# FUSION PROPAGATION AND STRUCTURING IN BAYESIAN NETWORKS

Judea Pearl

April 1985
Report No. CSD-850016

R-42
April 1985

# FUSION, PROPAGATION, AND STRUCTURING IN BAYESIAN NETWORKS*

Judea Pearl
Computer Science Department
University of California
Los Angeles, CA 90024
(judea@UCLA-locus)

Presented at the Symposium on
Complexity of Approximately Solved Problems
Columbia University
April 17-19, 1985

*This work was supported in part by the National Science Foundation, Grant #DSR 83-13875

# FUSION, PROPAGATION, AND STRUCTURING IN BAYESIAN NETWORKS

## ABSTRACT

Bayesian networks are directed acyclic graphs in which the nodes represent propositions (or variables), the arcs signify the existence of direct causal dependencies between the linked propositions, and the strengths of these dependencies are quantified by conditional probabilities. A network of this sort can be used to represent the causal knowledge of a domain expert and turns into a computational architecture if the links are used not merely for storing factual knowledge but also for directing and activating the data flow in the computations which manipulate this knowledge.

The first part of the paper deals with the task of fusing and propagating the impacts of new evidence and beliefs through Bayesian networks in such a way that, when equilibrium is reached, each proposition will be assigned a certainty measure consistent with the axioms of probability theory. It is shown that if the network is singly connected (e.g. tree-structured), then the propagation of updated probabilities can be accomplished by an isomorphic network of parallel and autonomous processors, and that the impact of new information can be imparted to all propositions in time proportional to the longest path in the network.

The second part of the paper deals with the problem of finding a tree-structured representation to an ensemble of probabilistically coupled propositions using auxiliary (dummy) variables, traditionally known as "hidden causes". It is shown that if such a tree-structured representation exists then it is possible to uncover the topology of the tree uniquely by observing pairwise dependencies among the available propositions (i.e. the leaves of the tree). Moreover, the entire tree structure, including the strengths of all internal relationships can be reconstructed in time proportional to $n \log n$, where $n$ is the number of leaves.

# FUSION, PROPAGATION, AND STRUCTURING IN BAYESIAN NETWORKS

Judea Pearl

## 1. INTRODUCTION

This study was motivated by attempts to devise a computational model for humans' inferential reasoning, namely, the mechanism by which people integrate data from multiple sources and generate a coherent interpretation of that data. Since the knowledge from which inferences are drawn is mostly judgmental--namely, subjective, uncertain, and incomplete--a natural place to start would be to cast the reasoning process in the framework of probability theory. However, the mathematician who approaches this task from the vantage of probability theory may dismiss it as a rather prosaic exercise. For if one assumes that human knowledge is represented by a joint probability distribution $P(x_1, \ldots, x_n)$ on a set of propositional variables $x_1, \ldots, x_n$, the task of drawing inferences from observations amounts to simply computing the marginal probabilities of a small subset, $H_1, \ldots, H_k$, of variables called hypotheses, conditioned upon a group of instantiated variables $e_1 \cdots e_m$ called evidence. Indeed computing $P(H_1, \ldots, H_k | e_1, \ldots, e_m)$ from a given joint distribution on all propositions is merely an arithmetic tediousness void of theoretical or conceptual interest.

It is not hard to see that this textbook view of probability theory presents a rather distorted picture of human reasoning and misses its most interesting aspects. Consider, for example, the problem of encoding an arbitrary joint distribution $P(x_1, \ldots, x_n)$ on a computer. If we need to deal with $n$ propositions, then to store $P(x_1, \ldots, x_n)$ explicitly would require a table with $2^n$ entries--an unthinkably large number by any standard. Moreover, even if we find some economical way of storing $P(x_1, \ldots, x_n)$ (or rules for generating it), there still remains the problem of manipulating it to compute the probabilities of propositions which people consider to be interesting. For example, to compute the marginal probability $P(x_i)$ would require summing $P(x_1, \ldots, x_n)$ over all $2^{n-1}$ combinations of the remaining $n-1$ variables $x_j$, $j \neq i$. Similarly, computing the conditional probability $P(x_i|x_j)$ from its textbook definition $P(x_i|x_j) = \frac{P(x_i, x_j)}{P(x_j)}$ would involve dividing two marginal probabilities, each resulting from summation over an exponentially large number of variable combinations. Human performance, by contrast, exhibits an opposite complexity ordering; probabilistic judgments on a small number of propositions (especially 2-place conditional statements such as the likelihood that a patient suffering from a given disease will develop a certain type of complication) are issued swiftly and reliably, while judging the likelihood of a conjunction of many propositions is done with great degree of difficulty and hesitancy. This suggests that the elementary building blocks which make up human knowledge are not the entries of a joint-distribution table, but rather the low-order marginal and conditional probabilities defined over small clusters of propositions.

Further light on the structure of probabilistic knowledge can be shed by observing how people handle the notion of independence. Whereas a person may show reluctance to giving a numerical estimate for the conditional probability $P(x_i|x_j)$, no hesitation will normally be encountered when that person is asked to state merely whether $x_i$ and $x_j$ are dependent or independent, namely, whether knowing the truth of $x_j$ will or will not alter the belief in $x_i$. The 3-place relationships of conditional dependency (i.e. $x_i$ influences $x_j$ given $x_k$) are likewise judged by people with a great deal of clarity, conviction, and consistency.

This suggests that the notions of dependence and conditional dependence are more basic than the numerical values attached to probability judgments, contrary to the picture painted in most textbooks on probability theory, where the latter is presumed to provide the criterion for testing the former. Moreover, the nature of probabilistic dependency between propositions is similar in many respects to that of connectivity in graphs. For instance, we find it plausible to say that a proposition $q$ affects proposition $r$ *directly*, while $s$ influences $r$ *indirectly*, via $q$. Similarly, we find it natural to identify the set of propositions which directly affect the truth value of $q$, and to describe them as the direct neighbors of $q$, which *isolate* $q$ from all other influences. This suggests that the fundamental structure of human knowledge can be represented by dependency graphs and that mental tracing of links in these graphs are responsible for the basic steps in querying and updating that knowledge.

## 1.1 *BAYESIAN NETWORKS*

Assume that we decide to represent our perception of a certain problem domain by sketching a graph in which the nodes represent propositions and the links connect those propositions that we judge to be *directly* related. We now wish to quantify the links by weights that signify the strength and type of dependencies between the connected propositions. If these weights are to be interpreted later as conditional probabilities, two problems must first be attended to: *consistency* and *completeness*. Consistency guarantees that we do not overload the graph with an excessive number of parameters; overspecification may lead to contradictory conclusions, depending on which parameter is consulted first. Completeness protects us from underspecifying the graph dependencies.

One of the attractive features of the joint-distribution representation of probability is the transparency by which one can synthesize consistent probability models or detect inconsistencies therein. In this representation, all we need to do is to assign non-negative weights to the atomic compartments in the space (i.e., conjunctions of propositions), make sure the weights sum to one, and a complete model, free of inconsistencies is created. By contrast, the synthesis process in the graph representation is much more hazardous. For example, assume you have three propositional variables, $x_1$, $x_2$, $x_3$, and you want to express their dependencies by specifying the three pairwise probabilities $P(x_1, x_2)$, $P(x_2, x_3)$, $P(x_3, x_1)$. It turns out that this will normally lead to inconsistencies; unless the parameters given satisfy some non-obvious relationship, there exists no probability model that will support all three probabilities.

Fortunately, the consistency-completeness issue has a simple solution, stemming from the chain-rule representation of joint-distributions. Choosing an arbitrary order on the variables $x_1, \cdots x_n$ we can write:

$$P(x_1,x_2, \cdots x_n) = P(x_n|x_{n-1} \cdots x_1)P(x_{n-1}|x_{n-2} \cdots x_1) \cdots P(x_3|x_2,x_1)\, P(x_2|x_1)P(x_1)$$

In this formula, each factor contains only one variable on the left side of the conditioning bar, and in that way the formula can be used as a prescription for consistently quantifying the dependencies among the nodes of an arbitrary graph. Given a graph G, assign an arbitrary order to its nodes and impose directionality on the links pointing from low-order to high-order nodes. To each node $x_i$ assign an arbitrary function $F_i(x_i,S_i)$ satisfying

$$\sum_{x_i} F_i(x_i,S_i) = 1$$

$$0 \leq F_i(x_i,S_i) \leq 1$$

where $S_i$ is the set of $x_i$'s parents and the summation ranges over all values of $x_i$. This assignment is complete and consistent; it defines a joint distribution function given by the product:

$$P(x_1 \cdots x_n) = \prod_i F_i(x_i,S_i)$$

and the functions $F_i(x_i,S_i)$ are the marginal distributions $P(x_i|S_i)$ dictated by $P(x_1, \cdots x_n)$. For example, the distribution corresponding to the graph of Figure 1 can be written by inspection:

$$P\ (x_1, x_2, x_3, x_4, x_5, x_6) = P(x_6|x_5)\, P(x_5|x_2, x_3)\, P(x_4|x_1x_2)\, P(x_3|x_1)\, P(x_2|x_1)\, P(x_1).$$

This also leads to a simple method of constructing a dependency-graph representation to any given joint distribution $P(x_1 \cdots x_n)$. We start by imposing an arbitrary order $d$ on the set of variables, $x_1 \cdots x_n$, then choose $x_1$ as a root of the graph, and assign to it the marginal probability $P(x_1)$ dictated by $P(x_1, \cdots x_n)$. Next, we form a

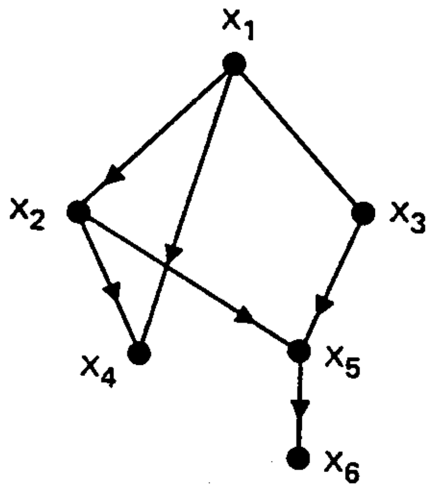


Figure 1

node to represent $x_2$; if $x_2$ is dependent on $x_1$ a link from $x_1$ to $x_2$ is established and quantified by $P(x_2|x_1)$. Otherwise, we leave $x_1$ and $x_2$ unconnected and assign the prior $P(x_2)$ to node $x_2$. At the $i^{th}$ stage, we form the node $x_i$ and establish a group of directed links to $x_i$ from the smallest subset of nodes $S_i \subseteq \{x_1 \cdots x_{i-1}\}$ satisfying the condition:

$$P(x_i|S_i) = P(x_i|x_{i-1}, \dots, x_1)$$

It is easy to show that the minimal subset $S_i$ is unique. Thus, the distribution $P(x_1, \cdots x_n)$, together with the order $d$ uniquely prescribe a set of parent nodes for each variable $x_i$, and that constitutes a full specification of a directed acyclic graph which represents the dependencies imbedded in $P(x_1, \cdots x_n)$. We shall call this graph "*Bayes Network*" or "*Influence Network*", interchangeably; the former to emphasize the judgmental origin of the quantifiers, the latter to vindicate the directionality of the links. When the nature of the interactions is perceived to be causal, then the term "*Causal Net-*

*work*" may also be appropriate. In general, however, an influence network may also represent associative or inferential dependencies, in which case the directionality of the arrows is used mainly for computational convenience.

In the strictest sense, these networks are not graphs but hypergraphs, because the dependency of a given node on its $k$ parents requires a function of $k+1$ arguments which, in general, could not be specified by $k$ two-place functions on the individual links. This, however, does not diminish the advantages of the network representation in highlighting the essential interactions between the variables, and in modelling the computational processes involved in inferential reasoning.

Note that the topology of a Bayes network may be extremely sensitive to the node ordering $d$; a network which has an inverted-tree structure in one ordering may turn into a complete graph if that ordering is reversed. For example, if $x_1, \ldots, x_n$ stands for the outcomes of $n$ independent coins and $x_{n+1}$ represents the output of a detector triggered if any of the coins comes up HEAD, then the influence network will be an inverted tree of $n$ arrows pointing from each of the variables $x_1, \ldots, x_n$ toward $x_{n+1}$. On the other hand, if the detector's outcome is chosen to be the first variable, say $x_o$, then the underlying influence network would be a complete graph.

This sensitivity may at first seem paradoxical; $d$ can be chosen arbitrarily, whereas people have fairly uniform conceptual structures, e.g., they agree on whether a pair of propositions are directly or indirectly related. The answer to this apparent paradox lies in the fact that the agreement regarding the structure of influence networks stem from the dominant role *causality* plays in the formation of these networks. Thus, the standard ordering imposed by the direction of causation also induces identical topologies on the networks that people adopt for encoding experiential knowledge. It is easy to speculate that if it were not for the social convention to adopt a standard ordering of events, conforming to the flow of time and causation, human communication would become an impossible task.

## 1.2 *CONDITIONAL INDEPENDENCE AND GRAPH SEPARABILITY*

To facilitate the verification of dependencies between the variables in a Bayes network, we need to establish a clear correspondence between the topology of the network and various types of independence. Ideally, we would have liked to associate independence between variables with the lack of connectivity between their corresponding nodes. Likewise, we would have liked to require that if the removal of some subset $S$ of nodes from the network renders nodes $x_i$ and $x_j$ disconnected, then this separation should indicate conditional independence between $x_i$ and $x_j$ given $S$, namely,

$$P(x_i|x_j, S) = P(x_i|S).$$

This would provide a clear graphical representation to the notion that $x_j$ does not affect $x_i$ directly but, rather, its influence is mediated by the variables in $S$.

Unfortunately, Bayes networks do not provide this simple representation of independence; a modified criterion of separability is required that takes into account the directionality of the arrows in the graph. Consider a triplet of variables $x_1$,$x_2$,$x_3$, where $x_1$ is connected to $x_3$ via $x_2$. The two links, connecting the pairs $(x_1, x_2)$ and $(x_2, x_3)$, can join at the midpoint $x_2$ in one of three possible ways:

(1) tail-to-tail, $x_1 \leftarrow x_2 \rightarrow x_3$

(2) head-to-tail, $x_1 \rightarrow x_2 \rightarrow x_3$ or $x_1 \leftarrow x_2 \leftarrow x_3$

(3) head-to-head, $x_1 \rightarrow x_2 \leftarrow x_3$

From the method of constructing the network, it is clear that (assuming $x_1$, $x_2$, $x_3$ are the only variables involved) in cases (1) and (2) $x_1$ and $x_3$ are conditionally independent given $x_2$, while in case (3) $x_1$ and $x_3$ are marginally independent (i.e., $P(x_3|x_1) = P(x_3)$) but may become dependent given the value of $x_2$. Moreover, if $x_2$ in case (3) has descendants $x_4$, $x_5$ $\cdots$, then $x_1$ and $x_3$ may also become dependent if any one of those descendant variables is instantiated. These considerations motivate the definition of a qualified version of path connectivity, applicable to paths with directed links, and sensitive to all the variables whose values are known at a given time.

DEFINITION: (a) A path $P$ is *connected with respect to a subset* $S_e$ of evidence variables if all successive links along P are *joined w.r.t.* $S_e$.

(b) Two links, meeting head-to-tail or tail-to-tail at node $X$, are *joined w.r.t.* $S_e$ if X is not in $S_e$.

(c) Two links meeting head-to-head at node $X$, are *joined w.r.t.* $S_e$ if $X$ or any of its descendants is in $S_e$.

This definition permits us to define *separability* with respect to a subset of observations which, in turn, provides a graphical criterion for testing conditional independence.

*DEFINITION*: A subset of variables $S_e$ is said to *separate* $x_i$ from $x_j$ if there is no path between $x_i$ and $x_j$ which is connected w.r.t. $S_e$.

It is not hard to see that if $S_e$ separates $x_i$ from $x_j$, then $x_i$ is conditionally independent of $x_j$ given $S_e$. Moreover, the procedure involved in testing separation w.r.t. a given subset $S_e$ is only slightly more complicated than that of testing whether $S_e$ is a separating cut set, and can be handled by visual inspection. In Figure 1, for example, one can easily verify that variables $x_2$ and $x_3$ are separated w.r.t. $S_e = \{x_1\}$ or $S_e = \{x_1, x_4\}$ but not w.r.t. $S_e = \{x_1, x_6\}$, because $x_6$, being a descendant of $x_5$, "joins" the head-to-head links at $x_5$, which amounts to forming a connected path between $x_2$ and $x_3$.

Note that whereas the structure of Bayes networks together with the directionality of its links depend strongly on the node ordering used in the network construction, conditional independence is a property of the underlying distribution and is, therefore, order-invariant. If we succeed to find an ordering $d$ in which a given conditional independence relationship becomes graphically transparent, that relationship will remain valid in all other orderings even though it may not induce a graph-separation pattern in the corresponding networks. This permits the use of Bayes networks for determining by inspection the *influence neighborhood* of any given node, namely, the ~~a~~ minimal set $S$ of variables that renders a given variable independent of every variable not in $S$. The separation criterion for Bayes networks dictates that the influence neighborhood consists of three types of neighbors: the direct parents, direct successors, and all direct parents of the latter. Thus, for example, in a Markov chain the influence neighborhood of any non-terminal node consists of its two immediate neighbors, while in trees the influence neighborhood consists of the (unique) father and the immediate successors. In Figure 1, however, the influence neighborhoods of $x_3$ is $\{x_1, x_5, x_2\}$.

## 1.3 *AN OUTLINE AND SUMMARY OF RESULTS*

The first part of the paper (Section 2) deals with the task of fusing and propagating the impacts of new evidence and beliefs through Bayesian networks in such a way that, when equilibrium is reached, each proposition will be assigned a certainty measure consistent with the axioms of probability theory. We first argue (Section 2.1) that any viable model of human reasoning should be able to perform this task by a self-activated propagation mechanism, i.e., by an array of simple and autonomous processors, communicating locally via the links provided by the Bayes network itself. In Section 2.2 we then show that these objectives can be fully realized in tree-structured networks, where each node has only one father. In section 2.3 we extend the result to networks with multiple parents, as long as they are singly connected, i.e., there exists only one (undirected) path between any pair of nodes. In both cases, we identify belief parameters, communication messages, and updating rules which guarantee that equilibrium is reached in time proportional to the longest path in the network and that, at equilibrium, each proposition will be accorded a belief measure consistent with probability theory. Possible approaches to achieve autonomous propagation in more general networks are discussed in Section 2.4.

Of these approaches, the second part of the paper (Section 3) explores the feasibility of turning a Bayes network into a tree by introducing dummy variables. In Section 3.1 we argue that such a technique would mimic the way people develop causal models, that the dummy variables correspond to the mental constructs known as "hidden causes", and that humans' relentless search for causal models is motivated by their desire to achieve computational features similar to those offered by tree-structured Bayes networks. After defining (in Section 3.2) the notions of star-decomposability and tree-decomposability, Section 3.3 treats triplets of random variables and asks under what conditions one is justified in attributing the observed dependencies to one central cause represented by a fourth variable. We show that these conditions are readily testable and, when the conditions are satisfied, that the parameters specifying the relations between the visible variables and the central cause can be determined uniquely. In Section 3.4 we extend these results to the case of a tree with $n$ leaves. We show that if there exists a set of dummy variables which decompose a given Bayes network into a tree, then the uniqueness of the triplets' decomposition enables us to configure that tree from pairwise dependencies among the variables. Moreover, the configuration procedure takes only $O(n \log n)$ steps. In Section 3.5 we evaluate the merits of this method and address the difficult issues of estimation and approximation.

# 2. FUSION AND PROPAGATION

## 2.1 *AUTONOMOUS PROPAGATION AS A COMPUTATIONAL PARADIGM*

Once an influence network is constructed, it can be used to represent the generic causal knowledge of a given domain, and can be consulted to reason about the interpretation of specific input data. The interpretation process involves instantiating a set of variables corresponding to the input data and calculating its impact on the probabilities of a set of variables designated as hypotheses. In general, this process can be executed by an external interpreter who may have access to all parts of the network, may use its own computational facilities, and may schedule its computational steps so as to take full advantage of the network topology with respect to the incoming data. However, the use of such an interpreter seems foreign to the reasoning process normally exhibited by humans [Shastri and Feldman, 1984]. Our limited short-term memory and narrow focus of attention, combined with our inflexibility of shifting rapidly between alternative lines of reasoning seem to suggest that our reasoning process is fairly local, progressing incrementally along prescribed pathways. Moreover, the speed and ease with which we perform some of the low level interpretive functions, such as recognizing scenes, comprehending text, and even understanding stories, strongly suggest that these processes involve a significant amount of parallelism, and that most of the processing is done at the knowledge level itself, not external to it.

A paradigm for modelling such phenomena would be to view an influence network not merely as a passive parsimonious code for storing factual knowledge but also as a computational architecture for reasoning about that knowledge. That means that the links in the network should be treated as the only pathways and activation centers that direct and propel the flow of data in the process of querying and updating beliefs. Accordingly, we assume that each node in the network is designated a separate processor which both maintains the parameters of belief for the host variable and manages the communication links to and from the set of neighboring, logically related, variables. The communication lines are assumed to be open at all times, i.e., each processor may at any time interrogate the belief parameters associated with its neighbors and update its own. In this fashion the impact of new evidence may propagate up and down the network until equilibrium is reached.

The ability to update beliefs by an autonomous propagation mechanism also has a profound effect on sequential implementations of evidential reasoning. Of course, when this architecture is simulated on sequential machines, the notion of autonomous processors working simultaneously in time is only a metaphor; however, it signifies the complete separation of the stored knowledge and the individual computations from the control mechanism which schedules these computations to achieve some control strategy goal. This guarantees an ultimate flexibility for a sequential controller; the computations can be performed in any order, without the need to remember which parts of the network have or have not been updated already. Thus, for example, belief updating may be activated by changes occurring in logically related propositions, by requests for evidence arriving from a central supervisor, by a predetermined schedule, or entirely at random. The communication and interaction between individual processes can be simulated using a blackboard architecture [Lesser and Erman, 1977] where each proposition is designated specific areas of memory to access and modify. Additionally, the uniformity of this propagation scheme renders it natural for formulation in object-oriented languages: each node is an object of the same generic type and the belief parameters are the messages by which interacting objects communicate.

The asynchronous nature of this model requires a solution to an instability problem. If a stronger belief in a given hypothesis means a greater expectation for the occurrence of a certain manifestation and if, in turn, a greater certainty in the occurrence of that manifestation adds further credence to the hypothesis, how can one avoid an infinite updating loop when the two processors begin to communicate with one another? We will show that in singly-connected networks such looping can be avoided by maintaining several belief parameters, one for each link, to identify the individual sources of belief in addition to its overall magnitude. Thus, a major objective of the next two sections is to present an appropriate set of belief parameters, communication messages, and updating rules which guarantee that the diffusion of updated beliefs is accomplished in a single pass and complies with the tenets of Bayes calculus.

## 2.2 *BELIEF PROPAGATION IN TREES*

We shall consider influence networks which are tree structured, namely, every node has only one incoming link. Additionally, we allow each node to represent a multivalued variable which may represent a collection of mutually exclusive hypotheses (e.g., identity of organism: $ORG_1$, $ORG_2$,...) or a collection of possible observations (e.g. patient's temperature: high, medium, low). Let a variable be labeled by a capital letter, e.g., $A$, $B$, $C$, ..., and its possible values subscripted, e.g., $A_1$, $A_2$, ..., $A_n$. Each directed link $A \rightarrow B$ is quantified by a fixed conditional probability matrix, $\underset{=}{M}(B|A)$, with entries: $M(B|A)_{ij} = P(B_j|A_i)$. Normally, the directionality of the arrow designates $A$ as the set of causal hypotheses and $B$ as the set of indicators or manifestations for these hypotheses.

The instantiated nodes, constituting the incoming evidence or *data* will be denoted by $D$. For the sake of clarity we will distinguish between the fixed conditional probabilities that label the links, e.g. $P(A|B)$, and the dynamic values of the updated node probabilities. The latter will be denoted by $\mathrm{BEL}(A_i)$, which reflects the overall belief accorded to proposition $A_i$ by all data so far received. Thus,

$$\mathrm{BEL}(A_i) \overset{\Delta}{=} P(A_i|D)$$

where $D$ is the value combination of all instantiated variables.

Consider the fragment of a tree depicted in Figure 2. The belief in the various values of $B$ depends on three distinct sets of data: i.e., data from the tree rooted at $B$, from the tree rooted at $C$, and from the tree above $A$. However, since $A$ separates $B$ from all variables except its descendants (see Section 1.2), this implies that the influence of the latter two sources of information on $B$ are completely summarized by their effect on $A$. More formally, let $D_d(B)$ stand for the data obtained from the tree rooted at $B$, and $D^u(B)$ for the data obtained from the rest of the network. We have

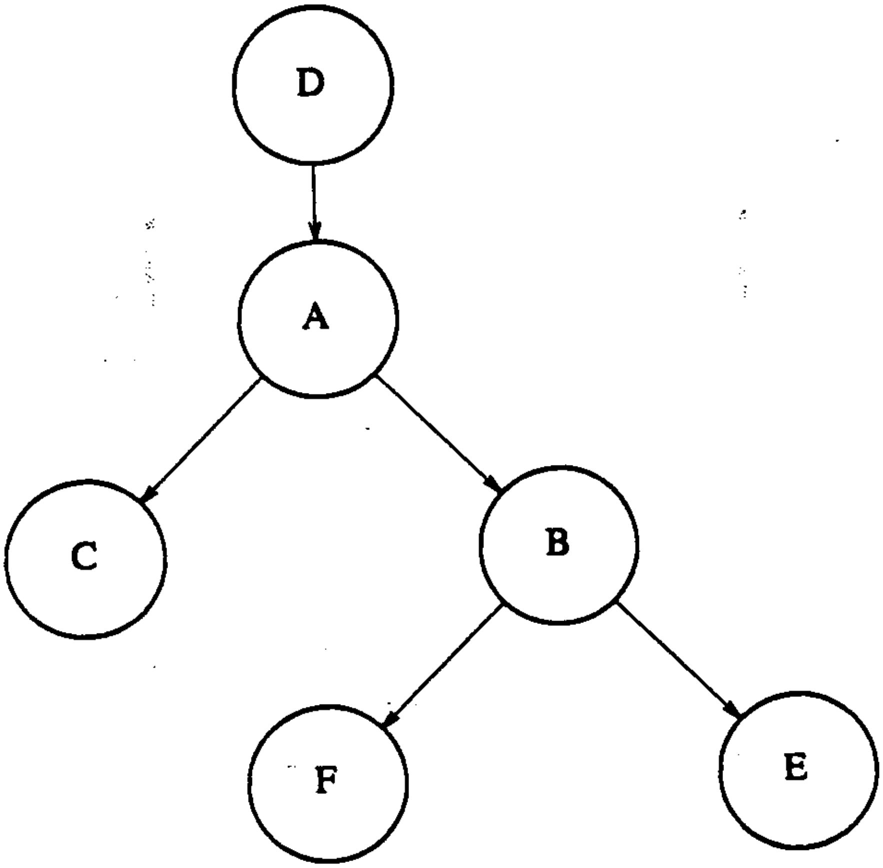


Figure 2

$$P(B_j|A_i, D^u(B)) = P(B_j|A_i) \tag{1}$$

which also leads to the usual "inter-siblings" conditional independence:

$$P(B_i, C_k|A_j) = P(B_i|A_j) \cdot P(C_k|A_j), \tag{2}$$

since the proposition $C = C_k$ is part of $D^u(B)$.

### Data Fusion

Assume we wish to find the belief induced on $B$ by some data $D$, part of which, $D^u(B)$, comes from above $B$ and part, $D_d(B)$, from below. Bayes theorem, together with (1), yields the product rule

$$\mathrm{BEL}(B_i) = P(B_i|D^u(B), D_d(B)) = \alpha P[D_d(B)|B_i] \cdot P[B_i|D^u(B)], \tag{3}$$

where $\alpha$ is a normalizing constant. This is a generalization of the celebrated Bayes formula for binary variables

$$O(H|E) = \lambda(E)\, O(H) \tag{4}$$

where $\lambda(E) = P(E|H)/P(E|\overline{H})$ is known as the likelihood ratio, and $O(H) = P(H)/P(\overline{H})$ as the prior odds [2].

Eq.(3) generalizes (4) in two ways. First, it permits the treatment of non-binary variables where the mental task of estimating $P(E|\overline{H})$ is often unnatural, and where conditional independence with respect to the negations of the hypotheses is normally violated (i.e., $P(E_1, E_2|\overline{H}) \neq P(E_1|\overline{H})P(E_2|\overline{H})$). Second, it identifies a surrogate to the prior probability term for any intermediate node in the tree, even *after* obtaining some evidential data. According to (3), the multiplicative role of the prior probability has been taken over by that portion of belief which is based *only* on the evidence gathered by the network *above* a variable, i.e., excluding the data collected from its descendants. Thus, the product rule (3) can be applied to any node in the network, without requiring a separate prior probability assessment. The root is the only node which requires a prior probability estimation. Since it has no network above, $D^u$(root) should be interpreted as the available background knowledge which remains unexplicated by the network below.

Eq.(3) suggests that the probability distribution of every variable in the network can be computed if the node corresponding to that variable contains the parameters

$$\lambda(B_i) = P(D_d(B)|B_i) \tag{5}$$

and

$$q(B_i) = P(B_i|D^u(B)). \tag{6}$$

$q(B_i)$ represents the causal or *prospective* support attributed to $B_i$ by its ancestors and $\lambda(B_i)$ represents the diagnostic or *retrospective* support $B_i$ receives from its descendants. The total strength of belief in $B_i$ would be obtained by *fusing* these two supports via the product

$$\text{BEL}(B_i) = \alpha\lambda(B_i)\, q(B_i). \tag{7}$$

Whereas only two parameters, $\lambda(E)$ and $O(H)$, were sufficient for binary variables, an $n$-valued variable needs to be characterized by two n-tuples:

$$\lambda(B) = \lambda(B_1), \lambda(B_2), \ldots, \lambda(B_n) \tag{8}$$

$$\underline{q}(B) = q(B_1), q(B_2), \ldots, q(B_n). \tag{9}$$

To see how information from several descendants fuse at node $B$, note that the data $D_d(B)$ in (5) can be partitioned into disjoint subsets, $D_d^1, D_d^2, \ldots, D_d^m$, one for each subtree emanating from (the $m$ children of) $B$. Since $B$ "separates" these subtrees apart, conditional independence holds:

$$\lambda(B_i) = P(D_d(B)|B_i) = \prod_k P(D_d^k|B_i) \tag{10}$$

and, so, $\lambda(B_i)$ can be formed as a product of the terms $P(D_d^k|B_i)$ if these are delivered to processor $B$ as messages from its children.

Thus, we see that, at each node of a Bayes tree, the fusion of all incoming data is purely multiplicative.

**Propagation Mechanism**

Assuming that the vectors $\lambda$ and $\underline{q}$ are stored with each node of the network, our task is now to prescribe how the influence of new information spreads through the network, namely, how the parameters $q$ and $\lambda$ of a given node can be determined from the $q$'s and $\lambda$'s of its neighbors. This is done easily by conditioning eqs.(5) and (6) on all the values that the neighbors can assume. For example, suppose $E$ is the $k^{th}$ son of $B_i$. To compute the $k^{th}$ multiplicand in the product of (10) from the value of $\lambda(E)$, we write

$$P(D_d^k|B_i) = \sum_j P(D_d(E)|B_i, E_j)\, P(E_j|B_i)$$

and obtain (using (1) and (5))

$$P(D_d^k|B_i) = \sum_j \lambda(E_j)\, P(E_j|B_i)$$

Namely, $P(D_d^k|B_i)$ is obtained by taking the $\lambda$ vector stored at the $k^{th}$ son of $B$ and multiplying it by the fixed conditional-probability matrix that quantifies the link between $B$ and $E$. Thus, the $\lambda$ vector of each node can be computed from the $\lambda$'s of its children by multiplying the latter by their respective link matrices, and then multiplying the resultant vectors together, term-by-term, as shown in (10).

A similar analysis, applied to the vector $q$, shows that the $q$ of any node can be computed from the $q$ of its father and the $\lambda$'s of its siblings, again after multiplication by the corresponding link matrices (see Appendix I). Moreover, no direct communication with the siblings is necessary since the information required of them already resides at the father's site (for the purpose of calculating its $\lambda$, as in (10)) and, so, it can be delivered down to the requesting son. These results, together with some efficiency considerations [Pearl, 1982], dictate the following propagation scheme (the proofs of validity are outlined in Appendix I).

1. Each processor computes two message vectors: $\underline{P}$ and $\underline{r}$. $\underline{P}$ is sent to every son while $\underline{r}$ is delivered to the father. The message $\underline{P}$ is identical to the belief distribution BEL of the sender and is computed from $\lambda$ and $q$ using (7). $\underline{r}$ is computed from $\lambda$ using the matrix multiplication:

$$\underline{r} = \underline{\underline{M}} \cdot \lambda \tag{11}$$

where $\underline{\underline{M}}$ is the matrix quantifying the link to the father. Thus, the dimensionality of $\underline{r}$

is equal to the number of hypotheses managed by the father. Each component of $\underline{r}$ represents the diagnostic contribution of the data below the host processor to the belief in one of the father's hypotheses. It corresponds to a single term of the product in (10).

2. When processor $B$ is called to update its parameters, it simultaneously inspects the $\underline{P}(A)$ message communicated by the father $A$ and the messages $\underline{r}_1$, $\underline{r}_2$, ..., communicated by each of its sons and acknowledges receiving the latter. Using these inputs, it then updates $\lambda$ and $\underline{q}$ as follows:

3. *Bottom-up propagation*: $\lambda$ is computed using a term-by-term multiplication of the vectors $\underline{r}_1$, $\underline{r}_2$, ..., (as in (10)):

$$\lambda(B_i) = (\underline{r}_1)_i \times (\underline{r}_2)_i \times \cdots = \prod_k (\underline{r}_k)_i \tag{12}$$

4. *Top-down propagation*: $\underline{q}$ is computed using:

$$q(B_i) = \beta \prod_j P(B_i|A_j)P(A_j)/(\underline{r}')_j \tag{13}$$

where $\beta$ is a normalizing constant and $\underline{r}'$ is the last message from $B$ to $A$ acknowledged by the father $A$. (The division by $\underline{r}'$ amounts to removing from $\underline{P}(A)$ the contribution due to $D_d(B)$ as dictated by the definition of $q$ in (6). An alternative way, avoiding this division, would be to obtain from the father directly the message $\underline{q}(A)\prod_k(\underline{r}_k)$ (i.e., $\frac{\underline{P}(A)}{\underline{r}'}$) where $k$ ranges over the siblings of $B$.)

5. Using the updated values of $\lambda$ and $\underline{q}$, the messages $\underline{p}$ and $\underline{r}$ are then recomputed as in step 1 and are posted on the message-boards reserved for the sons and the father, respectively.

This updating scheme is shown schematically in Figure 3, where multiplications and divisions of any two vectors stand for term-by-term operations.

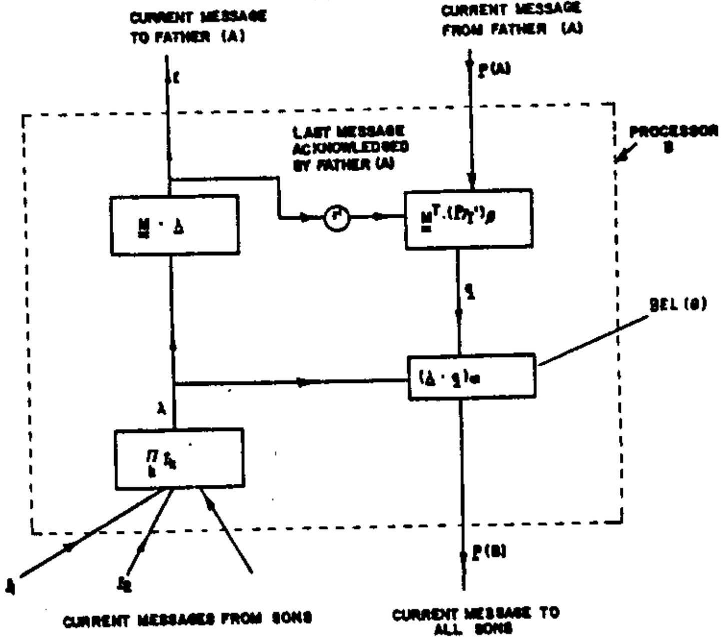

CURRENT MESSAGE TO FATHER (A)
CURRENT MESSAGE FROM FATHER (A)
$\underline{r}$
$\underline{p}$(A)
LAST MESSAGE ACKNOWLEDGED BY FATHER (A)
PROCESSOR B
$\underline{\underline{M}} \cdot \lambda$
$\underline{\underline{M}}^T \cdot (\underline{p}/\underline{r}')_\beta$
$\underline{q}$
BEL (B)
$(\lambda \cdot \underline{q})_\alpha$
$\lambda$
$\prod_k \lambda_k$
$\lambda_1$
$\lambda_2$
$\underline{p}$(B)
CURRENT MESSAGES FROM SONS
CURRENT MESSAGE TO ALL SONS

Figure 3

Terminal and data nodes in the tree require special treatments. Here we have to distinguish between the two cases:

1. *Anticipatory node*: a leaf node that has not been instantiated yet. For such variables, $\underline{p}$ should be equal to $\underline{q}$ and, therefore, we should set $\lambda=(1,1,...,1)$ (also implying $\underline{r}=(1,1,...,1)$).

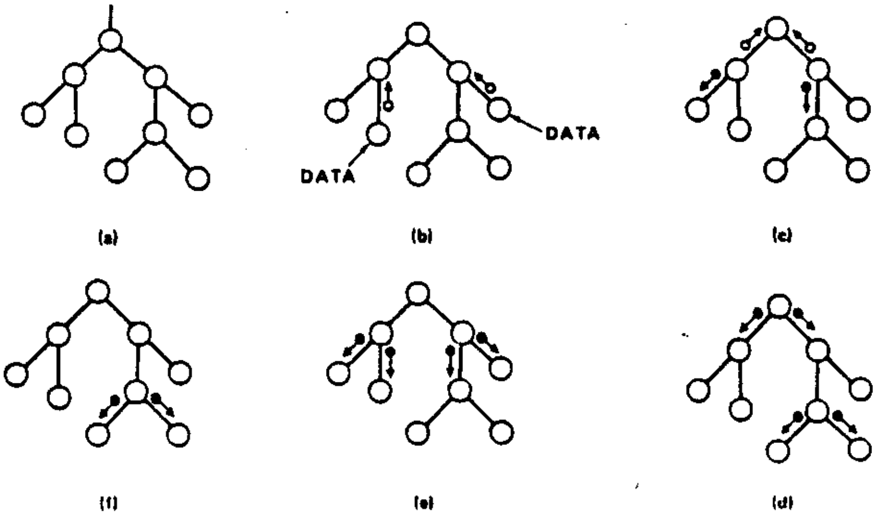
2. *Data-node*: a variable with instantiated value. Following eqs.(5) and (6), if the $j^{th}$ state of $B$ was observed to be true, we set $\lambda = \underline{q} = (0, ..., 0,1,0, ..., 0)$ with 1 at the $j^{th}$ position.

Similarly, the boundary condition for the root node is established by substituting the prior probability instead of the message $\underline{P}(A)$ expected from the father.

**An Illustration**

Figure 4 shows six successive stages of belief propagation through a simple binary tree, assuming that updating is activated by changes in the belief parameters of neighboring processes. Initially (Figure 4a), the tree is in equilibrium and all terminal nodes are anticipatory. As soon as two data nodes are activated (Figure 4b), white tokens are placed on their links, directed towards their fathers. In the next phase, the fathers, activated by these tokens, absorb the latter and manufacture the appropriate number of tokens for their neighbors (Figure 4c): white tokens for their fathers and black ones for the children (the links through which the absorbed tokens have entered do not receive new tokens, thus reflecting the division of $\underline{P}$ by $\underline{r}'$). The root node now receives two white tokens, one from each of its descendants. That triggers the production of two black tokens for top-down delivery (Figure 4d). The process continues in this fashion until, after six cycles, all tokens are absorbed and the network reaches a new equilibrium.

Figure 4



**Properties of the Updating Scheme**

1. The local computations required by the updating scheme are efficient in both storage and time. For an $m$-ary tree with $n$ values per node, each processor should store $n^2+mn+2n$ real numbers, and perform $2n^2+mn+2n$ multiplications per update.

2. The local computations and the final belief distribution are entirely independent of the control mechanism that activates the individual operations. They can be activated by either data-driven or goal-driven (e.g., requests for evidence) control strategies, by a clock, or at random.

3. New information diffuses through the network in a single pass. Infinite relaxations have been eliminated by maintaining a two-parameter system ($\underline{q}$ and $\underline{r}$) to decouple top and bottom evidences. The time required for completing the diffusion (in parallel) is equal to the diameter of the network.

## 2.3 *PROPAGATION IN SINGLY-CONNECTED NETWORKS*

The tree structures treated in the preceding section require that only one variable be considered a cause of any other variable. This restriction simplifies computations but its representational power is rather limited, because it forces us to group together all causal factors sharing a common consequence into a single node. By contrast, when people associate a given observation with multiple potential causes, they weigh one causal factor against another as independent variables, each pointing to a specialized area of knowledge. As an illustration, consider the following situation:

> Mr. Holmes received a phone call from his neighbor notifying him that she heard a burglar alarm sound from the direction of his home. As he was preparing to rush home, Mr. Holmes recalled that the last time the alarm had been triggered by an earthquake. Driving home, he heard a radio newscast reporting an earthquake 200 miles away [Kim and Pearl, 1983].

Mr. Holmes perceives two episodes which may be potential causes for the alarm sound, an attempted burglary and an earthquake. Even though burglaries can safely be assumed independent of earthquakes, still the radio announcement reduces the likelihood of a burglary, as it "explains away" the alarm sound. Moreover, the two causal events are perceived as individual variables each pointing to a separate frame of knowledge.

This interaction among multiple causes is a prevailing pattern of human reasoning. When a physician discovers evidence in favor of one disease, it reduces the credibility of other diseases, although the patient may as well be suffering from two or more disorders simultaneously. A suspect who provides an alternative explanation for being present at the scene of the crime appears less likely to be guilty, even though the explanation furnished does not preclude his committing the crime.

This section extends the propagation scheme to graph structures which permit a node to have multiple parents and captures "sideways" interactions via common successors. However, the graphs are restricted to be *singly connected*, namely, at most one path (undirected) exists between any pair of nodes.

**Fusion Equations**

Consider a fragment of a singly connected network, as depicted in Figure 5. The link $B \rightarrow A$ partitions the graph into two parts: an upper subgraph $G^+_{BA}$, and a lower subgraph $G^-_{BA}$, the complement of $G^+_{BA}$. These two graphs contain two sets of *data* which we shall call $D^+_{BA}$ and $D^-_{BA}$, respectively. Likewise, the links $C \rightarrow A$, $A \rightarrow X$, and $A \rightarrow Y$ define the subgraphs $G^+_{CA}$, $G^-_{AX}$, and $G^-_{AY}$ which contain the data sets $D^+_{CA}$, $D^-_{AX}$, and $D^-_{AY}$, respectively. Since $A$ is a common child of $B$ and $C$, it does not separate $G^+_{BA}$ and $G^+_{CA}$ apart. However, it does separate the following three subgraphs: $G^+_{BA} \bigcup G^+_{CA}$, $G^-_{AX}$, and $G^-_{AY}$, and we can write

$$P(D^+_{BA}, D^+_{CA}, D^-_{AX}, D^-_{AY}|A_i) = P(D^+_{BA}, D^+_{CA}|A_i)\, P(D^-_{AX}|A_i)\, P(D^-_{AY}|A_i) \tag{14}$$

Thus, using Bayes rule, the overall strength of belief in $A_i$ can be written:

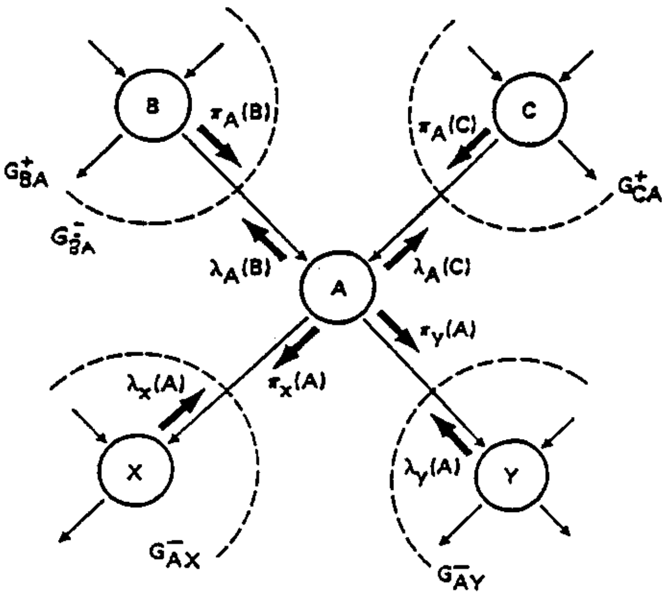


Figure 5

$$\mathrm{BEL}(A_i) = P(A_i|D^+_{BA}, D^+_{CA}, D^-_{AX}, D^-_{AY}) = \alpha\, P(A_i|D^+_{BA}, D^+_{CA})\, P(D^-_{AX}|A_i)\, P(D^-_{AY}|A_i) \tag{15}$$

where $\alpha$ is a normalizing constant. Further partitioning over the values of $B$ and $C$, we get

$$\mathrm{BEL}(A_i) = \alpha\, P(D^-_{AX}|A_i)\, P(D^-_{AY}|A_i)[\sum_{jk} P(A_i|B_jC_k)\, P(B_j|D^+_{BA})\, P(C_k|D^+_{CA})]. \tag{16}$$

Eq.(16) shows that the probability distribution of each variable $A$ in the network could be computed if three types of parameters are made available: (1) the current strength of the causal evidence, $\pi$, contributed by each incoming link to $A$;

$$\pi_A(B_j) = P(B_j|D^+_{BA}) \tag{17}$$

(2) the current strength of the diagnostic evidence, $\lambda$, contributed by each outgoing link from $A$;

$$\lambda_X(A_i) = P(D^-_{AX})|A_i) \tag{18}$$

and (3) the fixed conditional probability matrix, $P(A|B, C)$, which relates the variable $A$ to its immediate causes. Accordingly, we let each link carry two dynamic parameters, $\pi$ and $\lambda$, and let each node store the information contained in $P(A|B,C)$.

With these parameters at hand, the fusion equation (16) becomes

$$\text{BEL}(A_i) = \alpha \; \lambda_x(A_i) \; \lambda_Y(A_i) \sum_{jk} P(A_i|B_jC_k) \; \pi_A(B_j) \; \pi_A(C_k) \tag{19}$$

Alternatively, if the two parameters $\pi$ and $\lambda$ are available at a given link, we can compute the belief distribution of the parent node by the product

$$\text{BEL}(B_j) = \alpha \; \pi_A(B_j) \; \lambda_A(B_j) \tag{20}$$

**Propagation Equations**

Assuming that the vectors $\underline{\pi}$ and $\underline{\lambda}$ are stored with each link, our task is now to prescribe how the influence of new information spreads through the network.

*Updating* $\underline{\lambda}$

Starting from the definition of $\lambda_A(B_i) = P(D^-_{BA}|B_i)$ we partition the data $D^-_{BA}$ into its components: $A$, $D^-_{AX}$, $D^-_{AY}$, $D^+_{CA}$, and summing over all values of $A$ and $C$ we get

$$\lambda_A(B_i) = \alpha \sum_j [\pi_A(C_j) \sum_k \lambda_X(A_k) \; \lambda_Y(A_k) \; P(A_k|B_iC_j)]. \tag{21}$$

Eq.(21) shows that only three parameters (in addition to the conditional probabilities $P(A|B, C)$) need to be involved in updating the diagnostic parameter vector $\lambda_A(B)$:

$\pi_A(C)$, $\lambda_X(A)$, and $\lambda_Y(A)$. This is expected since $D^-_{BA}$ is completely summarized by $X$, $Y$, and $C$.

### *Updating* $\pi$

The rule for updating the causal parameter $\pi_X(A)$ can be obtained from the formula:

$$\pi_X(A_i) = \alpha\lambda_Y(A_i)[\sum_{jk} P(A_i|B_jC_k) \ \pi_A(B_j)\pi_A(C_k)] \tag{22}$$

Thus, similar to $\lambda_A(B)$, $\pi_X(A)$ is also determined by three neighboring parameters: $\lambda_Y(A)$, $\pi_A(B)$, and $\pi_A(C)$.

Eqs.(21) and (22) also demonstrate that a perturbation of the causal parameter, $\pi$, will not affect the diagnostic parameter, $\lambda$, on the same link, and vice versa. The two are orthogonal to each other since they depend on two disjoint sets of data. Therefore, any perturbation of beliefs due to new evidence propagates through the network and is absorbed at the boundary without reflection. A new equilibrium state will be reached after a finite number of updates which, in the worst case, is equal to the diameter of the network.

Eq.(21) also reveals that if no data is observed below $A$, (i.e., all $\lambda$'s pointing to $A$ are unit vectors), then all $\lambda$'s emanating from $A$ are also unit vectors. This means that evidence gathered at a node does not influence its spouses until their common son gathers diagnostic evidence. This reflects the special connectivity conditions established in Section 1.2, and matches our intuition regarding multiple causes. In Mr. Holmes's case, for example, seismic data pertaining to earthquakes would not have influenced the likelihood of a burglary prior to receiving the neighbor's telephone call.

## 2.4 *SUMMARY*

The preceding two sections show that the architectural objectives of propagating beliefs coherently, through an active network of primitive, identical, and autonomous processors can be fully realized in singly-connected graphs. Instabilities due to cyclic inferences are avoided by using multiple, source-identified belief parameters, and equilibrium is guaranteed to be reached in time proportional to the network diameter.

The primitive processors are simple, repetitive, and save for performing the matrix multiplications, require no working memory. Thus, this architecture lends itself naturally to hardware implementation, capable of real-time interpretation of rapidly changing data. It also provides a reasonable model of neural nets involved in cognitive tasks such as visual recognition, reading comprehension [Rumelhart, 1976], and associative retrieval [Anderson, 1983], where unsupervised parallelism is an uncontested mechanism.

It is also interesting to note that the marginal conditional probabilities on the links of the network retain their viability throughout the updating process. This is remarkable because $P(A|B)$ only defines the belief of $A$ under very special sets of circumstances, namely, when the value of $B$ is known with absolute certainty, and when no other evidential data is available. In normal circumstances, though, all internal nodes in the network are subject to some uncertainty and, more seriously, after observing evidence $e$, the relation between BEL($A$) and BEL($B$) is no longer governed by $P(A|B)$, but by $P(A|B, e)$, which may be vastly different. The ability to maintain a constant set of weights on the links is essential, since having to adjust the weights with the arrival of each new data would be computationally prohibitive. One is tempted to speculate, therefore, that this may be the reason that people choose the marginal conditional probabilities as standard primitives for organizing stable conceptual information which, in turn, also explains why people are more proficient in assessing the magnitude of these relationships rather than of any other probabilistic quantity.

The efficacy of singly-connected networks in supporting autonomous propagation raises the question of whether similar propagation mechanisms exist in less restrictive networks (e.g., the one in Figure 1), in which multiple parents possess common ancestors, thus forming (undirected) loops. So far, our investigation has failed to find a propagation method for loops that retains all the advantages cited above. For example, a straightforward way of handling the network of Figure 1 would be to appoint a local interpreter for the loop $x_1$, $x_2$, $x_3$, $x_5$ that will pass messages directly between $x_1$ and $x_5$, accounting for the interactions between $x_2$ and $x_3$. This amounts basically to collapsing nodes $x_2$ and $x_3$ into a single node, representing the compound variable $(x_1, x_2)$. The method works well on small loops, but as soon as the number of variables exceeds 3 or 4, collapsing requires handling huge matrices and washes away the natural conceptual structure imbedded in the original network.

An alternative method would be for each node to continue communicating with its neighbors as if the network was singly-connected, ignoring the possibility of loops. This will set up messages circulating indefinitely around the loop, until equilibrium is approached. The convergence and coherence properties of such a process are yet uncertain.

A third method of propagation is based on "stochastic relaxation" [Geman and Geman, 1984]. Each processor interrogates the states of the variables within its influence neighborhood (see Section 1.2), computes a belief distribution for the values of its host variable, then selects one of these values with probability that equals the computed belief. The value chosen will subsequently be interrogated by the neighbors upon computing their beliefs, and so on. This scheme is guaranteed convergence, but usually requires very long relaxation times to reach a steady state.

Finally, an approach which is discussed more fully in Section 3 is based on the introduction of auxiliary variables that turn the network into a tree. Consider, for example, the tree of Figure 2. The leaves $C$, $D$, $E$, $F$ are tightly coupled in the sense that no two of them can separate the other two. Therefore, if we were to construct an influence network based on these variables *alone*, a complete graph would ensue. Yet, the inclusion of the variables $A$ and $B$ manages to turn that graph into a tree. The question is now: Which networks can be broken up into trees by introducing dummy variables? In some respect, this method is similar to that of appointing external interpreters to handle non-separable components of the graph, because the dummy variables are assigned processors that mediate between the original variables. However, the dummy-variables scheme enjoys the added advantage of uniformity: the processors representing the dummy variables can be identical to those representing the real variables, in full compliance with our architectural objectives. Moreover, there are strong reasons to believe that the process of reorganizing data structures by adding fictitious variables mimics an important component of conceptual development in human beings. These considerations are discussed in the section that follows.

## 3. STRUCTURING CAUSAL TREES

### 3.1 *CAUSALITY, CONDITIONAL INDEPENDENCE AND TREE ARCHITECTURE*

Human beings exhibit an almost obsessive urge to conceptually mold empirical phenomena into structures of cause-and-effect relationships. This tendency is, in fact, so compulsive that it sometimes comes at the expense of precision and often requires the invention of hypothetical, unobservable entities such as "ego", "elementary particles", and "supreme beings" to make theories fit the mold of causal schema. When we try to explain the actions of another person, for example, we invariably invoke abstract notions of mental states, social attitudes, beliefs, goals, plans and intentions. Medical knowledge, likewise, is organized into causal hierarchies of invading organisms, physical disorders, complications, clinical states, and only finally, the visible symptoms.

We take the position that, like many other psychological compulsions, human obsession with causation is computationally motivated. Causal models are only attractive because they provide effective data-structures for representing empirical knowledge, namely, they can be queried and updated at high speed with minimal external supervision. This position behooves us to take a closer look at the structure of causal models and determine what it is that makes them so effective. In other words, what are the computational assets of those fictitious variables called "causes" that make them worthy of such relentless human pursuit, and what renders causal explanations so pleasing and comforting once they are found?

The paradigm expounded in this paper is that the main ingredient responsible for the pervasive role of causal models is their *centrally-organized architecture*, i.e., an architecture in which the dependencies among the variables are mediated by one central mechanism.

If you ask *n* persons in the street what time it is, the answers will undoubtedly be very similar. Yet instead of suggesting that somehow the answers evoked, or the persons surveyed tend to influence each other, we postulate the existence of a central cause, the standard time, and the commitment of each person to adhere to that standard. Thus, instead of dealing with a complex *n*-ary relation, the causal model in this example consists of a network of *n* binary relations, all connected star-like to one central node which serves to dispatch information to and from the connecting variables. Psychologically, this architecture is much more pleasing. Since the activity of each variable is constrained by only one source of information (i.e., the central cause), no conflict in activity arises; any assignment of values consistent with the central constraints will also be globally consistent, and moreover, a change in any of the variables can communicate its impact to all other variables in only two steps.

Computationally speaking, causes are names given to auxiliary variables which encode a summary of the interaction between the visible variables and, once calculated, permit us to treat the visible variables as if they were mutually independent.

The dual summarizing-decomposing role of a causal variable is analogous to that of an orchestra conductor; it achieves coordinated behavior through central communication and thereby relieves the players from having to communicate directly with each other. In the physical sciences, a classical example of such coordination is exhibited by the construct of a *field* (e.g. gravitational, electrical, or magnetic). Although there is a one-to-one mathematical correspondence between the electric field and the electric charges in terms of which it is defined, nearly every physicist takes the next step, and ascribes physical reality to the electric field, imagining that in every point of space there is some real physical phenomenon taking place which determines the magnitude and direction which tag the point. This psychological construct offers a tremendous advantage without which it is hard to conceive the development of the electrical science. It decomposes the complex phenomena associated with interacting electric charges into two independent processes: (1) the creation of the field at a given point by the surrounding charges and (2) the conversion of the field into a physical force once another charge passes by that point.

The advantages of centrally coordinated architectures are not unique to star-structured networks, but are also characteristic of tree structures, because every internal node in the tree centrally coordinates the activities of its neighbors which, otherwise, are completely independent of each other. In a management hierarchy, for example, where employees can only communicate with each other through their immediate superiors, the passage of information is swift, economical, conflict-free, and highly parallel. These computational attributes, we postulate, give rise to the satisfying sensation called "in-depth understanding", which people experience when they discover causal models consistent with observations.

The topological concept of central coordination is parallel to the probabilistic notion of *conditional independence*. In our preceding example, the answers to the question "what time it is" would be viewed as random variables that are bound together by a *spurious correlation* (Simon 1952, Suppes 1970) and become independent of each other once we know the state of the mechanism causing the correlation, i.e., the standard time. Thus, conditional independence captures both functions of our orchestra conductor: coordination and decomposition.

The most familiar connection between causality and conditional independence is embodied in the scientific notion of a *state*. It was devised to break up the influence that the past exerts on the future by providing a sufficiently detailed description of the present. In probabilistic terms this came to be known as a Markov property; future events are conditionally independent of past events, given the current state of affairs.

Conditional independence, however, is not limited to separating the past from the future; it is often induced on events which occur at the same time and constitutes, in fact, the most universal and distinctive characteristic featured by the notion of causality. In medical diagnosis, for example, a group of co-occurring symptoms often become independent of each other once we know the disease that caused them. When some of the symptoms directly influence each other, the medical profession invents a name for that interaction (e.g., complication, clinical state, etc.) and treats it as a new auxiliary variable which again assumes the decompositional role characteristic of causal agents; knowing the exact state of the auxiliary variable renders the interacting symptoms independent of each other. In other words, it constitutes a sufficient summary for determining the likely development of each individual symptom in the group, and so, additional knowledge regarding the states of the other symptoms becomes superfluous.

Given that tree-dependence captures the main feature of causation and that it provides a convenient computational medium for performing updating and predictions, it is very suggestive to associate causal models with tree-structured Bayes networks, like those treated in Section 2.2. However, unlike with the trees of Section 2.2, we now assume that only the leaves are directly accessible to empirical observations and the internal nodes represent hidden causes; namely, we do not know any of the conditional probabilities that link the internal nodes to the leaves, nor the structure of the tree--those would have to be learned. The Bayesian network corresponding to the leaves only, will most probably be a complete graph. However, to be able to use the computational advantages of tree structures, as developed in Section 2.2, we invent new variables and attempt to restructure the network into a tree.

Our first task would be to assume that there exist dummy variables which decompose the network into a tree, and then ask whether the internal structure of such a tree can be determined from observations made solely on the leaves. If it can, then the structure found would constitute an operational definition for the hidden causes often found in causal models. Additionally, if we take the view that "learning" entails the acquisition of computationally effective representations for nature's regularities, then the procedure of configuring the tree may reflect an important component of human learning.

A related structuring task was treated by Chow and Liu (1968), who also used tree-dependent random variables to approximate an arbitrary joint distribution. However, in Chow's trees all nodes denote observed variables, and so, the conditional probabilities for any pair of variables is assumed given. By contrast, the internal nodes in our trees denote dummy variables, artificially concocted to make the representation tree-like. The problem of configuring probabilistic models using auxiliary variables is mentioned by Hinton *et al*. (1984) as one of the tasks that a Boltzmann machine should be able to solve. However, no performance results have been reported and it is not clear whether the relaxation techniques employed by the Boltzmann machine can easily escape from local minima and whether it can readily accept the restriction that the resulting structure be a tree. The method described in the following sections offers a solution to this problem, but it assumes some restrictive conditions: all variables are bi-valued, a solution tree is assumed to exist and all inter-leaf correlations are known precisely.

## 3.2 *PROBLEM DEFINITION AND NOMENCLATURE*

Consider a set of $n$ binary-valued random variables $x_1, \cdots, x_n$ with a given probability mass function $P(x_1, \cdots, x_n)$. We address the problem of representing $P$ as a marginal of an $(n+1)$-variable distribution $P_S(x_1, \ldots, x_n, w)$, that renders $x_1, \cdots, x_n$ conditionally independent given $w$, i.e.

$$P_s(x_1, \cdots, x_n, w) = \prod_{i=1}^{n} P_s(x_i|w) P_s(w) \tag{23}$$

$$P(x_1, \ldots, x_n) = \alpha \prod_{i=1}^{n} P_s(x_i|w=1) + (1-\alpha) \prod_{i=1}^{n} P_s(x_i|w=0) \tag{24}$$

The functions $P_S(x_i \mid w)$, $w=0, 1$, $i=1, \ldots, n$, can be viewed as $2 \times 2$ stochastic matrices relating each $x_i$ to the central hidden variable $w$ (see Fig. 6a), hence we name $P_S$ a *star-distribution* and call $P$ *star-decomposable*. Each matrix contains two independent parameters, $f_i$ and $g_i$, where

$$f_i = P_s(x_i = 1 \mid w=1)$$

$$g_i = P_s(x_i = 1 \mid w=0) \tag{25}$$

and the central variable $w$ is characterized by its prior probability $P_s(w=1) = \alpha$ (see Figure 6b).

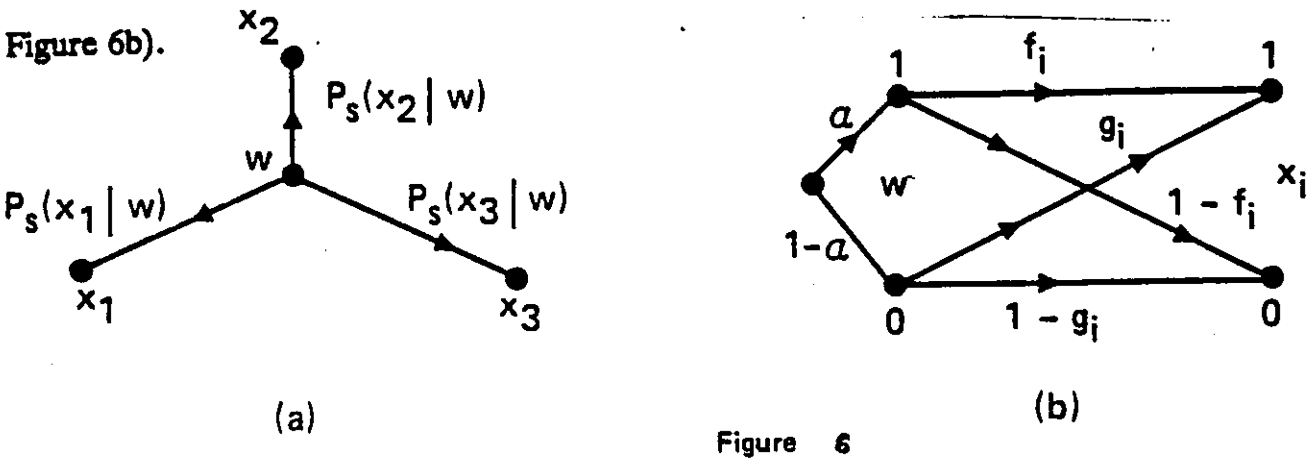


Figure 6

The advantages of having star-decomposable distributions are several. First, the product form of $P_s$ in (23) makes it extremely easy to compute the probability of any combination of variables. More importantly, it is also convenient for calculating the conditional probabilities $P(x_i|x_j)$, describing the impact of an observation $x_j$ on the probabilities of unobserved variables. The computation requires only two vector multiplications.

Unfortunately, when the number of variables exceeds 3, the conditions for star-decomposability become very stringent, and are not likely to be met in practice. Indeed, a star-decomposable distribution for $n$ variables has $2n+1$ independent parameters, while the specification of a general distribution requires $2^n-1$ parameters. Lazarfeld (1966) considered star-decomposable distributions where the hidden variable $w$ is permitted to range over $\lambda$ values, $\lambda>2$. Such an extension requires the solution of $\lambda n+\lambda-1$ non-linear equations to find the values of its $\lambda n+\lambda-1$ independent parameters. In this paper, we pursue a different approach, allowing a larger number of binary hidden variables, but insisting that they form a tree-like structure (see Figure 7), i.e., each triplet forms a star but the central variables may differ from triplet to triplet. Trees often portray meaningful conceptual hierarchies and, computationally, are almost as convenient as stars.

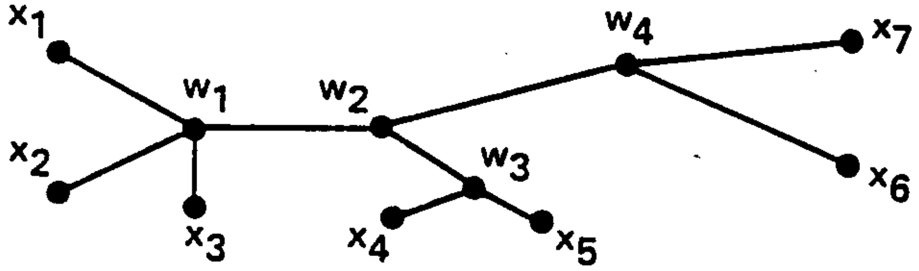


Figure 7

We shall say that a distribution $P(x_1,x_2, \cdots , x_n)$ is *tree-decomposable* if it is a marginal of a tree distribution

$$P_T(x_1,x_2, \cdots , x_n,w_1,w_2, \cdots , w_m) \qquad m \leq n-2$$

where $w_1,w_2, \cdots , w_m$ correspond to the internal nodes of an unrooted tree $T$ and $x_1, x_2, \cdots , x_n$ to its leaves. Given a tree structure and an assignment of variables to its nodes, the form of the corresponding distribution can be written by inspection. We first choose an arbitrary node as a root. This, in turn, defines a unique father $F(y_i)$ for each node $y_i \in \{x_1, \cdots , x_n, w_1, \cdots , w_m\}$ in the tree, except the chosen root, $y_1$. The joint distribution is simply given by the product form:

$$P_T(x_1 \cdots x_n, w_1 \cdots w_m) = P(y_1) \prod_{i=2}^{m+n} P[y_i| F(y_i)] \tag{26}$$

For example, if in Figure 7 we choose $w_2$ as the root we obtain:

$$P_T(x_1 \cdots x_7, w_1 \cdots w_4) =$$

$$P(x_7|w_4)\ P(x_6|w_4)\ P(x_5|w_3)\ P(x_4|w_3)\ P(x_3|w_1)\ P(x_2|w_1)\ P(x_1|w_1)\ P(w_1|w_2)\ P(w_3|w_2)\ P(w_4|w_2)\ P(w_2$$

Throughout this discussion we shall assume that each $w$ has at least three neighbors; otherwise it is superfluous. In other words, an internal node with two neighbors can simply be replaced by an equivalent direct link between the two. Note that any two leaves are conditionally independent given the value of any internal node on the path connecting them.

If we are given $P_T(x_1, \cdots x_n, w_1, \cdots w_m)$ then, clearly, we can obtain $P(x_1, \cdots x_n)$ by summing over the $w$'s. We now ask whether the inverse transformation is possible, i.e., given a tree-decomposable distribution $P(x_1, \cdots x_n)$, can we recover its underlying extension $P_T(x_1 \cdots x_n, w_1 \cdots w_m)$ ? We shall show that: (1) the tree distribution $P_T$ is unique, (2) it can be recovered from $P$ using $n\log n$ computations, and (3) the structure of $T$ is uniquely determined by the second order probabilities of $P$. The construction method depends on the analysis of star-decomposability for triplets which is presented next.

## 3.3 *STAR-DECOMPOSABLE TRIPLETS*

In order to test whether a given 3-variable distribution $P(x_1,x_2,x_3)$ is star-decomposable, we first solve eq.(24) and express the parameters $\alpha, f_i, g_i$ as a function of the parameters specifying $P$. This task was carried out by Lazarfeld (1966) in terms of the seven joint-occurrence probabilities

$$\begin{aligned} p_i &= P\ (x_i=1) \\ p_{ij} &= P\ (x_i=1,\ x_j=1) \\ p_{ijk} &= P\ (x_i=1,\ x_j=1,\ x_k=1) \end{aligned} \tag{27}$$

and led to the following solution:

Define the quantities

$$[ij] = p_{ij} - p_i p_j \tag{28}$$

$$S_i = \left( \frac{[ij][ik]}{[jk]} \right)^{\frac{1}{2}} \tag{29}$$

$$\mu_i = \frac{p_i p_{ijk} - p_{ij} p_{ik}}{[jk]} \tag{30}$$

$$K = \frac{S_i}{p_i} - \frac{p_i}{s_i} + \frac{\mu_i}{S_i p_i} \tag{31}$$

and let $t$ be the solution of

$$t^2 + Kt - 1 = 0 \tag{32}$$

The parameters $\alpha, f_i, g_i$ are given by:

$$\alpha = \frac{t^2}{1+t^2} \tag{33}$$

$$f_i = p_i + S_i \left(\frac{1-\alpha}{\alpha}\right)^{\frac{1}{2}} \tag{34}$$

$$g_i = p_i - S_i \left(\frac{\alpha}{1-\alpha}\right)^{\frac{1}{2}} \tag{35}$$

Moreover, the differences $f_i - g_i$ are independent of $p_{ijk}$,

$$f_i - g_i = S_i = \left(\frac{[ij][ik]}{[jk]}\right)^{\frac{1}{2}} \tag{36}$$

The conditions for star-decomposability are obtained by requiring that the preceding solutions satisfy:

(a) $S_i$ be real

(b) $0 \leq f_i \leq 1$

(c) $0 \leq g_i \leq 1$

Using the variances

$$\sigma_i = [p_i\,(1-p_i)]^{\frac{1}{2}} \tag{37}$$

and the correlation coefficients

$$\rho_{ij} = \frac{p_{ij} - p_i p_j}{\sigma_i \sigma_j} \tag{38}$$

requirement (a) is equivalent to the condition that all three correlation coefficients are non-negative. (If two of them are negative, we can rename two variables by their complements; the newly defined triplet will have all its pairs positively correlated.) We shall call triplets with this property *positively correlated*.

This, together with requirements (b) and (c), gives (see Appendix II):

**Theorem 1:** A necessary and sufficient condition for three dichotomous random variables to be star-decomposable is that they are positively correlated, and that the inequality:

$$\frac{p_{ik}p_{ij}}{p_i} \le p_{ijk} \le \frac{p_{ik}p_{ij}}{p_i} + \sigma_j \sigma_k \left(\rho_{jk} - \rho_{ij}\rho_{ik}\right) \tag{39}$$

is satisfied for all $i \in \{1, 2, 3\}$. When this condition is satisfied, the parameters of the star-decomposed distribution can be determined uniquely, up to a complementation of the hidden variable $w$, i.e., $w \to (1-w)$, $f_i \to g_i$, $\alpha \to (1-\alpha)$.

Obviously, in order to satisfy (39), the term $(\rho_{jk} - \rho_{ij}\rho_{ik})$ must be non-negative. This introduces a simple necessary condition for star-decomposability that may be used to quickly rule out many likely candidates.

**Corollary** -- A necessary condition for a distribution $P(x_1, x_2, x_3)$ to be star-decomposable is that all correlation coefficients obey the triangle inequality:

$$\rho_{jk} \geq \rho_{ji}\rho_{ik} \tag{40}$$

(40) is satisfied with equality if $w$ coincides with $x_i$; i.e., when $x_j$ and $x_k$ are independent given $x_i$. Thus, an intuitive interpretation of this corollary is that the correlation between any two variables must be stronger than that induced by their dependencies on the third variable; a mechanism accounting for direct dependencies must be present.

Having established the criterion for star-decomposability we may address a related problem: Suppose $P$ is not star-decomposable, can it be approximated by a star-decomposable distribution $\hat{P}$ that has the same second-order probabilities?

The preceding analysis contains the answer to this question. Note that the 3rd order statistics are represented only by the term $p_{ijk}$, and this term is confined by eq.(29) to a region whose boundaries are determined by 2nd- order parameters. Thus, if we insist on keeping all 2nd-order dependencies of $P$ in tact and are willing to choose $p_{ijk}$ so as to yield a star-decomposable distribution, we can only do so if the region circumscribed by (29) is non-empty. This leads to the statement:

**Theorem 2:** A necessary and sufficient condition for the 2nd order dependencies among the triplet $x_1, x_2, x_3$ to support a star-decomposable extension is that the six inequalities:

$$\frac{p_{ij}p_{ik}}{p_i} \leq x \leq \frac{p_{ij}p_{ik}}{p_i} + \sigma_j\sigma_k\,(\rho_{jk} - \rho_{ij}\rho_{ik}) \qquad i=1,2,3 \tag{41}$$

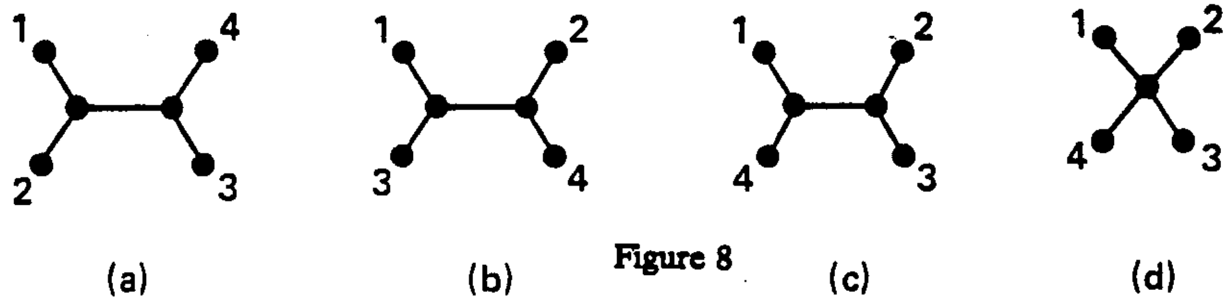
possess a solution for $x$.

## 3.4 *A TREE-RECONSTRUCTION PROCEDURE*

We are now ready to confront the central problem of this Section: Given a tree-decomposable distribution $P(x_1, \cdots, x_n)$, can we recover its underlying topology and the underlying tree-distribution $P_T(x_1, \cdots, x_n, w_1, \cdots, w_m)$?

The construction method is based on the observation that any three leaves in a tree have one and only one internal node that can be considered their *center*, i.e., it lies on all the paths connecting the leaves to each other. If one removes the center, the three leaves become disconnected from each other. This means that if $P$ is tree-decomposable then the joint distribution of any triplet of variables $x_i, x_j, x_k$ is star-decomposable, i.e., $P(x_i, x_j, x_k)$ uniquely determines the parameters $\alpha, f_i, g_i$ as in eq.(33), (34), and (35), where $\alpha$ is the marginal probability of the central variable. Moreover, if we compute the star decompositions of two triplets of leaves, both having the same central node $w$, the two distributions should have the same value for $\alpha = P_T(w=1)$. This provides us with a basic test for verifying whether two arbitrary triplets of leaves share a common center and a successive application of this test is sufficient for determining the structure of the entire tree.

Consider a 4-tuple $x_1$, $x_2$, $x_3$, $x_4$ of leaves in $T$. These leaves are interconnected through one of the four possible topologies shown in Figure 8. The topologies differ in



(a) (b) (c) (d)

Figure 8

the identity of the triplets which share a common center. For example, in the topology of Figure 8(a), the pair [(1,2,3), (1,2,4)] share a common center and so does the pair [(1,3,4), (2,3,4)]. In Figure 8(b), on the other hand, the sharing pairs are [(1,2,4), (2,4,3)] and [(1,3,4), (2,1,3)], and in Figure 8(d) all triplets share the same center. Thus, the basic test for center-sharing triplets enables us to decide the topology of any 4-tuple and, eventually, to configure the entire tree.

We start with any three variables $x_1$, $x_2$, and $x_3$, form their star decomposition, choose a fourth variable $x_4$, and ask to which leg of the star should $x_4$ be joined. We can answer this question easily by testing which pairs of triplets share centers, decide on the appropriate topology, and connect $x_4$ accordingly. Similarly, if we already have a tree structure $T_i$, with $i$ leaves, and wish to know where to join the $(i+1)^{th}$ leaf, we can choose any triplet of leaves from $T_i$ with central variable $w$, and test to which leg of $w$ should $x_{i+1}$ be joined. This, in turn, identifies a subtree $T_i'$ of $T_i$ that should receive $x_{i+1}$ and permits us to remove from further considerations the subtrees emanating from the unselected legs of $w$. Repeating this operation on the selected subtree $T_i'$ will eventually reduce it to a single branch, to which $x_{i+1}$ is joined.

It is possible to show (Tarsi and Pearl 1984) that by choosing, in each state, a central variable that splits the available tree into subtrees of roughly equal-size, the joining branch of $x_{i+1}$ can be identified in at most $\log_{\frac{k}{k-1}}(i)$ tests, where $k$ is the maximal degree of the tree $T_i$. This amounts to $0(n\log n)$ tests for constructing an entire tree of $n$ leaves.

So far we have shown that the structure of the tree $T$ can be uncovered uniquely. Next we show that the distribution $P_T$, likewise, is uniquely determined from $P$, i.e., that we can determine all the functions $P(x_i \mid w_j)$ and $P(w_j \mid w_k)$ in (4), for $i=1, \cdots n$ and $j, k=1, 2, \cdots m$. The functions $P(x_i \mid w_j)$ assigned to the peripheral branches of the tree are determined directly from the star decomposition of triplets involving adjacent leaves. In Figure 7, for example, the star decomposition of $P(x_1, x_2, x_5)$ yields $P(x_1 \mid w_1)$ and $P(x_2 \mid w_1)$. The conditional probabilities $P(w_i \mid w_k)$ assigned to interior branches are determined by solving matrix equations. For example, $P(x_1 \mid w_2)$ is obtained from the star decomposition of $(x_1, x_5, x_7)$, and it is related to $P(x_1 \mid w_1)$ via

$$P(x_1 \mid w_2) = \sum_{w_1} P(x_1 \mid w_1) P(w_1 \mid w_2)$$

This matrix equation has a solution for $P(w_1 \mid w_2)$ because $P(x_1 \mid w_1)$ must be nonsingular. It is only singular when $f_1 = g_1$, i.e., when $x_1$ is independent of $w_1$ and, therefore, independent of all other variables. Hence, we can determine the parameters of the branches next to the periphery, use those to determine more interior branches, and so on, until all the interior conditional probabilities $P(w_i \mid w_j)$ are determined.

Next, we shall show that the tree structure can be recovered without resorting to 3rd order probabilities; correlations among pairs of leaves suffice. This feature stems from the observation that when two triplets of a 4-tuple are star-decomposable with respect to the same central variable $w$ (e.g. 1,2,3 and 1,2,4 in Fig. 8(a)), then not only the values of $\alpha$ are the same but also the $f$ and $g$ parameters associated with the two common variables (e.g. 1 and 2 in Fig. 8(a)) must be the same. Whereas the value of $\alpha$

depends on a 3rd order probability, the difference $f_i - g_i$ depends only on 2nd order terms via eq.(36). Thus, requiring that $f_1 - g_1$ in Fig. 8(a) obtains the same value in the star decomposition of (1,2,3) as in that of (1,2,4), leads to the equation:

$$\frac{[12][13]}{[23]} = \frac{[12][14]}{[24]} \tag{42}$$

and, using (28), this yields

$$\rho_{13}\rho_{42} = \rho_{14}\rho_{32} \cdot \tag{43}$$

An identical equality will be obtained for each $f_i - g_i$, $i = 1,2,3,4$, relative to the topology of Figure 8(a). Similarly, the topology of Figure 8(b) dictates

$$\rho_{12}\rho_{43} = \rho_{14}\rho_{23} \tag{44}$$

and that of Figure 8(c):

$$\rho_{12}\rho_{34} = \rho_{13}\rho_{24} \tag{45}$$

Thus, we see that each of these three topologies is characterized by its own distinct equality, while the topology of Figure 8(d) by having all three equalities hold simultaneously. This provides the necessary 2nd-order criterion for deciding the topology of any 4-tuple tested; if the equality $\rho_{ij}\rho_{kl} = \rho_{ik}\rho_{jl}$ holds for some permutation of the indices, we decide on the topology ${}^{i}_{l}\!> \bullet\!-\!\bullet <\!{}^{j}_{k}$, if it holds for two such permutations, the entire 4-tuple is star decomposable. Note that the equality $\rho_{ij}\rho_{kl} = \rho_{ik}\rho_{jl}$ must hold for at least one permutation of the variables, or else the 4-tuple would not be tree-decomposable.

## 3.5 *CONCLUSIONS AND OPEN QUESTIONS*

This section provides an operational definition for entities called "hidden causes", which are not directly observable but facilitate the acquisition of effective causal models from empirical data. Hidden causes are viewed as dummy variables which, if held constant, induce probabilistic independence between sets of visible variables. It is shown that if all variables are bi-valued and if the activities of the visible variables are governed by a tree-decomposable probability distribution, then the topology of the tree can be uncovered uniquely from the observed correlations between pairs of variables. Moreover, the structuring algorithm requires only $n \log n$ steps.

The method introduced in this paper has two major shortcomings: It requires precise knowledge of the correlation coefficients and it only works when there exists an underlying model that is tree-structured. In practice, we often have only sample estimates of the correlation coefficients, and it is therefore unlikely that criteria based on equalities (as in eq.(43)) will ever be satisfied exactly. It is possible, of course, to relax these criteria and make topological decisions by seeking proximities rather than equalities. For example, instead of searching for an equality $\rho_{ij}\rho_{kl} = \rho_{ik}\rho_{jl}$, we can decide the 4-tuple topology on the basis of the permutation of indices that minimizes the difference $\rho_{ij}\rho_{kl} - \rho_{ik}\rho_{jl}$. Experiments show, however, that the structure which evolves by such a method is very sensitive to inaccuracies in the estimates $\rho_{ij}$, because no mechanism is provided to retract erroneous decisions made in the early stages of the structuring process. Ideally, the topological membership of the $(i+1)^{th}$ leaf should be decided not merely by its relations to a single triplet of leaves chosen to represent an internal node *w*, but also by its relations to all previously structured triplets which share *w* as a center. This, of course, will substantially increase the complexity of the algorithm.

Similar difficulties plague the task of finding the best tree-structured *approximation* to a distribution which is not tree-decomposable. Even though we argued that natural data which lend themselves to causal modeling should be representable as tree-decomposable distributions, these distributions may contain internal nodes with more than two values. The task of determining the parameters associated with such nodes is much more complicated and, in addition, rarely yields unique solutions. Unique solutions, as shown in section 3.4, are essential for building large structures from smaller ones. We leave open the question of explaining how approximate causal modeling, an activity which humans seem to perform with relative ease, can be embodied in computational procedures that are both sound and efficient.

# APPENDIX I

## THE VALIDITY OF THE UPDATING SCHEME FOR TREES

From the fact the $\lambda$ is only influenced by changes propagating from the bottom and $\underline{q}$ only by changes from the top, it is clear that the tree will reach equilibrium after a finite number of updating steps. It remains to show that, at equilibrium, the updated parameters $P(V_i)$, in every node $V$, correspond to the correct probabilities $P(V_i|D^u(V), D_d(V))$ or (see eq.(3)), that the equilibrium values of $\lambda(V_i)$ and $q(V_i)$ actually equal the probabilities $P(D_d(V)|V_i)$ and $P(V_i|D^u(V))$. This can be shown by induction: bottom-up for $\lambda$ and then top-down for $\underline{q}$.

*Validity of* $\lambda$: $\lambda$ is certainly valid for leaf nodes, as was explained above in setting the boundary conditions. Assuming that the $\lambda$'s are valid at all children of node $B$, the validity of $\lambda(B)$ computed through steps (11) and (12) follows directly from the conditional independence of the data beneath $B$'s children (10).

*Validity of* $\underline{q}$: if all the $\lambda$'s are valid, then $\underline{P}$ is valid for the root node. Assuming now that $\underline{P}(A)$ is valid, let us examine the validity of $\underline{q}(B)$, where $B$ is any child of $A$. By definition (eq.(6)), $\underline{q}(B)$ should satisfy:

$$q(B_i) = P(B_i|D^u(B)) = \sum_j P(B_i|A_j)P(A_j|D^u(A), D_d(S))$$

where $S$ denotes the set of $B$'s siblings. The second factor in the summation differs from $P(A_j) = P(A_j|D^u(A), D_d(A))$ in that the latter has also incorporated $B$'s message $(\underline{r}')_j$ in the formation of $\lambda(A_j)$ (eq.12). When we divide $P(A_j)$ by $(\underline{r}')_j$, as prescribed in (13), the correct probability ensues.

APPENDIX II

## CONDITIONS FOR STAR-DECOMPOSABILITY

Let

$$p_i = P\ (x_i=1)$$

$$p_{ij} = P\ (x_i=1,\ x_j=1) \qquad \text{(II-1)}$$

$$p_{ijk} = P\ (x_i=1,\ x_j=1,\ x_k=1)$$

The seven joint-occurrence probabilities, $p_1, p_2, p_3, p_{12}, p_{13}, p_{23}, p_{123}$, uniquely define the seven parameters necessary for specifying $P(x_1, x_2, x_3)$, for example:

$$P(x_1 = 1, x_2 = 1, x_3 = 0) = p_{12} - p_{123}$$

$$P(x_1 = 1, x_2 = 0) = p_1 - p_{12} \qquad \text{etc.}$$

and will be used in the following analysis.

Assuming $P$ is star-decomposable (eqs.22 and 23), we can express the joint-occurrence probabilities in terms of $\alpha, f_i, g_i$ and obtain seven equations for these seven parameters.

$$p_i = \alpha f_i + (1-\alpha)\ g_i \qquad \text{(II-2)}$$

$$p_{ij} = \alpha f_i f_j + (1-\alpha)\ g_i g_j \qquad \text{(II-3)}$$

$$p_{ijk} = \alpha f_i f_j f_k + (1-\alpha)\ g_i g_j g_k$$

(II-4)

These equations can be manipulated to yield product forms on the right-hand sides:

$$p_{ij} - p_i p_j = \alpha(1-\alpha)(f_i - g_i)(f_j - g_j) \tag{II-5}$$

$$p_i p_{ijk} - p_{ij} p_{ik} = \alpha(1-\alpha)\, f_i g_i\, (f_j - g_j)(f_k - f_k) \tag{II-6}$$

Eq.(II-5) comprises three equations which can be solved for the differences $f_i - g_{i,}$ $i=1, 2, 3$, giving

$$f_i - g_i = S_i = \pm \left( \frac{[ij][ik]}{[jk]} \right)^{\frac{1}{2}} \tag{II-7}$$

where the bracket $[ij]$ stands for the determinant

$$[ij] = p_{ij} - p_i p_j \tag{II-8}$$

These, together with (II-2), determine $f_i$ and $g_i$ in terms of $S_i$ and $\alpha$ (still unknown):

$$f_i = p_i + S_i \left( \frac{1-\alpha}{\alpha} \right)^{\frac{1}{2}} \tag{II-9}$$

$$g_i = p_i - S_i \left( \frac{\alpha}{1-\alpha} \right)^{\frac{1}{2}} \tag{II-10}$$

To determine $\alpha$, we invoke eq.(II-6) and obtain

$$\left(\frac{\alpha}{1-\alpha}\right)^{\frac{1}{2}} = t \qquad (\text{or}, \alpha = \frac{t^2}{1+t^2}) \tag{II-11}$$

where $t$ is the solution to

$$t^2 + Kt - 1 = 0 \tag{II-12}$$

and $K$ is defined by:

$$K = \frac{S_i}{p_i} - \frac{p_i}{s_i} + \frac{\mu_i}{S_i p_i} \tag{II-13}$$

$$\mu_i = \frac{[jk,\, i]}{[jk]} = \frac{p_i p_{ijk} - p_{ij} p_{ik}}{[jk]} \tag{II-14}$$

It can be easily verified that $K$ (and, therefore, $\alpha$) obtains the same value regardless of which index $i$ provides the parameters in (II-13).

From eq.(II-13) we see that the parameters $S_i$ and $\mu_i$ of $P$ govern the solutions of (II-12) which, in turn, determine whether $P$ is star-decomposable via the resulting values of $\alpha$, $f_i$, $g_i$. These conditions are obtained by requiring that:

(a) $S_i$ be real

(b) $0 \le f_i \le 1$

(c) $0 \le g_i \le 1$

Requirement (a) implies that, of the three brackets in (II-7), either all three are non-negative or exactly two are negative. These brackets are directly related to the correlation coefficient, via:

$$\rho_{ij} = [ij]\,[p_i\,(1-p_i)]^{-\frac{1}{2}}[p_j\,(1-p_j)]^{-\frac{1}{2}} = \frac{[ij]}{\sigma_i\sigma_j} \qquad \text{(II-15)}$$

and so, requirement (a) is equivalent to the condition that all three correlation coefficients are non-negative. If two of them are negative, we can rename two variables by their complements; the newly defined triplet will have all its pairs positively correlated.

Now attend to requirement (b). eq.(II-9) shows that $f_i$ can be negative only if $S_i$ is negative, i.e, if $S_i$ is identified with the negative square root in (II-7). However, the choice of negative $S_i$ yields a solution $(f_i', g_i', \alpha')$ which is symmetrical to that stemming from a positive $S_i$ $(f_i, g_i, \alpha)$, with $f_i' = g_i$, $g_i' = f_i$, $\alpha' = 1-\alpha$. Thus, $S_i$ and $f_i$ can be assumed non-negative, and it remains to examine the condition $f_i \leq 1$ or, equivalently, $t \geq \frac{S_i}{1-p_i}$ (see (II-9) and (II-11)). Imposing this condition in (II-12) translates to:

$$p_{ijk} \leq \frac{p_{ij}p_{ik}}{p_i} + \sigma_k\sigma_j[\rho_{jk} - \rho_{ij}\rho_{ik}] \qquad \text{(II-16)}$$

Similarly, inserting requirement (c), $g_i \geq 0$, in eq.(II-12) yields the inequality:

$$\frac{p_{ik}p_{ij}}{p_i} \leq p_{ijk} \qquad \text{(II-17)}$$

which, together with (II-16), lead to Theorem 1, Section 3.3.

## ACKNOWLEDGEMENT

I thank Norm Dalkey, Thomas Ferguson, Jin Kim, Michael Tarsi, and Eli Gafni for sharing their time, knowledge, and ideas.

N. Dalkey has called my attention to the work of Lazarfeld [Lazarfeld 1966], and has provided a continuous stream of valuable advice. Thomas Ferguson made helpful comments on Section 3. Jin Kim is responsible for deriving the propagation equations of section 2.3 [Kim & Pearl 1983], and for using the propagation scheme is his decision support system CONVINCE [Kim 1983]. Michael Tarsi has helped develop the tree construction algorithm of Section 3.4, and has proven its optimality [Tarsi & Pearl 1984]. Eli Gafni has spent many hours discussing the relations between Markov fields, stochastic relaxation, and belief propagation.

The results of Section 3 will be presented at the International Joint Conference on Artificial Intelligence, University of California, Los Angeles, August 19-23, 1985.

# REFERENCES

Anderson, John R., (1983), "The Architecture of Cognition", Harvard University Press, Cambridge, MA.

Chow, C.K. and Liu, C.N., (1968), Approximating Discrete Probability Distributions with Dependence Trees, *IEEE Trans. Inf. Theory*, IT-14, 462-467.

Geman, S. and Geman, D., (1984), Stochastic Relaxations, Gibbs Distributions, and the Bayesian Restoration of Images, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-6, No. 6, 721-742, November.

Hinton, G.E., Sejnowski, T.J., and Ackley, D.H., (1984), Boltzman Machines: Constraint Satisfaction Networks that Learn, Technical Report CMU-CS-84-119, Department of Computer Science, Carnegie-Mellon University.

Kim, J., (1983), "CONVINCE: A CONversational INference Consolidation Engine." Ph.D. Dissertation, University of California, Los Angeles.

Kim, J. and Pearl, J., (1983), A Computational Model for Combined Causal and Diagnostic Reasoning in Inference Systems, *Proceedings of IJCAI-83*, 190-193.

Lazarfeld, ., (1966), Latent Structure Analysis, *in* Stouffer, Guttman, Slachman, Lazarfeld, Star, and Claussen (eds.), "Measurement and Prediction", Wiley, New York.

Lesser, V.R. and Erman, L.D., (1977), A Retrospective View of HEARSAY II Architecture, *Proc. 5th Int. Joint Conf. AI*, Cambridge, MA, 790-800.

Pearl, J., (1982), Distributed Bayesian Processing for Belief Maintenance in Hierarchical Inference Systems, UCLA-ENG-CSL-8211, UC Los Angeles, January.

Rumelhart, D.E., (1976), Toward an Interactive Model of Reading, *Center for Human Info. Proc. CHIP-56*, UC San Diego, La Jolla, CA.

Shastri, L. and Feldman, J.A., (1984), Semantic Networks and Neural Nets, TR-131, Computer Science Dept., The University of Rochester, Rochester, NY, June.

Simon, H.A., (1959), Spurious Correlations: A Causal Interpretation, *Journal of American Statistical Association*, 49, 469-492.

Suppes, P., (1970), "A Probabilistic Theory of Causality", North-Holland, Amsterdam.

Tarsi, M. and Pearl, J., (1984), Algorithmic Reconstruction of Trees, CSD-840061, Cognitive Systems Laboratory, UC Los Angeles, Los Angeles, CA.

## Figure Captions

Figure 1 - A typical Bayes network representing the distribution $P(x_1 \cdots x_6) = P(x_6|x_5)\, P(x_5|x_2,x_3)\, P(x_4|x_1,x_2)\, P(x_3|x_1)\, P(x_2|x_1)\, P(x_1)$.

Figure 2 - A segment of a tree illustrating data partitioning.

Figure 3 - The internal structure of a single processor performing belief updating for variable B.

Figure 4 - The impact of new data propagates through a tree by a message-passing process.

Figure 5 - Fragments of a singly connected network with multiple parents, illustrating data partitioning and belief parameters.

Figure 6 - (a) Three random variables, $x_1$, $x_2$, $x_3$ connected to a central variable $w$ by a star network.
(b) Illustrating the three parameters, $\alpha$, $f_i$, $g_i$, associated with each link.

Figure 7 - A tree containing four dummy variables and seven visible variables.

Figure 8 - The four possible topologies by which four leaves can be related.