P(\overline{\mathtt{c}}\|\overline{\mathtt{b}}) = 0.99$

Assume as above that we have observed c and that the explanation set is $X_E = \{A, B\}$. In these conditions the first two explanations are $\mathtt{ab}$, with $P(\mathtt{ab}|c) = 0.6389$, and $\overline{\mathtt{a}}\mathtt{b}$ with $P(\overline{\mathtt{a}}\mathtt{b}|c) = 0.3026$. In both cases, with any of the criteria, both explanations are simplified to $\mathtt{b}$. The reason is that the conditional independence implies that the probability of the observations given $\mathtt{b}$ is independent of the value of A.

It is important to observe as in the case of consecutive causes (A is a cause of B and B is a cause of C) of some observations ($\mathtt{c}$ in this case), these criteria select the immediate causes ($\mathtt{b}$), removing the primary or deep causes ($\mathtt{a}$), as the observations do not depend of them given the immediate causes. As a more intuitive example, assume that a car does not start and we have as possible explanations that there is no petrol, and as a cause of this that the tank has a leak. If these two explanations are chosen as most probable explanations, these criteria will simplify the explanation to *there is no petrol*, discarding the fact that *there is a leak in the tank*. Of course, sometimes we are more interested in these primary causes. Gámez,[9] has proposed an iterative procedure in which, after each simplification, the immediate (non simplified) cause is added to the observations and removed from the set of explanations, repeating everything again. The procedure finishes when there are no more variables in the set of explanations. In this way, the consecutive explanations of a set of observations are obtained, starting from the immediate causes to the primary ones.

□

4. Computation

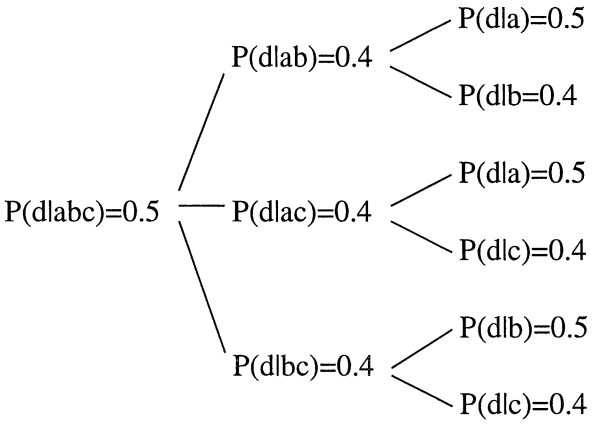
In this section we have to deal with the problem of simplifying explanations from a computational perspective. From this point of view, one of the main problems is

the fact that the best simplification cannot be found (in an exact way) by means of a search in which a literal is removed at each step, because the independence based criteria does not have a monotonicity property, as we can see in the following example.

Example 3. Let us to consider the network given by the graph $G = (\{A, B, C, D\}, \{A \to D, B \to D, C \to D\})$ and the conditional probabilities shown in the following Table (all the variables can take two states).

$P(\mathtt{a}) = 0.5$	$P(\mathtt{d}\|\mathtt{abc}) = 0.5$	$P(\overline{\mathtt{d}}\|\mathtt{abc}) = 0.5$
$P(\overline{\mathtt{a}}) = 0.5$	$P(\mathtt{d}\|\mathtt{ab}\overline{\mathtt{c}}) = 0.3$	$P(\overline{\mathtt{d}}\|\mathtt{ab}\overline{\mathtt{c}}) = 0.7$
	$P(\mathtt{d}\|\mathtt{a}\overline{\mathtt{b}}\mathtt{c}) = 0.3$	$P(\overline{\mathtt{d}}\|\mathtt{a}\overline{\mathtt{b}}\mathtt{c}) = 0.7$
$P(\mathtt{b}) = 0.5$	$P(\mathtt{d}\|\mathtt{a}\overline{\mathtt{b}}\overline{\mathtt{c}}) = 0.9$	$P(\overline{\mathtt{d}}\|\mathtt{a}\overline{\mathtt{b}}\overline{\mathtt{c}}) = 0.1$
$P(\overline{\mathtt{b}}) = 0.5$	$P(\mathtt{d}\|\overline{\mathtt{a}}\mathtt{bc}) = 0.3$	$P(\overline{\mathtt{d}}\|\overline{\mathtt{a}}\mathtt{bc}) = 0.7$
	$P(\mathtt{d}\|\overline{\mathtt{a}}\mathtt{b}\overline{\mathtt{c}}) = 0.5$	$P(\overline{\mathtt{d}}\|\overline{\mathtt{a}}\mathtt{b}\overline{\mathtt{c}}) = 0.5$
$P(\mathtt{c}) = 0.5$	$P(\mathtt{d}\|\overline{\mathtt{a}}\overline{\mathtt{b}}\mathtt{c}) = 0.5$	$P(\overline{\mathtt{d}}\|\overline{\mathtt{a}}\overline{\mathtt{b}}\mathtt{c}) = 0.5$
$P(\overline{\mathtt{c}}) = 0.5$	$P(\mathtt{d}\|\overline{\mathtt{a}}\overline{\mathtt{b}}\overline{\mathtt{c}}) = 0.5$	$P(\overline{\mathtt{d}}\|\overline{\mathtt{a}}\overline{\mathtt{b}}\overline{\mathtt{c}}) = 0.5$

Let $D = d$ be the evidence and $X_E = \{A, B, C\}$ be the explanation set. Then, abc is the second most probable explanation with $P(\mathtt{abc}|\mathtt{d}) = 0.133$. In the tree obtained when all the sub-configurations of abc are considered, we can see that ab is not an independence (or relevance) based simplification for abc, while its sub-configuration a is a valid simplification.



□

A property saying that if a sub-configuration x'_E is a relevance (independence) based simplification of x_E and there are two or more literals of x_E not in x'_E, then there is a relevance (independence) based simplification x''_E intermediate between X_E and X'_E, would allow to design an algorithm in which the simplifications of x_E

are carried out by removing one literal at each step. If in a given moment no single literal can be removed producing a simplification, then we would be sure that there is no simplification that can be obtained by removing more than one literal and we can stop. Unfortunately, such a property is not verified as the example above shows, and to the best of our knowledge the complete search space of all the sub-configurations should be explored if we want to be sure that the obtained simplification is exactly the *best* independence (or relevance) based simplification[6] To do this, $P(x_O|x'_E)$ for each $x'_E \subset x_E$, must be calculated through a propagation in the junction tree, and given that the number of sub-configurations grows exponentially with the number of variables in the explanation set, and that each computation of $P(x_O|x'_E)$ can require a probabilistic propagation, the process, in general, would be intractable. So, the search cannot be done in an exact way, and so approximate search methods have to be considered. These methods will try to use previous computations associated to configurations already evaluated, when we try to calculate the probability $P(x_O|x'_E)$ for a given configuration.

We are going to use two alternatives: cautious propagation and a heuristic method.

4.1. *Using Cautious Propagation*

In sensitivity analysis the same problem exists,[13] and the solution adopted by Jensen et al. was to use a modified scheme of inference in junction trees called *cautious propagation* (Jensen,[11]).

Cautious propagation is a modification of HUGIN propagation into a Shafer-Shenoy-like architecture[31], in which each separator in the junction tree stores two messages, and the probability tables of cliques are not modified during the propagation task. The method is called *cautious* because it does not change any probability table in the junction tree. Cautious propagation is less efficient than HUGIN, but given a configuration, x_H, and a configuration of observations x_O, it provides access to $P(x'_O|x_H)$ for a great number of subsets x'_O of X_O (as the method was developed to solve sensitivity analysis, the configuration x_H represents an hypothesis, but this is not relevant to our problem).

$P(x'_O|x_H)$ is *accessed* means that this value can be obtained using probability tables calculated previously, without requiring extra propagations.

Jensen also suggests to use cautious entering of evidence,[11] that is, items of evidence are always entered in a leaf of the junction tree and at most one item ($x_o \in x_O$) is inserted in any node. In most cases, it is necessary to add dummy nodes to the junction tree in order to make possible cautious entering of evidence.

The combination of cautious propagation with cautious entering of evidence enlarges the number of $P(x'_O)$s accessed. In particular, it gives access to $x_O \setminus x_o$ and x_o for any item $x_o = x_O^{\downarrow X_i}$, for all $X_i \in X_O$.

[6]See Section 5 for some previous reductions of the search space, based on the independences represented by the graph

As $P(x_O|x'_E) = \frac{P(x'_E|x_O)P(x_O)}{P(x'_E)}$ we can use this method but carrying out two cautious propagations, that is:

1. Let T be a junction tree without entered evidence and let T_O be a junction tree in which evidence x_O has been entered.

2. Enter cautiously (as evidence) x_E in T and perform cautious propagation. This gives access to a set of $P(x'_E)$s.

3. Perform HUGIN propagation in T_O. After this propagation the tables of T_O are conditioned on x_O. Furthermore, $P(x_O)$ can be calculated in this propagation by summing in the probability table of the root node, after the collectEvidence phase has been carried out.

4. Enter cautiously (as evidence) x_E in T_O and perform cautious propagation. As probability tables in T_O were conditioned on x_O this propagation gives access to $P(x'_E|x_O)$ for the same subset of sub-configurations of x_E as in the second step.

5. Now we can calculate $P(x_O|x'_E) = \frac{P(x'_E|x_O)P(x_O)}{P(x'_E)}$ for all x'_E accessed, and the best simplification is obtained according to definition of best simplification.

The solution obtained by this method can be interpreted as *approximate*, because the search space is not completely explored and so we cannot be sure that the best simplification found is the true best simplification.

4.2. *Regressive search method*

In this section we are going to propose an alternative method to the previous one. It is based on an incremental heuristic search. We start with the complete explanation and we try to remove a literal at each step (this is the reason we have called it *regressive*). The procedure stops when it is not possible to obtain a simplification by deleting a single literal. The procedure is as follows:

1. $x_S \leftarrow x_E$ (the explanation to be simplified)

2. Let S be the set of sub-configurations of x_S obtained by removing only a literal from x_S.

3. Remove from S those elements not being an I$\sim$simplification of x_E.

4. If $S = \emptyset$ finish returning x_S as the best simplification.

5. If $S \neq \emptyset$ do $x_S = x'_E \in S$, such that, x'_E minimizes the expression $abs(P(x_O|x'_E) - P(x_O|x_E))$.

6. Go to step 2.

As we can see, in order to look for the R~simplification, only steps 3 and 5 have to be modified according to the R~simplification criterion.

Although we have seen in example 3 that this way to proceed does not guarantee the success of the search, from the experiments (Section 6) we can conclude that in general the method has a good behavior.

Using this method the number of evaluated sub-configurations is reduced from $2^{|X_E|}$ (exhaustive search) to[7] $\frac{|X_E|^2+|X_E|}{2}$. Furthermore, this is an upper bound because the explanation is not always simplified to a unique literal. As $P(x_O|x'_E) = \frac{P(x'_E,x_O)}{P(x'_E)}$, two *upward* propagations[8] are necessary in order to calculate this value (with and without entered evidence). However, the computational cost of the process still is too high, so we are going to deal with this problem in the next subsection.

4.2.1. *Evaluation of sub-configurations*

Our proposal now is to avoid the need of carrying out a complete upward propagation when a new sub-configuration has to be evaluated. The idea is as follows:

> When a sub-configuration x''_E has to be evaluated, it is clear that another very similar sub-configuration has been previously evaluated: a configuration x'_E such that x''_E can be obtained by deleting a literal (a value for a variable) from x'_E. So it is very probable that a great part of the computations involved in the new propagation had been calculated before. In order to avoid these repetitions we always will work over the same junction tree, storing in it all the messages previously sent.

If $x''_E \subset x'_E$ and $|x''_E| = |x'_E| - 1$, then we will refer to x'_E as the *father* of x''_E. Notice the fact that a sub-configuration x''_E can have more than one father, because different literals can be added to x''_E; however, in the context of regressive search, as only a branch of the tree of configurations is explored, then the father is clearly identified.

The process of evaluating an explanation and its sub-configurations requested by the regressive algorithm could be as follows:

- Enter cautiously (as evidence) x_E in the junction tree.
- Perform an upward propagation storing all the messages sent among the nodes of the junction tree.

[7] One configuration of size $|X_E|$, plus $|X_E|$ configurations of size $(|X_E| - 1)$, plus $(|X_E| - 1)$ configurations of size $(|X_E| - 2)$,..., plus 3 configurations of size 2, plus 2 configurations of size 1. Therefore, we have to evaluate $\frac{|X_E|\cdot(|X_E|+1)}{2} = \frac{|X_E|^2+|X_E|}{2}$ configurations.

[8] By *upward* propagation we are denoting the process of evaluating a configuration x_E. As we have seen in the previous subsection, to do this only the first step of HUGIN propagation is needed. Concretely, the process has three steps: (1) enter the configuration (as evidence) in the junction tree, (2) select a root and perform collectEvidence, and (3) $P(x_E)$ is calculated by summing in the probability table of the selected root.

- Let x''_E be the sub-configuration to be evaluated, and x'_E its father configuration (previously evaluated). Let C be the node used as root in the evaluation of x'_E and $H_1, \ldots, H_n$ the neighbours of C in the junction tree. Then, x'_E can be divided in $x_E^1 \cup \ldots \cup x_E^{i-1} \cup x_E^i \cup x_E^{i+1} \cup \ldots \cup x_E^n$ (see Figure 1.a).

  We know x''_E is equal to x'_E except in one literal l. Let us suppose that this literal is included in x_E^i. Then, $x''_E = x_E^1 \cup \ldots \cup x_E^{i-1} \cup x_E^{i*} \cup x_E^{i+1} \cup \ldots \cup x_E^n$, where x_E^{i*} is obtained from x_E^i by removing that literal. So, we can evaluate x''_E following one of the two options given below:

  - C is maintained as the root. Then C sends a message to H_i requesting the correct message (x_E^{i*}). If H_i has enough information the message is sent, otherwise the necessary information is requested from H_i to its neighbours (except C). This option corresponds with Figure 1.b. To implement this procedure each separator of the junction tree stores a list of messages.
  - The root is moved to H_i. In this case C builds the message $x_E^c = x_E^1 \cup \ldots \cup x_E^{i-1} \cup x_E^{i+1} \cup \ldots \cup x_E^n$, and sends it to H_i. If H_i has enough information then x''_E can be evaluated, otherwise the necessary information is requested from H_i to its neighbours (except C). This option corresponds with Figure 1.c. To implement this procedure each separator of the junction tree stores two lists of messages.

  Using one of the previous options, we only calculate a few new messages, saving a lot of time in computation. By contrast, it is clear that this way to proceed needs more memory, because each link stores one or two lists of messages instead of a single message. This raises the issue of space efficiency, although given that cautious entering of evidence is used, we think that if the clique selected as the root is chosen in an appropiate way, then the number of different messages to be stored in each separator will be moderated (as can be viewed in Figures 2, 3 and 4).

- When no more sub-configurations have to be evaluated, it is interesting to see that some additional sub-configurations are accessed. That is, they can be evaluated without sending more messages (as in cautious propagation).

  In most cases it is not necessary to evaluate all the accessed sub-configurations, because if we fix the order of evaluation from less to more literals, then if a simplification is found, there is not need to evaluate the sub-configurations with a greater number of literals.

As occurs in cautious propagation, two junction trees are necessary, one of them without entered evidence (to obtain $P(x_E)$'s) and the other with entered evidence (to obtain $P(x_E, x_O)$'s). After this process, $P(x_O|x_E)$'s can be calculated.

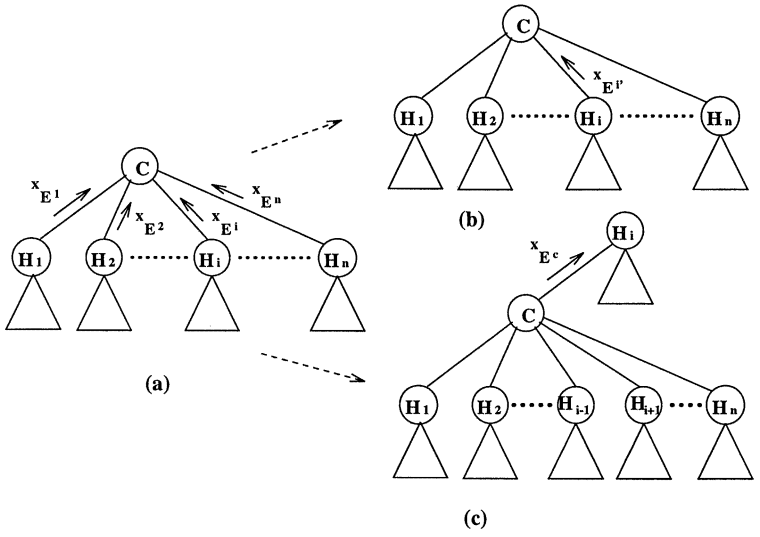


Fig. 1. Choosing a new root

The following example is an illustration of the previous algorithm.

Example 4. Figure 2.a shows a junction tree in which x_E = abcde has been entered cautiously, and the messages sent during the evaluation process (propagation). As $|x_E| = 5$, in the first stage of the regressive algorithm we have to evaluate five sub-configurations of length 4. The messages sent in those evaluations are shown in Figures 2.b to 2.f ([] denotes an empty message).

Let us consider abde as the best simplification of x_E in this stage. So, in the second stage of the search process four sub-configurations of length 3 have to be evaluated. The messages sent in these evaluations are shown in Figures 3.a to 3.d.

Let us suppose that no one of these sub-configurations is a simplification of x_E, so the best found simplification is abde. At this moment two observations can be done:

1. After evaluating x_E, only a few new messages are necessary to evaluate its sub-configurations.

2. In Figure 4 a summary of the messages sent in the complete process is shown. It is easy to see that the algorithm gives access to configurations de and cde in node 2 and configurations ab and abc are in node 5. As the length of these configurations is smaller than the length of the best simplification found, we can calculate their value and study if some of them constitutes the new best simplification.

Finally, some of the messages calculated in this process can be retained for the next configuration. For example, if the new configuration to simplify is abc$\overline{de}$, the

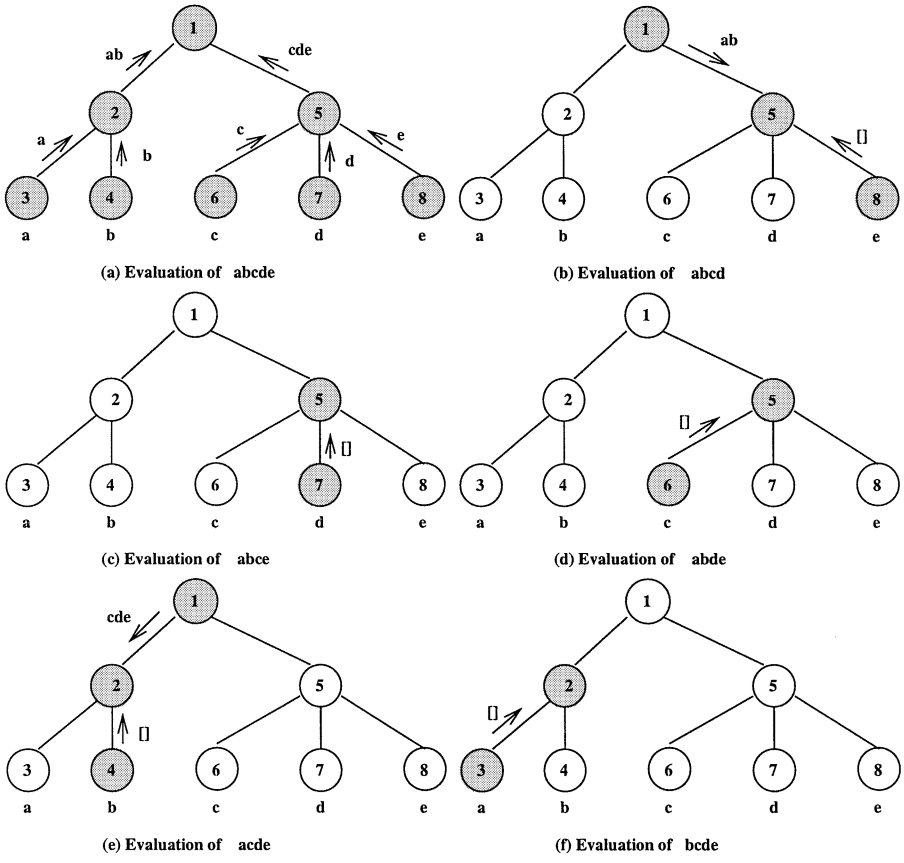


Fig. 2. Messages used in the evaluation of configuration `abcde` and its children. Notice the fact that after evaluating `abcde` only two new messages are sent in order to evaluate its children.

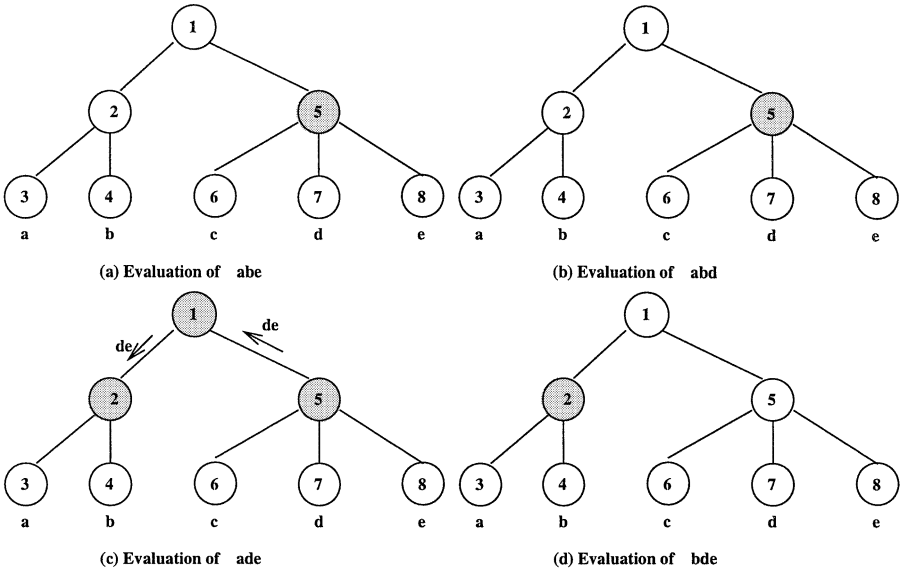


Fig. 3. Messages sent when evaluating children of sub-configuration abde

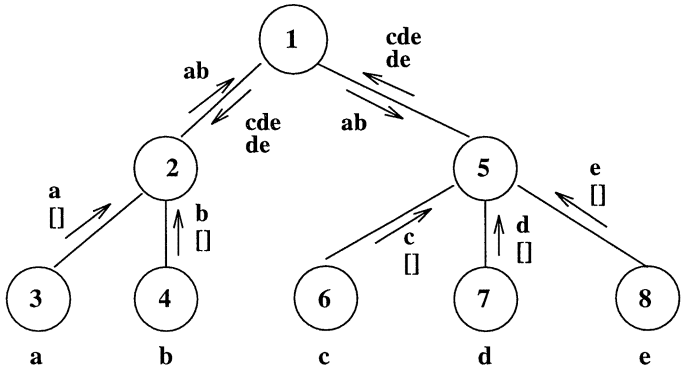


Fig. 4. Summary of calculated messages

process can start with the junction tree shown in (Figure 5) and not with an empty one.

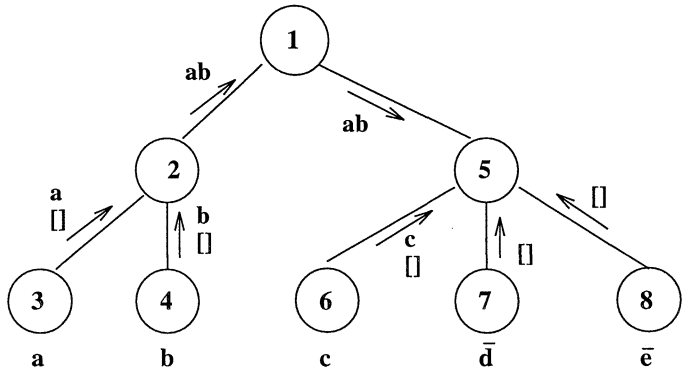


Fig. 5. Retained computations from configuration abcde to configuration $\mathtt{abc}\overline{de}$

□

5. Simplifications induced by the graph

In this section we try to take advantage of the graph topology in order to do an 'a priori' simplification. That is, before to carry out any probabilistic propagation, all the configurations will be simplified by removing some literals (corresponding to the same variables).

To obtain the set of literals to be removed, we study whether the explanation set X_E can be divided in two disjoint sets X_D and X_I, such that the evidence (X_O) is independent from X_I given X_D (this independence statement is usually denoted as $I(X_O|X_D|X_I)$). The previous statement implies the independence $I(x_O|x_D|x_I)$ for all configurations x_O, x_D, x_I of X_O, X_D and X_I respectively, so $x_E^{\downarrow X_D}$ is a simplification of x_E for all MPE x_E.

As we have stated in the first paragraph, our aim is to identify X_D and X_I (with X_D being a minimal set) by analyzing the directed acyclic graph (DAG). The key concept in this analysis is the *d-separation* criterion (Pearl,[22]), because we know that if X_O is d-separated from X_I given X_D in the graph G (denoted by $< X_O|X_D|X_I >_G^d$) then we can conclude $I(X_O|X_D|X_I)$. Therefore, we can obtain X_D and X_I from the graph G.

Although efficient algorithms have been developed for determining d-separation in graphs[10], as some authors have pointed out,[1] the d-separation criterion is difficult to manage and is rather subtle. So, we will transform the problem into an equivalent one, in which the use of d-separation is avoided, being replaced by a more 'uniform' criterion like *separation* in undirected graphs. To do so, we follow the algorithm proposed by Lauritzen et al.[17] It is based on the fact that if X_A, X_B and X_C are three disjoint subsets of nodes in G, then X_A is d-separated from X_B given X_C, if

and only if, X_A is separated from X_B given X_C in the graph $(G^{An(X_A \cup X_B \cup X_C)})_M$, where $(G^{An(X_A \cup X_B \cup X_C)})_M$ is the graph resulting from the moralization of the graph induced by the smallest ancestral set in G containing $X_A \cup X_B \cup X_C$:

$$< X_A|X_C|X_B >_G^d \Longleftrightarrow < X_A|X_C|X_B >_{(G^{An(X_A \cup X_B \cup X_C)})_M}^s \tag{7}$$

In our case $X_A \cup X_B \cup X_C$ is equal to $X_E \cup X_O$ and this knowledge is enough to obtain the ancestral graph (a minimal graph containing $X_E \cup X_O$ and all their ancestors in the graph). After this step we moralize the graph by adding edges among the parents of each node and making all the edges in the graph undirected).

Working over the undirected graph, the task to identify X_D is reduced to detect which nodes of X_E can be reached from some node of X_O, by using a path that does not contain any node of X_E. Finally X_I is obtained as the complementary set of X_D respect to X_E.

5.1. *'A priori' simplification of the explanation set*

As we have seen in the previous study, if $X_I \neq \emptyset$ (such that $I(X_O|X_D|X_I)$ holds) exists, then all the MPEs can be simplified by removing the elements corresponding to X_I, so "why not to do $X_E = X_D$ before obtaining the K MPEs?". If we do so, it is probable that the process of searching the K MPEs could be more efficient[9]. On the other hand, "is the best MPE for X_D the same as for X_E?". The following counter-example answers this question:

Example 6. Let us to consider the network given by the graph $G = (\{A, B, C\}, \{A \rightarrow B, B \rightarrow C\})$ and the conditional probabilities shown in the following table (all the variables can take two states):

$P(\mathtt{a}) = 0.3$	$P(\overline{\mathtt{a}}) = 0.7$
$P(\mathtt{b}\|\mathtt{a}) = 0.45$	$P(\overline{\mathtt{b}}\|\mathtt{a}) = 0.55$
$P(\mathtt{b}\|\overline{\mathtt{a}}) = 0.4$	$P(\overline{\mathtt{b}}\|\overline{\mathtt{a}}) = 0.6$
$P(\mathtt{c}\|\mathtt{b}) = 0.59$	$P(\overline{\mathtt{c}}\|\mathtt{b}) = 0.41$
$P(\mathtt{c}\|\overline{\mathtt{b}}) = 0.4$	$P(\overline{\mathtt{c}}\|\overline{\mathtt{b}}) = 0.6$

Let us to consider $X_E = \{A, B\}$, $X_O = \{C\}$ and $C = \mathtt{c}$ as the observed evidence. Under these considerations, the most probable explanation is the configuration $\overline{\mathtt{a}}\overline{\mathtt{b}}$ with probability 0.351. However, the independence $I(C|B|A)$ can be obtained from the graph, so we could take $X'_E = \{B\}$. With this new explanation set, the most probable explanation is $\mathtt{b}$ with probability 0.511. Therefore, we can see that the best MPE is not the same in both cases:

$$\overline{\mathtt{a}}\overline{\mathtt{b}}^{\downarrow B} = \overline{\mathtt{b}} \neq \mathtt{b}$$

[9]A direct relation between $|X_E|$ and the size of the junction tree has been established by Gámez.[9], and de Campos et al.[8]

□

In our opinion, the explanation set has been chosen by the user attending to some reasons, and this choice should be respected unless we can ensure that the best explanation is the same for X_E and X_D. Now, we are going to show that in some cases this equality can be warranted, concretely, when the independence $I(X_D|\emptyset|X_I)$ is also observed.

The well known *semigraphoid axioms*,[22] which are always verified by the independences associated with probability distributions, will be used:

- Symmetry: $I(X|Z|Y) \Longleftrightarrow I(Y|Z|X)$
- Decomposition: $I(X|Z|Y \cup W) \Longrightarrow I(X|Z|Y)$ & $I(X|Z|W)$
- Weak Union: $I(X|Z|Y \cup W) \Longrightarrow I(X|Z \cup W|Y)$
- Contraction: $I(X|Z|Y)$ & $I(X|Z \cup Y|W) \Longrightarrow I(X|Z|Y \cup W)$

where X, Y, Z and W are sets of variables.

Proposition 1. Let X_E be the explanation set and X_O the observed variables. Let X_D and X_I be two disjoint subsets of X_E such that $X_E = X_D \cup X_I$. Let us suppose the independence relations $I(X_I|X_D|X_O)$ and $I(X_I|\emptyset|X_D)$. Then, if x_E is the configuration which maximizes $P(X_E|x_O)$, $x_E^{\downarrow X_D}$ is the configuration which maximizes $P(X_D|x_O)$.

Proof. In the following we are going to denote $x_E^{\downarrow X_D}$ by x_D, and $x_E^{\downarrow X_I}$ by x_I. Our goal is to prove that if

$$x_E = x_D x_I = arg \max_{X_E} P(X_E|x_O),$$

then

$$x_D = arg \max_{X_D} P(X_D|x_O).$$

From $I(X_I|X_D|X_O)$ and $I(X_I|\emptyset|X_D)$, using contraction and weak union, we can obtain $I(X_I|X_O|X_D)$. Therefore,

$$P(x_E|x_O) = P(x_D x_I|x_O) = P(x_I|x_O)P(x_D|x_O)$$

hence $x_E = x_D x_I$ can be the configuration that maximizes $P(X_E|x_O)$ if and only if x_D and x_I are the subconfigurations that maximize $P(X_I|x_O)$ and $P(X_D|x_O)$, respectively.

□

The following proposition shows that under the conditions of proposition 1 the independence $I(X_I|\emptyset|X_O)$ is also true.

Proposition 2. Under the conditions of proposition 1 the independence relation

$I(X_I|\emptyset|X_O)$ holds.

Proof. By contraction in $\Rightarrow_1$ and decomposition in $\Rightarrow_2$ we have:

$$I(X_I|X_D|X_O)\ \&\ I(X_I|\emptyset|X_D)\ \Rightarrow_1\ I(X_I|\emptyset|X_D \cup X_O)\ \Rightarrow_2\ I(X_I|\emptyset|X_O)$$

□

Although conditions of proposition 1 can seem too hard, there are some domains in which they can be easily satisfied. For example, in diagnostic problems it is very usual to select a subset of the root nodes (diseases, components, etc ...) as the explanation set, so the independence $I(X_D|\emptyset|X_I)$ holds.

6. Experimental Evaluation

We have evaluated the algorithms using three networks:

1. The *alarm* belief network.[3] This network has been commonly used in the literature to test several kinds of algorithms (learning, propagation, ...). The alarm belief network (Figure 6) has 37 variables, each of them can take 2, 3 or 4 different states.

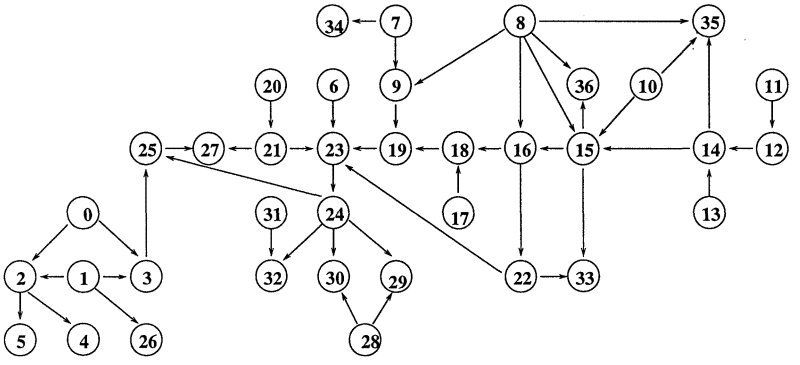


Fig. 6. The *alarm* belief network

2. An artificially generated network. This network (Figure 7) has 25 variables, each of them taking a number of states between 2 and 7.

3. The *car-starts*[10] belief network. This network (Figure 8) has 18 variables, each of them can take 2 or 3 different states.

For each network we have selected two different explanation sets, then the 20 most probable explanations were simplified using the proposed algorithms. Concretely, the six experiments are:

[10]This network is included in the package JavaBayes `www.cs.cmu.edu/~javabayes`

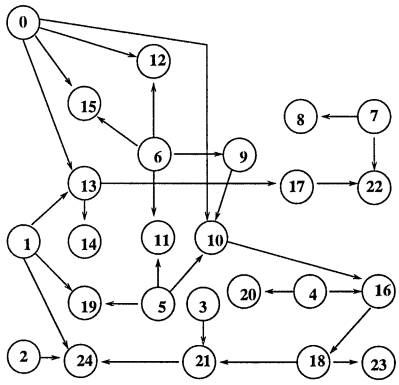


Fig. 7. The artificially generated belief network

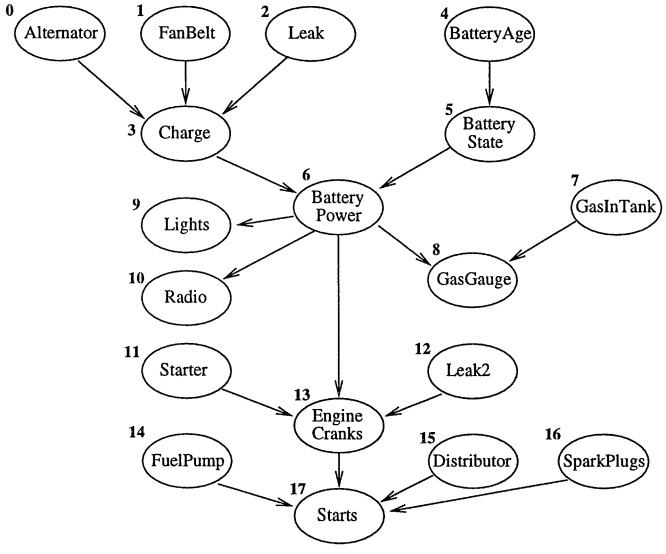


Fig. 8. The *car-starts* belief network

1. *Alarm* network. All the root nodes (variables) were selected as the explanation set, that is, $X_E = \{X_0, X_1, X_6, X_7, X_8, X_{10}, X_{11}, X_{13}, X_{17}, X_{20}, X_{28}, X_{31}\}$. Four variables were randomly selected to be observed, $X_O = \{X_{12}, X_{24}, X_{35}, X_{36}\}$.

2. *Alarm* network. Twelve variables were randomly selected as the explanation set, $X_E = \{X_5, X_6, X_8, X_{11}, X_{12}, X_{14}, X_{17}, X_{20}, X_{26}, X_{27}, X_{33}, X_{34}\}$. Four variables were randomly selected to be observed, $X_O = \{X_3, X_9, X_{19}, X_{36}\}$.

3. Artificial network. All the variables with odd index were selected as the explanation set, that is, $X_E = \{X_1, X_3, X_5, X_7, X_9, X_{11}, X_{13}, X_{15}, X_{17}, X_{19}, X_{21}, X_{23}\}$. Three variables were randomly selected to be observed, $X_O = \{X_4, X_{10}, X_{24}\}$.

4. Artificial network. All the variables with even index were selected as the explanation set, that is, $X_E = \{X_2, X_4, X_6, X_8, X_{10}, X_{12}, X_{14}, X_{16}, X_{18}, X_{20}, X_{22}, X_{24}\}$. Three variables were randomly selected to be observed, $X_O = \{X_5, X_{15}, X_{19}\}$.

5. *Car-starts* network. All the root nodes (variables) were selected as the explanation set, that is, $X_E = \{X_0, X_1, X_2, X_4, X_7, X_{11}, X_{12}, X_{14}, X_{15}, X_{16}\}$. Two variables were randomly selected to be observed $X_O = \{X_9, X_{17}\}$.

6. *Car-starts* network. Ten variables were randomly selected as the explanation set, $X_E = \{X_0, X_2, X_3, X_4, X_5, X_6, X_{10}, X_{12}, X_{13}, X_{15}\}$. Two variables were randomly selected to be observed $X_O = \{X_1, X_{11}\}$.

Tables 1 and 2 show the mean number of literals in the simplified explanations. The threshold used in all the experiments has been $\epsilon = 0.05$. The algorithms are:

- *Exhaustive*: an algorithm that explores the complete search space of sub-configurations for each explanation, and obtains the corresponding best simplification according to Definition 3 (and its adaptation to the criterion of R-simplification).

- *Cautious*: the algorithm exposed in Subsection 4.1.

- *Cautious*^{G}: the same algorithm but applied after obtaining the simplifications induced by the graph (Section 5).

- *Regressive*: the algorithm proposed in Subsection 4.2, applied after obtaining the simplifications induced by the graph.

- *Regressive**: the same algorithm but using the algorithm proposed in Subsection 4.2.1 to evaluate the sub-configurations, so the accessed sub-configurations are also evaluated.

In each experiment, we have obtained the following simplifications induced by the graph:

Table 1. Mean number of literals in each simplified explanation using I-simplification criterion.

	Alarm network		*Artificial network*		*Car-starts network*	
Experiment:	1	2	3	4	5	6
$\|X_E\|$	12	12	12	12	10	10
Exhaustive	2.3	3.45	3.45	1.5	2	2.85
Cautious	8.1	9.05	5.45	5.5	6.2	5.6
Cautious^G	7.75	7.65	4.75	2.3	5.6	3.8
Regressive	2.95	4.35	5.45	2.0	2	2.85
Regressive*	2.95	4.00	3.95	2.0	2	2.85

Table 2. Mean number of literals in each simplified explanation using R-simplification criterion.

	Alarm network		*Artificial network*		*Car-starts network*	
Experiment:	1	2	3	4	5	6
$\|X_E\|$	12	12	12	12	10	10
Exhaustive	1.6	1.9	3.2	1.5	1.5	2.6
Cautious	6.25	5.15	4.85	2.6	3.6	4.85
Cautious^G	6.1	4.45	4.25	2.3	3.3	3.35
Regressive	1.6	1.9	3.35	1.5	1.5	2.6
Regressive*	1.6	1.9	3.3	1.5	1.5	2.6

1. $I(X_O|X_E \setminus \{X_0, X_1, X_{28}, X_{31}\}|\{X_0, X_1, X_{28}, X_{31}\})$
2. $I(X_O|X_E \setminus \{X_{11}, X_{12}\}|\{X_{11}, X_{12}\})$
3. $I(X_O|X_E \setminus \{X_4, X_{16}, X_{20}\}|\{X_4, X_{16}, X_{20}\})$
4. $I(X_O|X_E \setminus \{X_7, X_{17}, X_{19}\}|\{X_7, X_{17}, X_{19}\})$
5. $I(X_O|X_E \setminus \{X_7\}|\{X_7\})$
6. $I(X_O|X_E \setminus \{X_4, X_5, X_{10}, X_{12}\}|\{X_4, X_5, X_{10}, X_{12}\})$

As can be noticed, we can have algorithms obtaining simplifications with similar mean number of literals, but with a high degree of di-similarity between the simplifications provided. For example, assume the explanation *abcde*, and the two simplifications *abc* and *ade* obtained by two different methods. Of course, they have the same number of literals, but they are quite different. For us, this fact does not constitute a problem, because if both configurations have been obtained as simplifications, then both account for the observed evidence with similar strength. Anyway, it is interesting to study how similar are the simplifications obtained by the proposed algorithms with respect to the best simplification obtained by the exhaustive algorithm. The similarity measure we use is the Hamming distance: let us represent a simplification x'_E as an array of length $|X_E|$, such that, $x'_E[i] = 1$ if

literal X_{E_i} is contained in simplification x'_E, and 0 otherwise. Then, the Hamming distance between two simplifications x'_E and x''_E is calculated as:

$$H(x'_E, x''_E) = \sum_{i=1}^{|X_E|} abs(x'_E[i] - x''_E[i]).$$

The Hamming distance computes the number of literals contained in simplication x'_E and not in x''_E plus the number of literals contained in x''_E and not in x'_E. Then, $H(x'_E, x''_E)$ can be expressed as

$$H(x'_E, x''_E) = A(x'_E, x''_E) + M(x'_E, x''_E),$$

where

$$A(x'_E, x''_E) = \sum_{\substack{i=1 \\ x'_E[i]=1}}^{|X_E|} (x'_E[i] - x''_E[i]) \text{ and } M(x'_E, x''_E) = \sum_{\substack{i=1 \\ x''_E[i]=1}}^{|X_E|} (x''_E[i] - x'_E[i]).$$

If x''_E is the simplication obtained by the exact method (exhaustive) and x'_E is the one obtained by an approximate method, the two previous quantities have a different meaning: $M(x'_E, x''_E)$ is the number of correct literals that the approximate method misses, and $A(x'_E, x''_E)$ is the number of literals that the approximate method adds to the correct ones.

Tables 3 and 4 show the mean numbers of missing (M) and added (A) literals of the different algorithms with respect to the exhaustive method. In these tables *caut.* is used as abbreviation of *cautious* and *reg.* as abbreviation of *regressive.*

Table 3. Mean number of missing (M) and added (A) literals for each simplified explanation using I-simplification criterion.

	Alarm network				*Artificial network*				*Car-starts network*			
Exp.:	1		2		3		4		5		6	
$\|X_E\|$	12		12		12		12		10		10	
	M	A	M	A	M	A	M	A	M	A	M	A
Caut.	0.2	6.0	0.1	5.7	0.38	2.38	0.55	4.55	0.0	4.2	0.3	3.05
Caut.^{G}	0.2	5.65	0.05	4.25	0.35	1.65	0.4	1.2	0.0	3.6	0.1	1.05
Reg.	0.2	0.85	0.2	1.1	0.4	2.4	0.4	0.9	0.0	0.0	0.0	0.0
Reg.*	0.2	0.85	0.25	0.8	0.45	0.95	0.4	0.9	0.0	0.0	0.0	0.0

From the results displayed in Tables 1, 2, 3, and 4 we can obtain two conclusions: First, the two proposed simplification criteria reduce considerably the size of the original explanations, as can be observed from the number of literals in the explanations obtained by the exhaustive method. Second, the approximate methods perform quite well, particularly the Regressive and Regressive* algorithms: they also obtain simplifications with a small number of literals, and more importantly,

Table 4. Mean number of missing (M) and added (A) literals for each simplified explanation using R-simplification criterion.

	Alarm network				*Artificial network*				*Car-starts network*			
Exp.:	1		2		3		4		5		6	
$\|X_E\|$	12		12		12		12		10		10	
	M	A	M	A	M	A	M	A	M	A	M	A
Caut.	0.05	4.7	0.0	3.25	0.35	2.0	0.0	1.1	0.0	2.1	0.2	2.45
Caut.^{G}	0.05	4.55	0.0	2.55	0.2	1.25	0.0	0.8	0.0	1.8	0.1	0.85
Reg.	0.0	0.0	0.0	0.0	0.15	0.3	0.0	0.0	0.0	0.0	0.0	0.0
Reg.*	0.0	0.0	0.0	0.0	0.1	0.2	0.0	0.0	0.0	0.0	0.0	0.0

these simplifications are quite similar to the exact ones, as can be seen from the number of missing and added literals (on the average, they add less than one literal and miss almost none literal with respect to the exact simplification).

In order to show an example of our simplification criteria, we can consider the following case taken from one of the experiments carried out over the network *car-starts*. The root nodes were selected as the explanation set, and *Lights=work, Starts=No* as the evidence. Then, the obtained MPE is:

$$Alternator = Ok, FanBelt = Ok, Leak = NoLeak, BatteryAge = New,$$
$$GasInTank = NoGas, Starter = Faulted, Leak2 = False, FuelPump = Ok,$$
$$Distributor = Ok \text{ and } SparkPlugs = Ok.$$

And the simplified explanation is (in this case both independence criteria {I,R} yield the same simplification):

$$BatteryAge = New \text{ and } Starter = Faulted.$$

7. Concluding Remarks

In this paper two simplification criteria and algorithms to carry out the simplification have been proposed. This kind of criteria has the advantage of having a consistent semantic based on independence concepts.

The simplification algorithms are based on a heuristic search combined with an evaluation procedure which minimizes the number of necessary computations to evaluate a configuration and its sub-configurations. Furthermore, we have seen how a study of the graph is interesting because some quick simplifications can be carried out before performing any probabilistic propagation.

The results obtained by the proposed algorithms are better than the results obtained when cautious propagation is directly applied to this task. Furthermore, we have seen how the results of cautious propagation are improved in a significant way when it is applied after obtaining the simplifications induced by the graph. Following the suggestion of one of the reviewers, for the future we plan to exploit the combination of both methods (regressive and cautious) by considering regressive search as an extension of cautious propagation, where subsequent messages are

collected in order to give access to a larger set of probabilities of subsets of the explanations.

With respect to the accuracy of the developed algorithms, we can see that the mean of the Hamming distance between the exact simplification and the obtained one, is always small. That is, the algorithms not only remove a great number of literals from the original explanation, but also the correct literals. As a consequence, the obtained simplification very often coincides with the exact one.

In relation with the simplifications induced by the graph, we think that its 'a posteriori' (that is, after searching for the K MPEs, but before starting the simplification process) use is always recommended, because it is a non expensive-time consuming process and after its application, the number of configurations to be evaluated is significantly smaller, so we obtain a high pay-off. Of course, if conditions of Proposition 1 hold, then we can use the simplification induced by the graph 'a priori', that is, reducing the explanation set before searching for the K MPEs.

In the experiments carried out the number of literals in the simplified explanations is between the 16.6% and the 33.3% of the literals in the initial explanation when the I-simplification criterion is used, and between the 12.5% and the 27.5% when the R-simplification criterion is used.

It is important a careful selection of the explanation set, as the final results can heavily depend of the variables in this set. This selection will be the object of future research. Now, what we can say with an intuitive basis is that we should include the variables relevant for our future decisions and not intermediate facts or non-observed variables. For example, in a medical application we should choose the diseases and not the intermediate consequences or the results of tests that have not been carried out.

From the experiments, we have learned that some different explanations can be simplified to the same configuration, so it is possible that we cannot give K simplified explanations if we start with K explanations. Due to this observation and given the complexity of abductive reasoning in BBNs, we plan to investigate in the development of approximate methods that directly yield simplified explanations, i.e., the process will only involve one step and not two as in the present work.

Acknowledgments

This work has been supported by the Spanish Comisión Interministerial de Ciencia y Tecnología (CICYT) under Project TIC97-1135-CO4-01. We would like to thank the anonymous reviewers for their careful comments.

References

1. S. Acid and L.M. de Campos. "Finding minimum d-separating sets in belief networks", *Proc. of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI–96)*, Portland, Oregon, 1996, pp. 3–10.

2. D.E. Appelt and M. Pollack. "Weighted abduction for plan ascription", Technical Report, Artificial Intelligence Center and Center for the Study of Language and Information, SRI International, Menlo Park, California, 1990.
3. I.A. Beinlich, H.J. Suermondt, R.M. Chavez, and G.F. Cooper. "The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks", In *Proceedings of the Second European Conference on Artificial Intelligence in Medicine*, Springer-Verlag, 1989, pp. 247–256.
4. U. Chajewska and J. Y. Halpern. "Defining explanation in probabilistic systems", *Proc. of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–97)*, San Francisco, CA, 1997, pp. 62–71.
5. E. Charniak and E. McDermott. *Introduction to Artificial Intelligence* (Addison-Wesley, 1985).
6. A.P. Dawid. "Applications of a general propagation algorithm for probabilistic expert systems", *Statistics and Computing*, **2** (1992) 25–36.
7. L.M. de Campos, J.A. Gámez, and S. Moral. "Partial Abductive Inference in Bayesian Belief Networks using a Genetic Algorithm", *Pattern Recognition Letters*, **20** (1999) 1211–1217.
8. L.M. de Campos, J.A. Gámez, and S. Moral, "On the problem of performing exact partial abductive inference in Bayesian belief networks using junction trees", In *Proceedings of the 8th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU'00)*, Madrid, 2000, pp. 1270–1277.
9. J.A. Gámez. "Inferencia abductiva en redes causales", Ph.D. Thesis, Departamento de Ciencias de la Computación e I.A. Escuela Técnica Superior de Ingeniería Informática. Universidad de Granada, 1998 (*In Spanish*).
10. D. Geiger, T. Verma, and J. Pearl. "d-Separation: From Theorems to Algorithms", In *Uncertainty in Artificial Intelligence, 5* eds. M. Henrion, R.D. Shachter, L.N. Kanal, and J.F. Lemmer. (North-Holland, Amsterdam, 1990) pp. 169–198.
11. F.V. Jensen. "Cautious propagation in Bayesian networks", *Proc. of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence (UAI–95)*, Montreal, Quebec, Canada, 1995, pp. 323–328.
12. F.V. Jensen. *An introduction to Bayesian Networks* (UCL Press, London, 1996).
13. F.V. Jensen, S.H. Aldenryd, and K.B. Jensen. "Sensitivity analysis in Bayesian networks", in *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, (Springer Verlag LNAI 946, 1995) pp. 243–250.
14. F.V. Jensen, S.L. Lauritzen, and K.G. Olesen. "Bayesian updating in causal probabilistic networks by local computation", *Computational Statistics Quarterly*, **4** (1990) 269–282.
15. H. Kautz and J. Allen. "Generalized plan recognition", *Proc. of National Conference on Artificial Intelligence*, 1986, pp. 32–37.
16. U.P. Kumar and U.B. Desai. "Image interpretation using Bayesian networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18** (1996) pp. 74–78.
17. S.L. Lauritzen, A.P. Dawid, B.N. Larsen, and H-G. Leimer. "Independence properties of directed Markov fields", *Networks*, **20** (1990) pp. 491–505.
18. Li, Z. and D'Ambrosio B. "An efficient approach for finding the MPE in belief networks", In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence*, Morgan and Kaufman, 1993, pp. 342–349.
19. R. E. Neapolitan. *Probabilistic Reasoning in Expert Systems. Theory and Algorithms* (Wiley Interscience, New York, 1990).
20. D. Nilsson. "An efficient algorithm for finding the m most probable configurations in Bayesian networks", *Statistics and Computing*, **2** (1998) pp. 159–173.

21. P. O'Rorke, S. Morris, and D. Schulenberg. "Theory formation by abduction: initial results of a case study based on the chemical revolution", Technical Report ICS-TR-89-25, University of California, Irvine, Department of Information and Computer Science, 1989.
22. J. Pearl. *Probabilistic Reasoning in Intelligent Systems* (Morgan Kaufmann, San Mateo, 1988).
23. Y. Peng and J.A. Reggia. "A probabilistic causal model for diagnostic problem solving. Part One", *IEEE Transactions on Systems, Man, and Cybernetics*, **17** (1987) pp. 146–162.
24. Y. Peng and J.A. Reggia. "A probabilistic causal model for diagnostic problem solving. Part Two", *IEEE Transactions on Systems, Man, and Cybernetics*, **17** (1987) pp. 395–406.
25. D. Poole and K. Kanazawa. "A decision-theoretic abductive basis for planning", *Proc. of AAAI Spring Symposium on Decision-Theoretic Planning*, Stanford University, March 1994, pp. 232–239.
26. H.E. Pople. "On the mechanization of abductive logic", *Proc. of the 3rd International Joint Conference on Artificial Intelligence*, 1973, pp. 147–152.
27. J.A. Reggia. "Diagnostic expert systems based on a set covering model", *International Journal of Man-Machine Studies*, **19** (1983) pp. 437–460.
28. R. Reiter. "A theory of diagnosis from first principles", *Artificial Intelligence*, **32** (1987) pp. 57–95.
29. B. Seroussi and J.L. Goldmard. "An algorithm directly finding the k most probable configurations in Bayesian networks", *International Journal of Approximate Reasoning*, **11** (1994) pp. 205–233.
30. G. Shafer. *Probabilistic Expert Systems.* (Society for Industrial and Applied Mathematics-SIAM, 1996).
31. P.P. Shenoy and G.R. Shafer. "Axioms for probability and belief-function propagation", In *Uncertainty in Artificial Intelligence, 4,* eds. R.D. Shachter, T.S. Levitt, L.N. Kanal, and J.F. Lemmer (North-Holland, Amsterdam, 1990) pp. 169–198.
32. S.E. Shimony. "The role of relevance in explanation I: Irrelevance as statistical independence", *International Journal of Approximate Reasoning*, **8** (1993) pp. 281–324.
33. S.E. Shimony. "Finding maps for belief networks is NP-hard", *Artificial Intelligence*, **68** (1994) pp. 399–410.
34. S.E. Shimony. The role of relevance in explanation II: Disjunctive assignments and approximate independence. *International Journal of Approximate Reasoning*, **13** (1995) pp. 27–60.
35. M.E. Stickel. "A prolog-like inference system for computing minimum-cost abductive explanations in natural language interpretation", Technical Report 451, AI Center, SRI International, 1988.
36. H.J. Suermondt. "Explanation of probabilistic inference in Bayesian belief networks", Technical Report KSL-91-39, Knowledge Systems Laboratory. Stanford University, Stanford, 1991.
37. P. Thagard. "Explanatory coherence", *Behavioral and Brain Sciences*, **12** (1989) pp. 435–467.
38. H. Xu. "Computing marginals for arbitrary subsets from marginal representation in Markov trees". *Artificial Intelligence*, **74** (1995) pp. 177–189.