

- [Pearl, 1988] J. Pearl (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publishers.
- [Russell & Wefald, 1991] S. Russell, E. Wefald (1991). Principles of metareasoning, *Artificial Intelligence*, vol. 49, pp. 361 – 395.
- [Shwe *et al.*, 1991] M.A. Shwe, B. Middleton, D.E. Heckerman, M. Henrion, E.J. Horvitz, H.P. Lehmann, G.F. Cooper (1991). Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. The probabilistic model and inference algorithms, *Methods of Information in Medicine*, vol. 30, pp. 241 – 255.
- [Smith, 1989] J.Q. Smith (1989). *Decision Analysis: a Bayesian Approach*, Chapman and Hall.
- [van der Gaag & Wessels, 1993] L.C. van der Gaag, M.L. Wessels (1993). *Test Planning for Diagnostic Belief Networks*, in preparation.

- [Bellazzi *et al.*, 1991] R. Bellazzi, C. Berzuini, S. Quaglini, D. Spiegelhalter, M. Leaning (1991). Cytotoxic chemotherapy monitoring using stochastic simulation on graphical models, in: M. Stefanelli, A. Hasman, M. Fieschi (eds.), *Proceedings of the Third Conference on Artificial Intelligence in Medicine*, Lecture Notes in Medical Informatics 44, Springer-Verlag, Berlin, pp. 227 – 238.
- [Ben-Bassat, 1978] M. Ben-Bassat (1978). Myopic policies in sequential classification, *IEEE Transactions on Computers*, vol. C-27, pp.170 – 174.
- [Bruza & van der Gaag, 1992] P.D. Bruza, L.C. van der Gaag (1992). *Index Expression Belief Networks for Information Disclosure*, Technical Report RUU-CS-92-21, Utrecht University.
- [Cooper, 1990] G.F. Cooper (1990). The computational complexity of probabilistic inference using Bayesian belief networks, *Artificial Intelligence*, vol. 42, pp. 393 – 405.
- [Glasziou & Hilden, 1989] P. Glasziou, J. Hilden (1989). Test selection measures, *Medical Decision Making*, vol. 9, pp. 133 – 141.
- [Gorry & Barnett, 1968] G.A. Gorry, G.O. Barnett (1968). Experience with a model of sequential diagnosis, *Computers and Biomedical Research*, vol. 1, pp. 490 – 507.
- [Heckerman *et al.*, 1992] D.E. Heckerman, E.J. Horvitz, B.N. Nathwani (1992). Toward normative expert systems; part 1: The pathfinder project, *Methods of Information in Medicine*, vol. 31, pp. 90 – 105.
- [Heckerman *et al.*, 1993] D. Heckerman, E. Horvitz, and B. Middleton. An approximate nonmyopic computation for value of information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 292-298, 1993.
- [Henrion, 1989] M. Henrion (1989). Some practical issues in constructing belief networks, in: L.N. Kanal, T.S. Levitt, J.F. Lemmer (eds.), *Uncertainty in Artificial Intelligence 3*, North-Holland Publishers, pp. 161 – 173.
- [Horvitz *et al.*, 1988] E.J. Horvitz, J.S. Breese and M. Henrion (1988). Decision theory in expert systems and artificial intelligence, *International Journal of Approximate Reasoning*, vol. 2, pp. 247 – 302.
- [Howard & Matheson, 1984] R.A. Howard, J.E. Matheson (1984). Influence diagrams, in: R.A. Howard, J.E. Matheson (eds.), *Readings on the Principles and Applications of Decision Analysis*, vol. 2, pp. 720 – 763. Strategic Decisions Group.
- [Jensen *et al.*, 1990] F.V. Jensen, J. Nielsen, H.I. Christensen (1990). *Use of Causal Probabilistic Networks as High Level Models in Computer Vision*, Technical Report R-90-39, University of Aalborg.
- [Lauritzen & Spiegelhalter, 1988] S.L. Lauritzen, D.J. Spiegelhalter (1988). Local computations with probabilities on graphical structures and their applications to expert systems, *Journal of the Royal Statistical Society, Series B*, vol. 50, pp. 157 – 224.
- [Levitt *et al.*, 1990] T.S. Levitt, T.O. Binford, G.J. Ettinger (1990). Utility-based control for computer vision, in: R.D. Shachter, T.S. Levitt, L.N. Kanal, J.F. Lemmer (eds.), *Uncertainty in Artificial Intelligence 4*, North-Holland Publishers, pp. 407 – 422.

first consider one pass of this loop. Concerning the `select-node` algorithm, we observe that the complexity of this algorithm is determined by the computation of the probabilities of the values of the evidence nodes and of the utilities; these steps require $\mathcal{O}(m \cdot n^2)$ computations. The computational complexity of `select-node` therefore is $\mathcal{O}(m \cdot n^2)$. The performance of the `stop-gathering` algorithm is exponential in the number of uninstantiated evidence nodes, that is, $\mathcal{O}(2^m \cdot n^2)$. One pass of the loop thus requires $\mathcal{O}(m \cdot n^2 + n^2 \cdot 2^m)$ computations. The loop is executed m times at most. Hence, the worst-case computational complexity of the algorithm is $\mathcal{O}(m^2 \cdot n^2 + m \cdot n^2 \cdot 2^m)$.

The `stop-gathering` algorithm presented in this paper is rather straightforward; its purpose here is to be illustrative of the basic idea. It will be evident that the algorithm can be heavily optimized by exploiting the independencies represented by the digraph of the belief network. Although the number of computations reduces considerably with these optimizations, the worst-case complexity of the algorithm remains exponential in the number of evidence nodes. This problem cannot be remedied if it has to be established to certainty that the probability of the hypothesis will never drop below the fixed threshold value. In a forthcoming paper we investigate the incorporation of heuristics into the `stop-gathering` algorithm, [van der Gaag & Wessels, 1993]; such heuristics allow the resulting algorithm to behave polynomially in most practical applications.

4 Conclusion

As more and more applications of the belief network framework are being developed, it becomes apparent that the framework lacks with regard to explicit means for exerting control over reasoning. In this paper, we have presented a method for selective gathering of evidence for diagnostic belief networks. In this method we have combined the belief network framework with issues from the field of decision theory and embedded the belief network framework in a two-layered computational architecture as is often encountered in artificial intelligence research.

In our method for selective gathering of evidence we have departed from basic concepts from classical decision theory. In addition, we have made several simplifying assumptions. It will require further research and experimentation to alleviate these assumptions. The method presented in this paper demonstrates that a method for selective gathering of evidence for diagnostic belief networks is feasible and therefore may serve as a starting point for such research.

To conclude, we observe that the control layer of our architecture not only provides a method for selective evidence gathering but also allows for the incorporation of a wide variety of controlling mechanisms, both independent as well as dependent on the domain for which an application is being developed.

References

- [Andreassen *et al.*, 1987] S. Andreassen, M. Woldbye, B. Falck, S.K. Andersen (1987). MUNIN - A causal probabilistic network for interpretation of electromyographic findings, *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, pp. 366 – 372.
- [Andreassen *et al.*, 1991] S. Andreassen, R. Hovorka, J. Benn, K.G. Olesen, E.R. Carson (1991). A model-based approach to insulin adjustment, in: M. Stefanelli, A. Hasman, M. Fieschi (eds.), *Proceedings of the Third Conference on Artificial Intelligence in Medicine*, Lecture Notes in Medical Informatics 44, Springer-Verlag, Berlin, pp. 239 – 248.

$Pr(m | p \wedge \neg f)$, increases to 0.970, and therefore still exceeds the indicated threshold. Once more we examine whether enough evidence has been obtained. To this end, we consider the probabilities

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

Since both probabilities now exceed the threshold value 0.8, there is no need to obtain evidence for the remaining evidence node pertaining to the patient's smoking history. \square

The following algorithm will be evident.

```

procedure stop-gathering( $E$ ,enough)
  quit := false;
  if  $Pr(h | c) \geq t$  then
    for all conjunctions  $c_E$  of values of evidence variables in  $E$  while
      not quit do
        if  $Pr(h | c \wedge c_E) < t$  then
          quit := true
        fi
      od;
    enough := not quit
  fi
end

```

The variable `enough` is initialized to `false` before this procedure is called. The variable `quit` is added for efficiency purposes: as soon as the probability of the truth of the hypothesis for a specific configuration of evidence nodes drops below the threshold value, there is no need to consider the remaining configurations.

The main algorithm integrates the two basic algorithms discussed above.

```

procedure selective-evidence-gathering
   $E := E(G)$ ;
  enough := false;
  while  $E \neq \emptyset$  and not enough do
    select-node( $E, E_j$ );
    process-evidence( $E_j$ );
     $E := E \setminus \{E_j\}$ ;
    stop-gathering( $E$ ,enough)
  od
end

```

The procedure `process-evidence` called from `selective-evidence-gathering` prompts the user for evidence for the selected evidence node and enforces processing the entered value in the belief network in the probabilistic layer.

3.3 Complexity

To conclude the presentation of our method for selective gathering of evidence for diagnostic belief networks, we discuss some issues concerning its computational complexity. In doing so, we assume that in the digraph G of the belief network the number of predecessors of a node is bound by a constant k , $0 \leq k \ll n$, where n is the number of nodes in G , rendering the complexity of computing probabilities using Pearl's algorithms of order $\mathcal{O}(n^2)$.

The worst-case computational complexity of the `selective-evidence-gathering` algorithm is determined by the complexity of the main loop of the algorithm. We

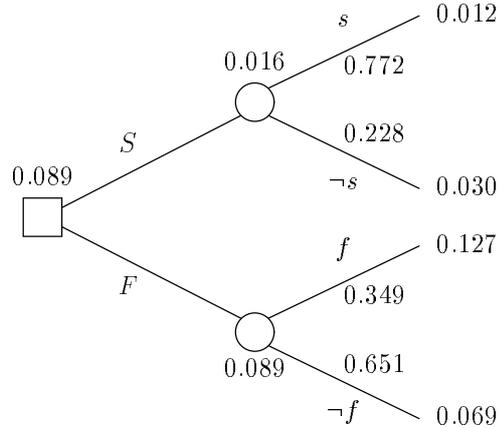


Figure 5: Decision tree, second step.

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, $Pr(m | p)$, equals 0.901 and therefore has exceeded the threshold value. For deciding whether or not the evidence entered so far is sufficient for confirming the hypothesis the following probabilities are calculated from the belief network in the probabilistic layer:

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

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 next evidence node for which evidence has to be entered, is F . Now assume that the user enters the information that the patient does not have the inconvenience of tingling fingers, that is, the evidence $F = false$ is given. The probability of the patient suffering from a myocard infarct,

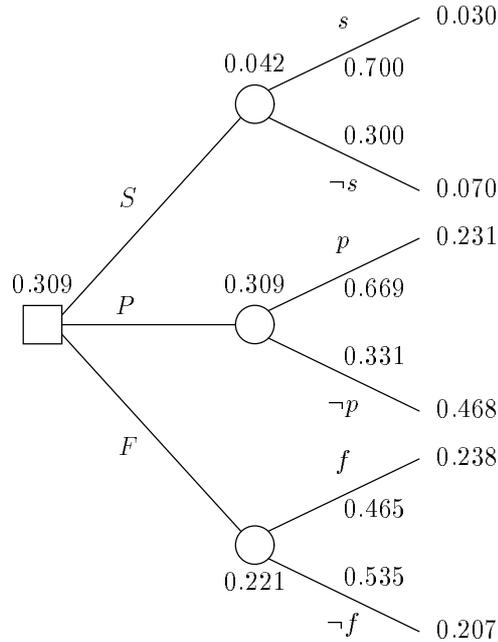


Figure 4: Decision tree, first step.

$$\begin{aligned}
 &= |0.67 - 0.901| = \\
 &= 0.231
 \end{aligned}$$

Figure 4 depicts the resulting decision tree and its associated utilities. Now, for each chance node the expected decision 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 $P = true$ 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 . \square

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

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  $|Pr(h | c) - Pr(h | c \wedge e_i)|$  and  $|Pr(h | c) - Pr(h | c \wedge \neg e_i)|$ ,
    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.

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, $I(G) = \emptyset$. The **select-node** algorithm first builds a decision tree D from $E(G)$. The prior probability of the presence of a myocardial 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

$$u(p) = |Pr(m) - Pr(m | p)| =$$

work 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 $E(G) = \{E_1, \dots, E_m\}$, $m \geq 1$; $I(G)$ 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

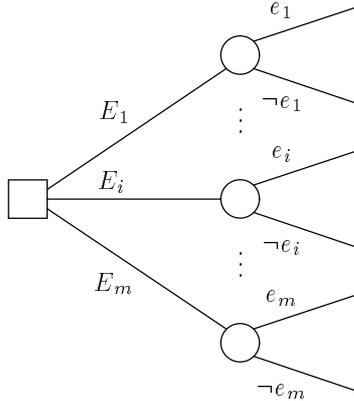


Figure 3: The decision structure employed.

outgoing edges correspond to the different decision options available, that is, to the decisions to obtain evidence for node E_i , $i = 1, \dots, m$. 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 $Pr(h | c)$, where c denotes the conjunction of all evidence obtained sofar. Now, for an uninstantiated evidence node E_i the difference between $Pr(h | c)$ and $Pr(h | c \wedge e_i)$ 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

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

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.

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].

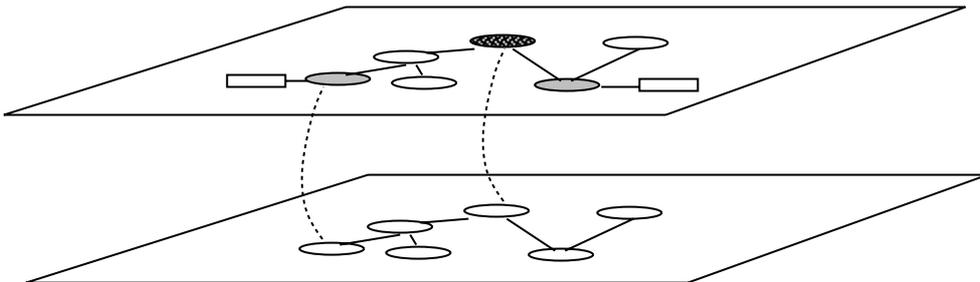


Figure 2: A two-layered belief network architecture.

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.

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 net-

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.

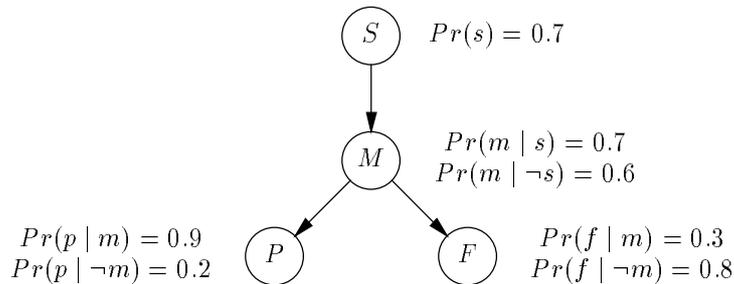


Figure 1: A fictitious diagnostic belief network.

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 $Pr(f | m) = 0.3$; 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. \square

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].

a treatment plan to ameliorate or cure the observed malfunction. Now consider implementing these tasks using the belief network framework. The framework comprises algorithms for computing probabilities of interest and for processing pieces of evidence, thus providing for performing the reasoning task. However, the second and third task mentioned above cannot be dealt with in the context of the belief network framework alone.

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*; $V = \textit{false}$ 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 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

Selective Evidence Gathering for Diagnostic Belief Networks

Linda van der Gaag & Maria Wessels

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

Abstract

The *belief network framework* for reasoning with uncertainty in knowledge-based systems has been around for some time now. As more and more practical applications employing the framework are being developed, it becomes apparent that the framework lacks with regard to explicit means for exerting control over reasoning. In this paper, we extend the belief network framework with a method for selective gathering of evidence for diagnostic applications. To this end, a belief network architecture is developed consisting of two layers: a *probabilistic layer* specifying a belief network and its associated algorithms, and a *control layer* providing the method for evidence gathering.

1 Introduction

Halfway through the 1980s, the theory of *belief networks* was introduced for reasoning with uncertainty in knowledge-based systems. The belief network framework provides a formalism for representing knowledge concerning a joint probability distribution on a set of variables discerned in a domain, and in addition provides a set of algorithms for reasoning with knowledge represented in this formalism. Since its introduction the belief network framework is becoming increasingly popular, and at present more and more knowledge-based systems are being developed using this framework, most notably in the area of medical diagnosis and therapy selection, [Andreassen *et al.*, 1987], [Andreassen *et al.*, 1991], [Bellazzi *et al.*, 1991], [Shwe *et al.*, 1991]; for applications in other domains, we refer to [Bruza & van der Gaag, 1992] and [Jensen *et al.*, 1990].

As experience with developing diagnostic belief networks progresses, it becomes apparent that the belief network framework lacks with regard to means for exerting control over reasoning. To illustrate this observation, we consider a diagnostic problem-solving context. A diagnostic knowledge-based system is typically concerned with three tasks, [Horvitz *et al.*, 1988]. The first task is the actual diagnostic reasoning: the system infers the most probable cause(s) of an observed malfunction given information about the manifestations of this malfunction. In order to reach an accurate diagnosis, however, it often is not necessary to inform the system of all manifestations: generally, only a few pieces of evidence suffice to confirm a specific diagnosis to sufficient extent. Therefore, instead of collecting all manifestations concerning an observed malfunction exhaustively, the system takes decisions during the reasoning process as to which additional information can best be requested from the user; furthermore, it decides on when to stop gathering information. Taking such decisions constitutes the second task of a diagnostic knowledge-based system. After the most probable causes of the malfunction have been determined, in other words, after a diagnosis has been reached, the third task of the system is to devise