Selective Evidence Gathering for Diagnostic Belief Networks

L. van der Gaag and M. Wessels

RUU-CS-93-31
October 1993

Utrecht University

Department of Computer Science

Padualaan 14, P.O. Box 80.089,
3508 TB Utrecht, The Netherlands,
Tel. : ... + 31 - 30 - 531454

Selective Evidence Gathering for Diagnostic Belief Networks

L. van der Gaag and M. Wessels

Technical Report RUU-CS-93-31
October 1993

Department of Computer Science
Utrecht University
P.O.Box 80.089
3508 TB Utrecht
The Netherlands

ISSN: 0924–3275

The provision of intelligent control over reasoning is generally considered one of the main contributions of artificial intelligence research to automated reasoning: knowledge-based systems thank their success largely to their ability to apply specialized knowledge for pruning search spaces and for selectively gathering evidence. As a first step towards extending the belief network framework with explicit means for exerting control, we present a method for selective gathering of evidence for diagnostic belief networks. The paper is organised as follows. In Section 2 we give an informal introduction to the belief network framework. Section 3 discusses our method for selective evidence gathering. The paper is rounded off with some conclusions in Section 4.

2 Preliminaries

As we have mentioned in our introduction, the belief network framework provides a formalism for representing knowledge concerning a joint probability distribution on a problem domain. A belief network comprises two parts: a *qualitative representation* and a *quantitative representation* of the distribution.

The qualitative part of a belief network is a graphical representation of the independencies between the statistical variables discerned in the domain; it takes the form of an acyclic directed graph. Each node in the digraph represents a variable that can take one of a set of values. In the sequel, we will restrict the discussion to binary variables taking one of the truth values *true* and *false*; the generalization to variables with more than two discrete values, however, is straightforward. We will adhere to the following notational convention: v denotes the proposition that the variable V takes the value *true*; $V = false$ will be denoted by $\neg v$. The arcs of the digraph represent dependencies between the variables. Informally speaking, we take an arc (V, W) in the digraph to represent a direct 'influential' or 'causal' relationship between the linked variables V and W; the direction of the arc designates W as the effect or consequence of the cause V. Absence of an arc between two nodes means that the corresponding variables do not influence each other directly, and that the variables are (conditionally) independent. The digraph of a belief network is generally configured by an expert from human judgment; hence the phrase *belief* network.

Associated with the graphical part of a belief network is a numerical assessment of the 'strengths' of the represented relationships: with each node of the digraph is associated a *probability assessment function* which basically is a set of (conditional) probabilities describing the influence of the values of the predecessors of the node on the probabilities of the values of the node itself. The assessment functions of a belief network provide all information necessary for uniquely defining a joint probability distribution that respects the independency relationships portrayed by the graphical part of the network.

Example 2.1 Consider the belief network shown in Figure 1, representing a small piece of fictitious medical 'knowledge' concerning the diagnosis of acute cardiac disorders. The information

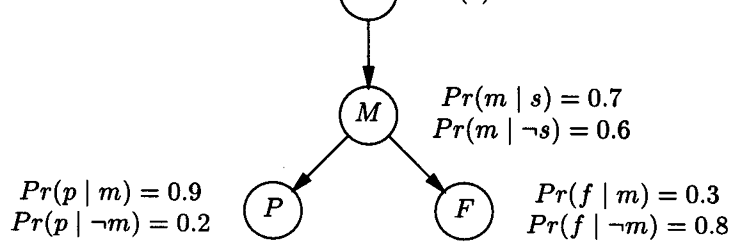

Figure 1: A fictitious diagnostic belief network.

represented in the network pertains to patients presenting at a first aid clinic. We like to stress that the belief network just serves illustrative purposes and should not be taken too seriously. The belief network comprises only four variables: the variable S represents the *smoking history* of a patient, the variable M represents the presence or absence of a *myocard infarct*, more commonly known as a heart infarct, the variable P represents whether or not a patient is suffering from *pain on the chest*, and the variable F represents whether or not a patient complains of *tingling fingers*. In the digraph, the smoking history of a patient is denoted as having a direct influence on the presence or absence of a myocard infarct in this patient. On having a myocard infarct, a patient is likely to complain of pain on the chest: this is expressed through the probability of chest pain given myocard infarct being equal to 0.9. On the other hand, given myocard infarct a patient is very unlikely to have tingling fingers which is expressed through the low probability $Pr(f \mid m) = 0.3$; the presence of tingling fingers in fact suggests that a patient is suffering from another disorder such as hyperventilation, not modelled in our incomplete network. □

For making probabilistic statements concerning the variables discerned in the problem domain, two algorithms are associated with the belief network formalism:

- an algorithm for (efficiently) computing probabilities of interest from a network, and
- an algorithm for processing evidence, that is, for entering evidence into the network and subsequently (efficiently) computing the revised probability distribution given this evidence.

Since a joint probability distribution on the variables is uniquely defined by the topology of the digraph of a belief network and its associated assessment functions, any probability of interest can be computed from these functions. Equally, the impact of a value of a specific variable becoming known, on each of the other variables in the network can be computed from the initial assessment functions. Now, observe that the assessment functions describe the joint probability distribution locally for each node and its predecessors. Calculation of a (revised) probability from the joint probability distribution defined by the belief network in a straightforward manner, however, is computationally expensive and will generally not be restricted to performing computations that are local in terms of the graphical part of the network. In the literature therefore, several less naive algorithms for computing probabilities of interest from a belief network and for processing evidence in the network have been proposed, see for example [Pearl, 1988] and [Lauritzen & Spiegelhalter, 1988]. Although all schemes proposed for evidence propagation are based on probability theory, they differ considerably with respect to the algorithms employed and their complexity; it should be noted that in general probabilistic inference in belief networks without any restrictions is NP-hard, [Cooper, 1990].

3 Selective Gathering of Evidence

The goal of diagnostic problem solving is to confirm a hypothesis concerning the cause of an observed malfunction to sufficient extent. This is generally achieved by successively gathering and processing evidence for variables representing manifestations. We have mentioned before that it often is not necessary to obtain all possible evidence before a sufficiently accurate diagnosis is reached: it generally suffices to carefully select a few variables for which to acquire evidence. Evidence gathering may be stopped as soon as a hypothesis is confirmed or disconfirmed to sufficient extent.

In this section, we develop a method for performing the task of selective evidence gathering in the context of diagnostic belief networks. In doing so, we make two simplifying assumptions. First, we restrict the number of variables representing hypotheses to one. This assumption excludes the possibility of interacting causes of an observed malfunction. Note that the belief network framework in essence allows for representing and dealing with multiple hypotheses. Dealing with multiple, interacting hypotheses in view of selective evidence gathering, however, is not straightforward. Secondly, we take a *myopic* approach to evidence gathering, [Gorry & Barnett, 1968], that is, variables to obtain evidence for are selected one by one. It is conceivable that in practical applications a non-myopic approach in which variables are selected groupwise might outperform any method for selective evidence gathering based on a myopic approach. Adopting a non-myopic approach, however, poses some serious computational problems. Further research is aimed at gaining insight in solving these problems, [Heckerman *et al.*, 1993].

Now envision performing the task of selective evidence gathering in the context of a diagnostic belief network. It may be outlined as follows:

1. Select the variable that is expected to contribute most to the confirmation or disconfirmation of the hypothesis;
2. Request the value of the selected variable from the user and process the entered value in the belief network;
3. Decide whether the hypothesis is confirmed or disconfirmed to sufficient extent;
4. If more evidence is necessary, then re-iterate the process from step 1; otherwise stop.

From this informal outline, it is seen that the task of selective gathering of evidence involves several issues that are not supported by the belief network framework.

From a knowledge-representation point of view, we observe that in the task of evidence gathering different types of variable are discerned. In a belief network, therefore, we have to distinguish between the following types of node, [Henrion, 1989]:

- the *hypothesis node* is a node that represents the hypothesis that has to be confirmed or disconfirmed;
- an *evidence node* is a node that represents a statistical variable whose value can be obtained by observation – an evidence node pertains (directly, or indirectly) to the confirmation or disconfirmation of the hypothesis;
- an *intermediate node* is a node that represents a statistical variable not classified in either of the former two groups.

Furthermore, additional knowledge is involved in computing the expected contribution of acquiring information on a specific variable to the confirmation or disconfirmation of the hypothesis. This computation for example requires weighing the costs of obtaining information versus the benefit resulting from a more accurate diagnosis. Note that the belief network formalism does not allow for incorporating such knowledge.

The belief network formalism can be extended to provide for the representation of additional knowledge as indicated above. In fact, the *influence diagram formalism* may be viewed as such an extension, [Howard & Matheson, 1984]. To our opinion, however, extensions to the belief network formalism itself tend to obscure the representation of a joint probability distribution and to decrease the framework's flexibility to perform different types of probabilistic computation. Therefore, we have decided not to enhance the basic formalism itself but to embed the belief network framework in a *two-layered* computational architecture instead. The first layer of this architecture specifies a belief network and its associated algorithms for computing probabilities and processing evidence. In the sequel, we will call this layer the *probabilistic layer* of the architecture. The second layer incorporates the method for selective gathering of evidence as outlined above and the additional knowledge required for this method; this layer is called the *control layer*. The two layers of the architecture cooperate in the following fashion: the control layer queries the probabilistic layer for information about the variables of interest and decides on what computations should be performed next by the probabilistic layer. The general idea is illustrated in Figure 2. The idea of designing a multi-layered architecture is not a new one – in fact, the idea pervades many areas of present artificial intelligence research, [Levitt *et al.*, 1990, Russell & Wefald, 1991].

In the following subsections, we further detail our method for selective gathering of evidence. In Section 3.1 we introduce a decision structure used by the method. Section 3.2 then presents full details on the method itself. We conclude with some observations concerning the computational complexity of the method presented.

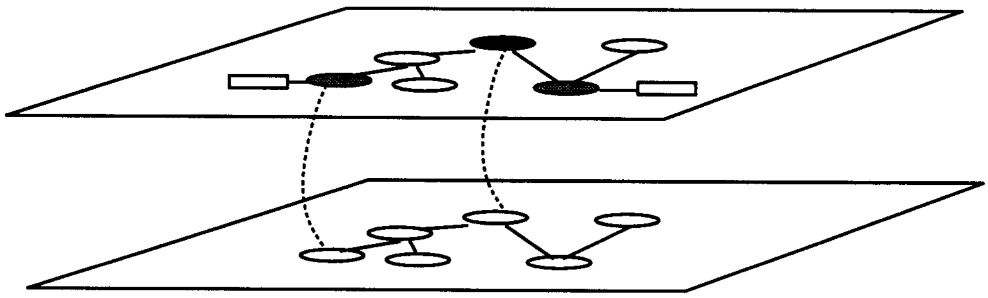

Figure 2: A two-layered belief network architecture.

3.1 The Decision Structure

As mentioned before, the task of selective gathering of evidence involves several decision problems. Classical decision theory provides a mathematical framework for solving these problems, [Smith, 1989]. Although decision theory has a longstanding history and has proven its worth for test planning, it is only recently that artificial intelligence research has begun to aim at exploiting this theory on a larger scale in knowledge-based systems, [Horvitz *et al.*, 1988].

The first decision problem in the task of selective evidence gathering for a diagnostic belief network is to select the evidence node that is expected to contribute most to the confirmation or disconfirmation of the hypothesis. Consider a belief network composed of a digraph G and a set of assessment functions, defining the joint probability distribution Pr. We take the digraph G to consist of the hypothesis node H; the set of evidence nodes is denoted as $E(G) = \{E_1, \ldots, E_m\}$, $m \geq 1$; $I(G)$ denotes the set of intermediate nodes. The problem of selecting the evidence node that is expected to contribute most to the hypothesis is modelled using a *decision tree* of the form depicted in Figure 3. The decision tree comprises one *decision node* whose outgoing edges

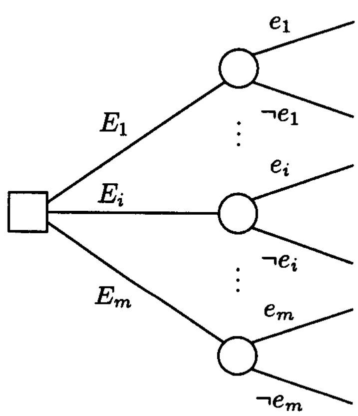

Figure 3: The decision structure employed.

correspond to the different decision options available, that is, to the decisions to obtain evidence for node E_i, $i = 1, \ldots, m$. In addition, the decision tree has m *chance nodes* representing the separate evidence nodes; the edges emanating from a chance node represent the values the corresponding evidence node can adopt. With each leaf of the decision tree is associated a utility indicating the usefulness of 'knowing' the corresponding evidence value in view of the confirmation or disconfirmation of the hypothesis. To this end, decision theory provides several types of *utility function*, [Ben-Bassat, 1978, Glasziou & Hilden, 1989, Heckerman *et al.*, 1992]. These functions may be based on probabilistic information only and not contain any other information about the domain at hand; an example of such a function is the well-known entropy measure of information content. Yet, a utility function may also involve non-probabilistic aspects from the domain, such as the costs of obtaining evidence. In practice, the utility function best chosen will generally be one that involves probabilistic as well as other domain-dependent information. For ease of exposition, however, we will use in this paper a simple (linear) utility function based on probabilistic information only. In the method presented, this utility function can easily be replaced by, for example, the entropy measure.

Suppose that after obtaining and processing some evidence the probability of the truth of the hypothesis node H equals $Pr(h \mid c)$, where c denotes the conjunction of all evidence obtained sofar. Now, for an uninstantiated evidence node E_i the difference between $Pr(h \mid c)$ and $Pr(h \mid c \wedge e_i)$ indicates the confidence gained in h if the evidence $E_i = true$ is observed; a similar observation holds for the evidence $E_i = false$. This motivates our utility function u being defined by

$$u(E_i) = |Pr(h \mid c) - Pr(h \mid c \wedge E_i)|$$

for all $E_i \in E(G)$, where E_i takes a value from $\{e_i, \neg e_i\}$. We note that this utility function satisfies the axioms of utility theory and reflects the value of imperfect information.

The decision tree thus constructed is evaluated using *foldback analysis*, [Smith, 1989]: for a chance node its expected utility is computed from the utilities associated with the leaves of the corresponding subtree and the probabilities of the values of the node. For the decision node the maximum expected utility is computed from the expected utilities of the chance nodes. The evidence node yielding the maximum expected utility is the one likely to contribute most to the confirmation or disconfirmation of the hypothesis.

3.2 Details of the Method

We will now present full details on our method for selective gathering of evidence building on the aspects discussed above. The method is composed of two basic algorithms. The first algorithm selects the evidence node that is expected to provide the largest support for or against the hypothesis. The second algorithm decides when evidence gathering may be stopped. We will discuss these basic algorithms separately before combining them into the main algorithm. The computational complexity of the resulting algorithm is discussed in Section 3.3.

In the description of the first basic algorithm we use $E \subseteq E(G)$ to denote the set of uninstantiated evidence nodes and c to denote the conjunction of all evidence obtained sofar.

procedure select-node(E,E_j)
  build a decision tree D from E as in Figure 3;
  for each evidence node E_i in E **do**
    compute the utilities $|Pr(h \mid c) - Pr(h \mid c \wedge e_i)|$ and $|Pr(h \mid c) - Pr(h \mid c \wedge \neg e_i)|$,
    and associate these with the corresponding leaves of D
  od;
  for each chance node in D **do**
    compute its *expected utility*
  od;
  compute the *maximum expected utility* for the decision node
  and let E_j be the evidence node that is responsible for it
end

Note that all probabilities required by this algorithm can be computed from the belief network in the probabilistic layer of our architecture.

After application of the algorithm the user is prompted for a value for the evidence node E_j selected by the algorithm. The entered value is subsequently processed in the belief network and the algorithm is called for the set of remaining evidence nodes $E \setminus \{E_j\}$. Repeated application of the algorithm thus yields a sequence of evidence nodes for which evidence has to be entered. The following example illustrates the application of the algorithm.

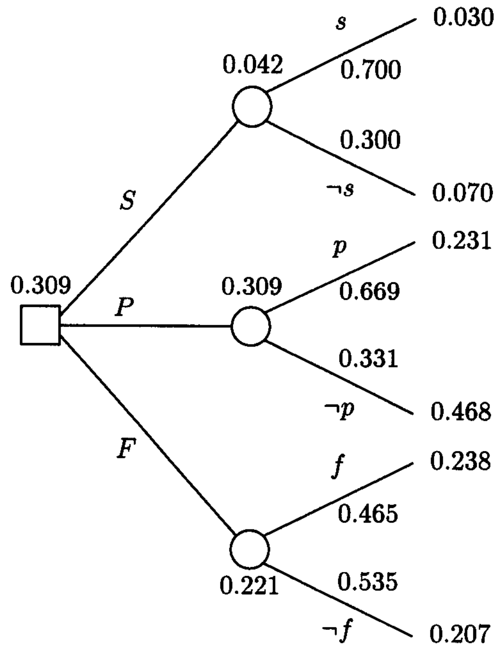

Figure 4: Decision tree, first step.

Example 3.1 Consider once more the belief network shown in Figure 1, representing some medical knowledge concerning the diagnosis of acute cardiac disorders. The hypothesis node is node M, the evidence nodes are $E(G) = \{S, P, F\}$ and there are no intermediate nodes, that is, $I(G) = \varnothing$. The select-node algorithm first builds a decision tree D from $E(G)$. The prior probability of the presence of a myocard infarct is computed from the belief network and is found to be $Pr(m) = 0.67$; if this probability seems to be extremely high, then recall that the information of the network is conditional on a patient's presenting to a first aid clinic. In addition, the utilities for the leaves of the tree are computed. For example the utility yielded by the evidence that a patient suffers from pain on the chest equals

$$\begin{aligned} u(p) &= |Pr(m) - Pr(m \mid p)| = \\ &= |0.67 - 0.901| = \\ &= 0.231 \end{aligned}$$

Figure 4 depicts the resulting decision tree and its associated utilities. Now, for each chance node the expected utility is computed. For example, for the chance node P we find the expected utility

$$\begin{aligned} \hat{u}(P) &= Pr(p) \cdot u(p) + Pr(\neg p) \cdot u(\neg p) = \\ &= 0.669 \cdot 0.231 + 0.331 \cdot 0.468 = \\ &= 0.309 \end{aligned}$$

Next, the maximum expected utility is computed for the decision node, yielding the value 0.309. Since node P is responsible for this value, the user is requested to enter evidence concerning the presence or absence of chest pain in the patient. It depends on the specific evidence entered what the user will have to do next. Suppose that the evidence $P = \mathit{true}$ is entered, that is, the user confirms the patient's suffering from chest pain. Then, a new decision tree is built and the corresponding utilities are computed as shown in Figure 5. This results in the user being requested to enter evidence for node F. □

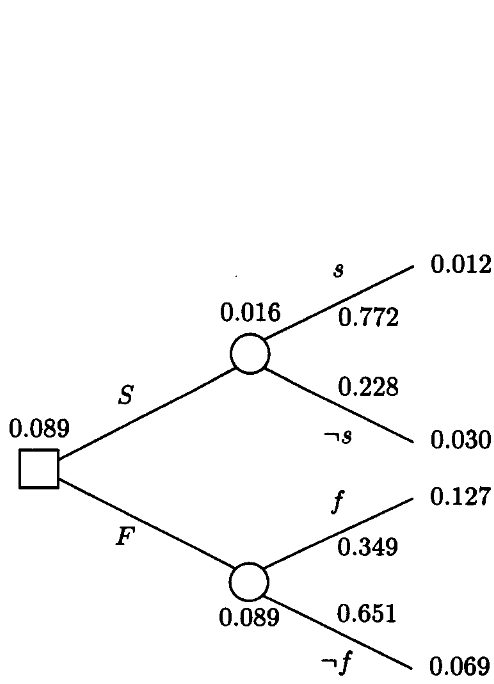

Figure 5: Decision tree, second step.

Note that the sequence of nodes yielded by repeated application of the algorithm is not guaranteed to be the best because of the uncertainties involved. It may happen that the actual evidence obtained for a selected node is not so informative as hoped for. This situation arises when the expected utility is largely determined by an outcome that has a high probability and a high gain of confidence. However, if the outcome turns out to be the one with the low probability and the low gain of confidence, it will fall short of expectations.

We now turn to the algorithm for deciding when to stop evidence gathering. Recall that the purpose of reasoning with a belief network in diagnostic problem solving is to confirm or disconfirm a given hypothesis. We say that a hypothesis is *confirmed* if the probability of the truth of the hypothesis given evidence for all evidence nodes has exceeded a certain threshold value t, $0 \leq t \leq 1$. Likewise, a hypothesis is *disconfirmed* if the probability of the truth of the hypothesis given evidence for all evidence nodes has dropped below a threshold value, for example, $1 - t$.

Now suppose that after obtaining several pieces of evidence the probability of the truth of the hypothesis has surpassed the threshold value t. Theoretically, we have to obtain evidence for all remaining evidence nodes before we can confirm the hypothesis. However, if we know that the probability of the truth of the hypothesis can never drop below the threshold value whatever evidence may be obtained for these nodes, it is not necessary to actually obtain the evidence before confirming the hypothesis. Therefore, before we confirm the hypothesis we compute the probabilities of the truth of the hypothesis for all possible combinations of evidence for the remaining evidence nodes. We illustrate this with an example.

Example 3.2 Consider once more the belief network shown in Figure 1. Suppose that the threshold value for confirming the hypothesis m equals 0.8. Furthermore, suppose that we have observed and processed the evidence $P = true$. The probability of m given this evidence, $Pr(m \mid p)$, equals 0.901 and therefore has exceeded the threshold value. For deciding whether or not the evidence entered sofar is sufficient for confirming the hypothesis the following probabilities are calculated from the belief network in the probabilistic layer:

$$
\begin{aligned}
Pr(m \mid p \wedge s \wedge f) &= 0.797 \\
Pr(m \mid p \wedge \neg s \wedge f) &= 0.717 \\
Pr(m \mid p \wedge s \wedge \neg f) &= 0.974 \\
Pr(m \mid p \wedge \neg s \wedge \neg f) &= 0.959
\end{aligned}
$$

Since for some combinations of evidence for the remaining evidence nodes S and F, the probability of the truth of the hypothesis drops below the threshold value, it is necessary to obtain some additional evidence before the hypothesis can be confirmed. Recall from Example 3.1 that the