Computer Science Department Technical Report
University of California
Los Angeles, CA 90024-1596

LEARNING SIMPLE CAUSAL STRUCTURES

Dan Geiger
Azaria Paz
Judea Pearl

July 1991
CSD-910022

. To appear in International Journal of Intelligent Systems.

TECHNICAL REPORT
R-152
October 1990

Learning Simple Causal Structures

Dan Geiger
dgeiger@nrtc.northrop.com
Northrop Research and
Technology Center
One Research Park
Palos Verdes, CA 90274

Azaria Paz
paz@techsel.bitnet*
Cognitive Systems Lab.
Computer Science Department
University of California
Los Angeles, CA 90024

Judea Pearl
judea@cs.ucla.edu
Cognitive Systems Lab.
Computer Science Department
University of California
Los Angeles, CA 90024

*Current address: Technion, Computer Science Department, Haifa, Israel 32000

Abstract

Humans use knowledge of causation to derive dependencies among events of interest. The converse task, that of inferring causal relationships from patterns of dependencies, is far less understood. This paper establishes conditions under which the directionality of some dependencies is uniquely dictated by probabilistic information — an essential prerequisite for attributing a causal interpretation to these dependencies. An efficient algorithm is developed that, given data generated by an undisclosed simple causal schema, recovers the structure of that schema, as well as the directionality of all links that are uniquely orientable. A simple schema is represented by a directed acyclic graph (dag) where every pair of nodes with a common direct child have no common ancestor nor is one an ancestor of the other. Trees, singly-connected dags and directed bi-partite graphs are examples of simple dags. Conditions ensuring the correctness of this recovery algorithm are provided.

1 Introduction

The study of causation, because of its pervasive usage in human communication and problem solving, is central to the understanding of human reasoning. Any reasoning task that deals with changing environments relies on the distinction between cause and effect. For example, a central task in applications such as diagnosis, qualitative physics, plan recognition and language understanding, is that of abduction, i.e., finding a satisfactory explanation for a given set of observations, where explanation builds on the notion of causation.

Most AI works have given the term "cause" a procedural semantics, attempting to match the way people use it in inference tasks, but were not concerned with what makes people believe that "a causes b", as opposed to, say, "b causes a" or "c causes both a and b." [1,14]. An empirical semantics for causation is important for several reasons. First, by formulating the empirical components of causation we gain a better understanding of the meaning conveyed by causal utterances, such as "a explains b", "a suggests b", "a tends to cause b", and "a actually caused b". These utterances are the basic building blocks from which knowledge bases are assembled. Second, any autonomous learning system attempting to build a causal model of its environment cannot rely exclusively on procedural semantics but must be able to translate direct observations to cause-effect relationships.

Formal definitions of causation rely heavily on temporal information. Suppes [18], for example, introduces a probabilistic definition of causation assuming that temporal ordering of all events is known. Shoham makes an identical assumption [14]. In this paper we propose a non-temporal semantics, one that determines the directionality of causal influence without resorting to temporal information, in the spirit of [15] and [6]. Such semantics should be applicable, therefore, to the organization of concurrent events or events whose chronological precedence cannot be determined empirically. Such situations are common in economics, medicine, and in the behavioral sciences where we say, for example, that old age explains a certain disability, not the other way around, even though the two occur together (in many cases it is the disability that precedes old age).

Another feature of our formulation is the appeal to probabilistic dependence, as opposed to functional or deterministic dependence. This is motivated by the observation that most causal connections found in natural discourse, for example "reckless driving causes accidents," are probabilistic in nature [17]. Given that statistical analysis cannot distinguish causation from covariation, we must still identify the asymmetries that prompt people to perceive causal structures in empirical data, and we must find a computational model for such perception.

Our attack on the problem is as follows; first, we pretend that Nature possesses "true" cause-effect relationships and that these relationships form a *causal schema*, namely, a directed acyclic graph (dag) where each node represents a variable in the domain and the parents of that node correspond to its direct causes, as designated by Nature. Next we assume that Nature annotates the causal schema by assigning probabilistic parameters to its links, such that, direct causes of each variable render that variable conditionally independent of all other variables except its consequences. Nature permits scientists to observe the resulting distribution and to ask questions about its properties, but hides the underlying causal schema.

Definition A *causal schema* is a directed acyclic graph where each link

$a \rightarrow b$ corresponds to a direct causal influence of a on b. A joint probability distribution is said to be *generated* by D if P can be factored as follows:

$$P(u_1, ..., u_n) = \prod_{u_i \in U} P(u_i \mid \pi(u_i))$$

where $\pi(u_i)$ are the variables corresponding to the parents of node u_i. And P cannot be factored this way if any link of D is deleted.[1]

We investigate the feasibility of recovering the schema's topology uniquely and efficiently from features of the joint distribution.

This formulation contains several simplifications of the actual task of scientific discovery. It assumes, for example, that scientists obtain the distribution, rather than events sampled from that distribution. This assumption is justified when a large sample is available, sufficient to reveal all the dependencies embedded in the distribution. It also assumes that the scientist can observe all relevant variables. The possibility of having relevant variables which can not be measured, prevents us from distinguishing between *spurious correlations* [15] and genuine causes, a distinction that is impossible within the confines of a closed world assumption[2]. However, solving this simplified problem is an essential component in any attempt to deduce causal relationships from measurements, and this is the main concern of this paper.

Clearly, if Nature wishes to confuse the scientist it could choose a distribution that hides some of the causal links. For example, if Nature makes two causal paths have precisely equal strengths and of opposite signs the scientist would have no way of distinguishing this incidental cancellation from a permanent absence of a causal connection in the underlying schema. In general, to allow for such cancellations, the scientist would never be able to rule out the possibility that the underlying schema is a complete graph; a structure that with a clever choice of parameters can mimic the behavior of any other schema. We therefore need to impose some restrictions on the complexity of the structures under consideration. In this paper we limit the complexity by assuming that no accidental cancellations take place and

[1] This factorization is equivalent to the requirement that given its parents each variable be independent of all other variables except its consequences. And that no proper subset of its parents has this property.

[2] See [20] for a way of relaxing this assumption

that the underlying structure is a *simple* dag. A simple dag is one in which every pair of nodes with a common direct child have no common ancestor nor is one an ancestor of the other. Such dags are known to permit fast updating procedures, and are therefore worthy of pursuit [7].

The theory developed below addresses the following problem: We observe a probability distribution P and ask whether P could have been generated from a simple causal schema D, what properties of P allow the efficient recovery of one such schema, and under what conditions D is unique. The recovery algorithm developed here considerably generalizes the method of [3] and the method of Rebane and Pearl, as it does not assume the distribution to be *dag-isomorph* [12, Chapter 8] nor that the network be *singly-connected.* The generalization implies, for example, that the assumption of a multivariate normal distribution is sufficient for a complete recovery of simple causal schemas. The algorithm works in many other cases as well.

For example, consider the simple causal schema showing the relationships between diseases and findings (figure 1). Each node d_i represents a disease and is associated with the marginal distribution $P(d_i)$, and each node f_j represents a finding and is associated with the conditional probability $p(f_j|d_{j_1},...,d_{j_k})$ where $d_{j_1},...,d_{j_k}$ are diseases variables that correspond to the direct parents of node f_i. The product of all these conditional distributions,

$$P(d_1...d_n, f_1...f_m) = \prod_1^n P(d_i) \prod_1^k p(f_j|d_{j_1},...,d_{j_k}),$$

forms a probability distribution that is generated by the causal schema of figure 1. Note that the directionality of some links can never be recovered from this distribution because some directionalities do not constrain it. For example, identical probability distributions are generated by an alternative causal schema in which the link between d_1 and f_1 is reversed. Even the remaining links might not always be orientable by merely observing P. However, under the following assumptions all the remaining links can be oriented uniquely and efficiently without searching through the enormous space of alternatives. We assume:

1. Every combination of diseases and symptoms has some positive probability of occurring (i.e., exceptions are always present).

2. Each link represents a genuine causal influence of a disease on a symptom, i.e., $P(s|d) \neq P(s)$ where d is a parent of s.

3. Two symptoms caused by a common disease d are dependent unless it is known whether the disease has or hasn't occurred (i.e., no accidental cancellations).

4. All diseases are mutually independent.

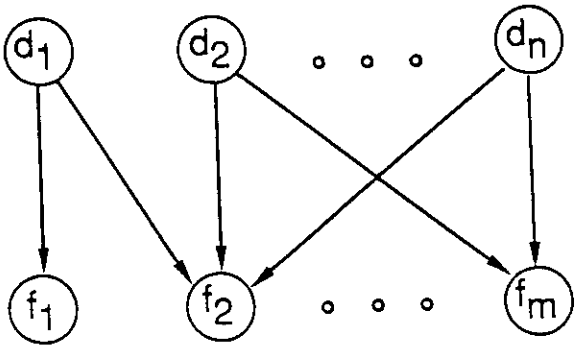

Figure 1: A causal schema representing symptoms and diseases

These conditions guarantee that the directionality of all links whose orientation constrains the distribution can be determined uniquely from a probability distribution. This transition from symmetric probabilistic association to unique directionality is an essential prerequisite for attributing a causal interpretation to these links.

Below we formalize requirements (1) through (4) and provide an algorithm for the recovery of the dag of figure 1 as well as any other simple causal schema. We first introduce the concept of a Bayesian network, then we examine the relationship between causal schemas and Bayesian networks and then we provide an algorithm that finds a simple Bayesian network that *well-represents* a given distribution if such a representation exists. Finally, we show that the algorithm is applicable to the recovery of simple dags when the distribution is Gaussian.

2 Probabilistic Dependence: Background and Definitions

Our model of an empirical environment consists of a finite set of variables U and a distribution P over these variables. Variables in a medical domain, for example, represent entities such as "cold", "hay fever", and "sneeze". An empirical environment can be *represented* graphically by an acyclic directed graph, called a *Bayesian network* of P, as follows: We select a linear order on all variables in U. Each variable is represented by a node. The parents of a node v correspond to a minimal set of variables that make v conditionally independent of all lesser variables in the selected order. Different orderings may produce different graphs. For example, one representation of the three variables above is the chain *cold* $\rightarrow$ *sneeze* $\leftarrow$ *hay fever* which is produced by the order *cold*, *hay fever*, then *sneeze* (assuming *cold* and *hay fever* are independent causes of sneezing). Another ordering of these variables: *sneeze*, *hay fever*, then *cold* would yield the complete dag of figure 2.b, because no single variable renders *cold* independent of the other lesser variable.

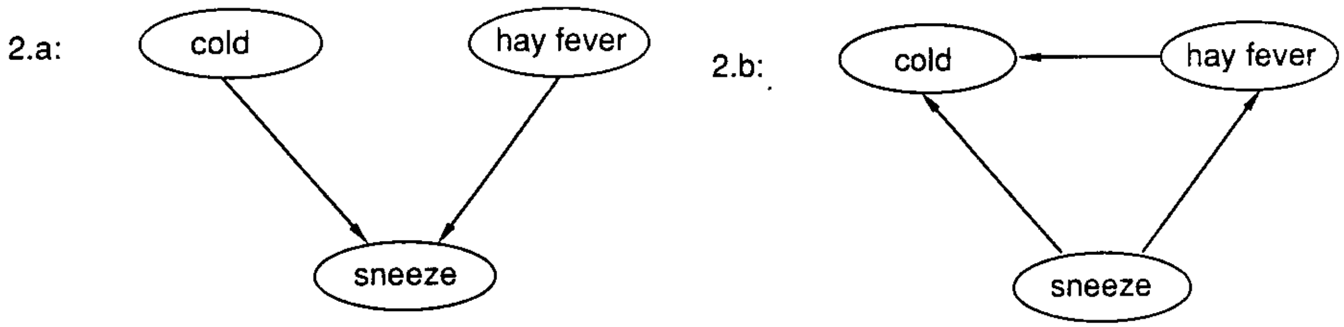

Figure 2: Two Bayesian networks representing cold, hay fever, and sneeze.

Note that the directionality of some links differs between the two alternative Bayesian network representations. In the first graph directionality matches our perception of cause-effect relationships while in the second it does not, being merely a spurious by-product of the ordering chosen for the construction. We shall see that, despite the arbitrariness in choosing the construction ordering of a Bayesian network, our algorithm will declare the network of figure 2.a as the preferred one. The basis for differentiating among alternative representations are the dependence relationships encoded in the different dags, which point to the existence of the unique simple dag representation of figure 2.a. But first, we must establish some notational conventions.

Throughout this paper we consider a finite set of variables $U = \{u_1, ..., u_n\}$ each having a finite *domain of values* $d(u_i)$, and a probability distribution P over a set of variables U having the Cartesian product $\bigtimes_{u_i \in U} d(u_i)$ as its sample space. We say that P is *strictly-positive* if every combination of values assigned to the variables has a non-zero probability. We use lowercase letters possibly subscripted (e.g a, b, x or u_i) to denote variables, and uppercase letters (e.g. X, Y, or Z) to denote sets of variables. A bold lowercase or uppercase letter refers to a value of a variable or a set of variables, respectively (e.g., $\boldsymbol{a}$ is a value of a). A value $\boldsymbol{X}$ of a set of variables X is a member in the Cartesian product $\bigtimes_{x \in X} d(x)$ is the set of values of x. The notation $X = \boldsymbol{X}$ stands for $x_1 = \boldsymbol{x}_1, ..., x_n = \boldsymbol{x}_n$ where $X = \{x_1, ..., x_n\}$ and $\boldsymbol{x}_i$ is a value of x_i. The notation $P(X \mid Y)$ stands for $P(X = \boldsymbol{X} | Y = \boldsymbol{Y})$ for all values $\boldsymbol{X}$ of X and $\boldsymbol{Y}$ of Y. When Y is the empty set, $P(X \mid Y)$ is just $P(X)$.

Definition Let $U = \{u_1, ..., u_n\}$ be a finite set of variables with $d(u_i)$ and P as above. If X, Y, and Z are three disjoint subsets of U, then X is *conditionally independent* of Y given Z, denoted $I(X, Z, Y)$, if for every three sets of values $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ of X, Y, and Z, respectively, the following equation holds:

$$P(X = \boldsymbol{X} \mid Z = \boldsymbol{Z}, Y = \boldsymbol{Y}) \; = P(X = \boldsymbol{X} \mid Z = \boldsymbol{Z})$$

whenever $P(Z = \boldsymbol{Z}, Y = \boldsymbol{Y}) > 0$. A statement $I(X, Z, Y)$ is called an *independency* and its negation is called a *dependency.*

Definition A directed acyclic graph D is said to be a *Bayesian network representing* a probability distribution P over a finite set of variables U if

D is constructed from P by the following steps: assign an arbitrary total order $d : u_1, u_2, , ..., u_n$ to the elements of U and designate a distinct node in D for each variable in U. For each element u_i in U, identify a *minimal* set of predecessors $\pi(u_i)$ such that $I(u_i, \pi(u_i), \{u_1, ..., u_{i-1}\} \setminus \pi(u_i))$ holds in P. Assign a direct link from every node corresponding to an element in $\pi(u_i)$ to the node corresponding to u_i.

Equivalently, due to the definition of conditional independence, a directed acyclic graph D is a Bayesian network representing a probability distribution P if and only if P can be factorized into,

$$P(u_1, ..., u_n) = \prod_{u_i \in U} P(u_i \mid \pi(u_i)),$$

and no link of D can be deleted without distroying this factorization.

The definition of Bayesian networks is based on the notion of conditional independence and is often claimed to be unrelated to the notion of causation (e.g., see the discussion by Herman Rubin in [9]). We show below that if a Bayesian network is regarded as a representation of causal relationships, then the patterns of dependencies these relationships impose must coincide with the consequences of the definition of Bayesian networks. Conversely, we show that under the assumptions defined in this paper, the causal schema can be inferred from the conditional independencies embedded in the Bayesian network.

The following definition of *graphical dependence* and *graphical independence*[3] captures our intuition about the type of dependencies that are accompanied by cause-effect relationships. And the theorem that follows shows that precisely these dependencies are captured by the definition of Bayesian networks. Some preliminary definitions are needed.

Definition A *skeleton* of a dag D is the undirected graph formed from D by ignoring directionality of the links. A *trail* in D is a sequence of directed links that corresponds to an undirected path in the skeleton of D. *A head-to-head connection* in a dag is a trail t consisting of two links of the form $a \rightarrow b \leftarrow c$, and node b is called *a head-to-head node with respect to t*. A trail t is *active by Z* if (1) every head-to-head node wrt t either is or has a descendent in Z and (2) every other node along t is outside Z. Otherwise, the trail is said to be *blocked by Z*.

[3] Called d-separation in [12].

Definition [12] Let D be a Bayesian network and X, Y, and Z be three disjoint sets of nodes. Then, X and Y are said to be *graphically independent* given Z if there exists no active trail by Z between a node in X and a node in Y. Otherwise, X and Y are *graphically dependent* given Z.

For example, the propositions "It is raining" (r), "the pavement is wet" (w) and "John slipped on the pavement" (s) can be represented by a three node chain, from r through w to s; it indicates that rain and wet pavement could cause slipping, yet wet pavement is designated as the *direct cause*; rain could cause someone to slip if it wets the pavement, but not if the pavement is covered. Knowing the condition of the pavement renders "slipping" and "raining" independent, and this is captures by our definition rendering node r and s *graphically independent* by node w. Furthermore, if we assume that "broken pipe" (b) is another direct cause for wet pavement, as in figure 3, then an induced dependency exists between the two events that may cause the pavement to get wet: "rain" and "broken pipe". Although they appear connected in figure 3, these variables are originally independent and become dependent once we learn that the pavement is wet or that someone broke his leg. An increase in our belief in either cause would decrease our belief in the other as it would "explain away" the observation. Indeed, by our definition, "rain" and "broken pipe" are *graphically independent* given nothing is known, but they are *graphically dependent* once we know the pavement is wet.

Theorem 1 [19] *Let D be a Bayesian network representing a probability distribution P over a finite set of variables U. If X and Y are graphically independent given Z in D, then X and Y are probabilistically independent given Z in P.*

The above theorem provides a graphical criterion for identifying independence in a probability distribution that is represented by a Bayesian network. Furthermore, Geiger and Pearl have shown that this criterion cannot be strengthen in the sense that no additional independence assertions can be identified in P unless numeric parameters quantifying the network are examined [4,5].

Clearly, one may construct many Bayesian networks from a given probability distribution and the task is to find among them the network that

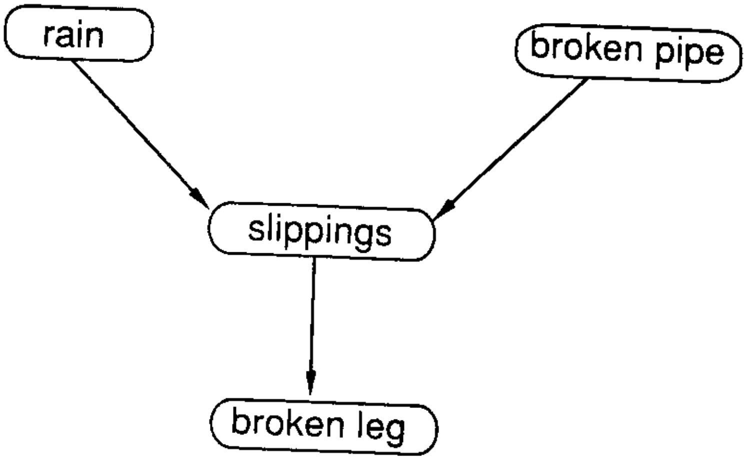

Figure 3: A Bayesian network representing reasons for slipping

corresponds to the causal schemata used by Nature to generate the observed distribution. For this aim we assume that no accidental cancellations occur. For example, in a network of the form $a \leftarrow c \rightarrow b$ one expects that changes in a reflect changes in c which project changes for b, hence making a and b dependent. The next definition formalizes this requirement.

Definition A *trek*[4] is a trail containing no head-to head connections (i.e., it has the form $a \leftarrow \ldots \leftarrow c \rightarrow \ldots \rightarrow b$). A Bayesian network representing a probability distribution P is said to **represent** P **well** if whenever two nodes a and b are connected with a trek then a and b are marginally dependent, i.e., $I(a, \emptyset, b)$ does not hold in P. Equivalently, we will say that P is *well-represented* by D.

We will concentrate on simple dags:

Definition A dag is *simple* if every pair of nodes with a common direct child have no common ancestor nor is one an ancestor of the other.

We show below that under the four assumptions of absence of unmeasurable variables, no accidental cancellations of causal influences, a strictly-positive distribution P, and the existence of a simple Bayesian network that

[4] Terminology of [6]

represents P well, the recovery of a Bayesian network from P is unique (up to isomorphism). Therefore, we can conclude that the orientations recovered coincide with the causal schemata that generated P.

3 The Main Result

The algorithm below determines whether a given probability distribution P can be well-represented by a simple Bayesian network and it finds such a network if it exists. The algorithm assumes that P is strictly-positive. This assumption is justified whenever categorical relationships can be excluded from analysis (as often happens, for example, in medical domains).

The Recovery Algorithm

Input: A strictly-positive probability distribution P over a finite set of variables U.
Output: A simple Bayesian network that represents P well if such exists, or acknowledgment that no such network exists.

1. Start with a complete undirected graph.

2. Remove every edge $a - b$ for which $I(a, U \setminus \{a, b\}, b)$ holds in P.

3. Remove every edge $a - b$ for which $I(a, \emptyset, b)$ holds in P.

4. Orient every pair of edges $a - b$ and $b - c$ towards b whenever $a - b - c$ is in the graph and $I(a, \emptyset, c)$ holds in P.

5. Orient the remaining links without introducing new head-to-head connections and such that the resulting dag is simple. If the resulting orientation is not feasible then "FAIL". Performing the following steps does this:

   (a) Label every head-to-head node with 1 and all other nodes with 0.

   (b) While there are undirected edges with end-labels $(0, 1)$ and there are no undirected edges with end-labels $(1, 1)$, direct the $(0, 1)$ edges from 1 to 0 and relabel the corresponding 0 node to 1.

(c) If an undirected edge with end-labels $(1, 1)$ appears, then "FAIL".

(d) Remove (temporarily) from the graph all directed edges. If the remaining graph is not a forest (i.e., a set of trees), then "FAIL".

(e) Orient every tree in the resulting forest without introducing head-to-head connections (e.g., for each tree in the forest select an arbitrary node and make the tree be a directed tree rooted at that node). Restore the temporarily-removed directed edges of step *d*.

(f) If the resulting graph is not a simple dag, then "FAIL".

6. If the resulting simple dag does not represent P well then "FAIL". Otherwise, output the resulting network.

The following sequence of claims establishes the correctness of the algorithm and the uniqueness of the recovered network. Proof details are given in the appendix.

Theorem 2 *Let P be a strictly-positive distribution and let D be a simple Bayesian network that represents P. Then, for every link $a - b$ in D, $I(a, U \setminus \{a, b\}, b)$ does not hold in P (and therefore is not removed in step 2).*

Theorem (2) guarantees that step 2 of the algorithm does not remove links that are needed for the construction of a simple Bayesian network representation of P.

Theorem 3 *Let P be a strictly-positive distribution. If P can be well-represented by a simple Bayesian network D, then the* skeleton*of D is equal to the graph constructed in step 3.*

Theorem 3 shows that step 3 of the algorithm identifies the skeleton of a simple Bayesian network that represents P well, if such exists. Thus, if P can be well-represented by a simple Bayesian network then it must be one of the orientations of the undirected graph produced by step 3. Hence by checking all possible orientations of this graph, one can decide whether a strictly-positive distribution can be well-represented by a simple Bayesian network. Notably, as a corollary, we obtain that all such representations have the same skeleton.

The next two theorems justify an efficient way of establishing the orientations of the skeleton found in step 3. The first theorem states that Step 4 is well defined, namely no link is oriented both ways. The second theorem states that every link that is oriented must be oriented the way the algorithm defines.

Theorem 4 *Let P be a strictly-positive distribution. If P is well-represented by a simple Bayesian network, then no link would be oriented both ways by step 4.*

Theorem 5 *Let P be a strictly-positive distribution. If D is a simple Bayesian network that represents P well, and $a - b - c$ is a chain in the skeleton of D, then the trail $a \rightarrow b \leftarrow c$ is part of D if and only if $I(a, \emptyset, c)$ holds in P.*

Step 5 leaves freedom to choose the orientation of some links in the skeleton. For example, the dags: $a \rightarrow b \rightarrow c$, $a \leftarrow b \leftarrow c$, and $a \leftarrow b \rightarrow c$ are three possible orientations of $a - b - c$. However, these three dags are indistinguishable (isomorphic) in the sense that they portray the same set of independence assertions. Hence no algorithm that relies on measuring independence can distinguish between them. On the other hand, the dag $a \rightarrow b \leftarrow c$ is distinguishable from the previous three because it portrays a new independence assertion, $I(a, \emptyset, c)$, which is not represented in either of the former dags. And our algorithm uses this distinction to orient these edges.

Isomorphism defines the theoretical limitation on the ability to identify directionality of links, using information about independence.

Definition Two Bayesian networks D_1 and D_2 are *isomorphic* if every probability distribution representable by D_1 is also representable by D_2 and vice versa.

Theorem 6 [9] *Two Bayesian networks are isomorphic iff they share the same skeleton and the same head-to-head connections emanating from non-adjacent nodes.*

Corollary 7 *Two simple Bayesian networks are isomorphic iff they share the same skeleton and the same head-to-head connections.*

Proof: Follows from Theorem 6 and from the fact that in simple dags every head-to-head connection must emanate from non-adjacent nodes. □

Corollary 7, shows that all orientations of step 5 that do not introduce a head-to-head connection yield isomorphic dags because these simple dags share the same skeleton and the same head-to-head connections. Thus, in order to decide whether or not P can be well-represented by a simple Bayesian network it is sufficient to examine **one** simple dag produced by step 5, as performed by step 6, because all other dags are isomorphic.

It remains to show that steps (a) through (f) do the orientation of step 5 correctly, namely, that these steps find an orientation that yields a simple dag without introducing new head-to-head connections if such a dag exists and fail if no such orientation is possible.

In steps a and b, any link whose end-labels are $(0,1)$ must be oriented from 1 to 0 or else a new head-to-head connection is introduced. In step c, if an undirected link with end-labels $(1,1)$ is created then there exists no way to orient that link without creating a new head-to-head connection. In step d, if the remaining undirected graph is not a forest, then it contains a cycle and there exists no way to orient a cycle without introducing new head-to-head connections. At step e, any orientation induced cannot create new head-to-head connections because any edge that is part of a head-to-head connection would have been already oriented in step b. Hence, if the graph resulting from step e is not a simple dag, then step e must have created a cycle. Thus, any alternative orientation of step e that would not add a head-to-head connection must yield a graph that is not a simple dag. Therefore step f correctly fails if one orientation of step e does not create a simple dag, and succeeds otherwise.

4 Gaussian Bayesian networks

Often continuous variables are needed to model an empirical environment (e.g., body heat, blood pressure, bone density, etc.) and these can also be represented using directed acyclic graphs.

Definition A *Gaussian Bayesian network* is a dag where each node represents a variable that is the linear combination of the variables corresponding to its parents, plus a term representing noise. The noise sources are assumed to be independent, normally distributed, and have zero means and non-zero variances. More explicitly, a variable corresponding to node x is governed by

$$x = a_1 z_1 + a_2 z_2 + \dots a_k z_k + z$$

where $z_1, z_2, \dots, z_k$ are the variables corresponding to the parents of x, and z is a noise term.[5]

Notice that each Gaussian Bayesian network D is associated with a joint Gaussian distribution P because all noise terms are Gaussian and linear combinations of Gaussian variables are Gaussian. The network D is said to *represent* P *well* if every two nodes connected with a trek always correspond to variables whose correlation coefficient is non-zero.

Interestingly, since for Gaussian distributions every type of dependence must be linear, the algorithm of the preceding section can answer the following question: Given a correlation matrix of a Gaussian distribution P, find whether P can be well-represented by a simple Gaussian causal network, and if so find this network.

For example, suppose we are measuring a correlation matrix

$$\begin{pmatrix} 1 & 0 & r_{13} & r_{14} \\ 0 & 1 & r_{23} & r_{24} \\ r_{13} & r_{23} & 1 & r_{13}r_{14} + r_{23}r_{24} \\ r_{14} & r_{24} & r_{13}r_{14} + r_{23}r_{24} & 1 \end{pmatrix}$$

of a multivariate Gaussian distribution P (where $r_{ij} \neq 0$) and suppose the means of all variables is zero. Using the recovery algorithm the underlying network can be recovered. We start with a complete graph of four vertices. Then we remove any link for which $I(z_i, U \setminus \{z_i, z_j\}, z_j)$ holds in P. Such a statement holds if and only if the determinant of the correlation matrix resulting from removing line i and column j is zero. Computing a determinant can be performed in $O(n^3)$ steps by Gauss elimination where n is the size of the matrix. In our example, link $z_3 - z_4$ is removed in this step

[5] Traditionally, noise terms are not depicted in Gaussian networks.

because

$$\begin{vmatrix} 1 & 0 & r_{13} \\ 0 & 1 & r_{23} \\ r_{14} & r_{24} & r_{13}r_{14} + r_{23}r_{24} \end{vmatrix} = 0.$$

Next we remove link $z_1 - z_2$ because the correlation between z_1 and z_2 is zero, a necessary and sufficient condition for $I(z_1, \emptyset, z_2)$ to hold in P. Lastly we orient the links and obtain the network below. The coefficient associated with each link $z_i - z_j$ is simply the correlation between its end points, namely, the entry (i, j) of the correlation matrix.

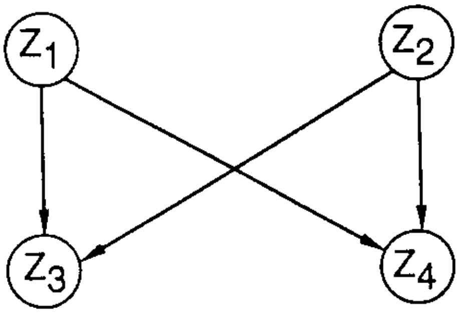

Figure 4: The recovered network

The task of recovering non-simple Gaussian networks is treated in [6] where the subject of Gaussian networks is covered more fully. The main advantage of our recovery algorithm is its *polynomial complexity* which is due to not using any search procedures in the recovery process.

5 Summary and Discussion

In the absence of temporal information, determining directionality of interactions is essential for inferring causal relationships. This paper provides conditions under which the directionality of some links in a causal schema is uniquely determined by the dependencies that surround the link. It is shown that if a distribution is generated from a simple causal schema, then the topology of the schema can be recovered uniquely, provided that the distribution satisfies some reasonable restrictions. Although the assumption of simple dags is somewhat restrictive, the recovery of such dags demonstrates the feasibility of determining causal asymmetries from information about dependencies, which is inherently symmetric. It also highlights the nature of these asymmetries thus, facilitating extensions to general graphs (see last paragraph).

Another useful feature of our algorithm is that its input can be obtained either from empirical data or from expert judgments or a combination thereof. Traditional methods of data analysis rely exclusively on statistical records which might not be available. Independence assertions, on the other hand, are readily provided by domain experts.

We are far from claiming that the method presented in this paper discovers genuine physical influences between causes and effects. First, a sensitivity analysis is needed to determine how vulnerable the algorithm is to errors associated with inferring conditional independencies from sampled data. Second, such a discovery requires breaking away from the confines of the closed world assumption, while we have assumed that the set of variables U adequately summarizes the domain. This assumption does not enable us to distinguish between genuine causes and spurious correlations [15]; a link $a \rightarrow b$ that has been determined by our procedure may as well be represented by a chain $a \leftarrow c \rightarrow b$ where c is an unmeasured variable, not accounted for when the network is first constructed. Thus, the dependency between a and b which is marked as causal when $c \notin U$ is in fact spurious, and this can only be revealed when c becomes observable (or at least describable). Such transformations are commonplace in the development of scientific thought: What is currently perceived as a cause may turn into a spurious effect when more refined knowledge becomes available. The initial perception, nevertheless, serves an important cognitive function in providing a tentative and expedient encoding of dependence patterns at that level of abstraction.

Future research should explore structuring techniques that incorporate variables outside U. The addition of these so called "hidden" variables often renders graphical representations more compact and more accurate. For example, a network representing a collection of interrelated medical symptoms would be highly connected and of little use, but when disease variables are added, the interactions can often be represented by a simple network. Facilitating such decomposition is the main role of "hidden variables" in neural networks [8] and is also incorporated in the program TETRAD [6]. Pearl and Tarsi provide an algorithm that generates tree representations with hidden variables, whenever such a representation exists [13]. An extension of this algorithm to simple networks would further enhance our understanding of causal structuring.

Another valuable extension would be an algorithm that recovers general dags. Such algorithms have been suggested for distributions that are *graph-isomorph* [16,20]. The basic idea is to identify with each pair of variables x and y a minimal subset S_{xy} of other variables that shields x from y[6], to link by an edge any two variables for which no such subset exists, and to direct an edge from x to y if there is a variable z linked to y but not to x, such that $I(x, S_{xz} \cup \{y\}, z)$ does not hold (see Pearl 1988, page 397, for motivation). The algorithm of Spirtes et al. (1989) requires an exhaustive search over all subsets of variables, while that of [20] prunes the search starting from the Markov network. When applied to a distribution that is not graph isomorph the algorithm of Verma and Pearl yields a bidirected graph, where some of the edges obtain arrows pointing both ways, indicating spurious correlations due to hidden common causes. Remarkably, when the underlying distribution is dag isomorph save for the existence of hidden variables then the bidirected edges summarize precisely the totality of all hidden causes in the model, and those with single arrows are guaranteed to match the corresponding arrows in the model [20]. It is not clear whether the assumption of dag isomorphism (or even embedded dag isomorphism) is realistic in processing real data such as medical records or natural language texts. Notably, when applied to simple dags, both algorithms may use exponential number of independence verifications, since the parent set of a node in a simple dag may still be as large as the number of nodes in that dag (excluding one). The recovery algorithm developed here is polynomial.

Acknowledgements

The first author thanks Jeff Barnett for comments that helped improve the presentation.

[6] The set S_{xy} will turn out to contain ancestors of x or ancestors y.

References

[1] de Kleer, J.; and Brown, J. S. 1986. Theories of causal ordering. *Artificial Intelligence*, 29(1):33–62.

[2] Geiger, D. 1990. *Graphoids: A Qualitative Framework for Probabilistic Inference*. PhD thesis, UCLA Computer Science Department. Also appears as a Technical Report (R-142) Cognitive Systems Laboratory, CS, UCLA.

[3] Geiger, D.; Paz, A.; and Pearl, J. 1990. Learning causal trees from dependence information. In *AAAI*, pp. 770-776, Boston, Massachusetts.

[4] Geiger D.; and Pearl J. 1988. On the logic of causal models. In *Uncertainty in Artificial Intelligence 4*, Shachter R. D.; Levitt T.S.; Kanal L.N.; and Lemmer J.F. (Editors). Elsevier Science Publishers B.V. (North-Holland), pp. 3–12. 1990.

[5] Geiger, D.; Verma, T.S.; and Pearl, J. 1990. Identifying independence in Bayesian networks. *Networks*, 20, pp. 507-534.

[6] Glymour, C.; Scheines, R.; Spirtes, P.; and Kelly, K. 1987. *Discovering Causal Structure*. Academic Press, New York.

[7] Heckerman D. 1990. A tractable inference algorithm for diagnosing multiple diseases. In *Uncertainty in Artificial Intelligence 5*, Shachter R. D.; Levitt T.S.; Kanal L.N.; and Lemmer J.F. (Editors). Elsevier Science Publishers B.V. (North-Holland), pp. 163–171. 1990.

[8] Hinton, G. E. 1989. Connectionist learning procedures. *Artificial Intelligence*, 40(1-3):185–234.

[9] Pearl, J.; Geiger, D.; and Verma, T. S. 1989. The logic of influence diagrams. In J. Q. Smith R. M. Oliver (eds.), *Influence Diagrams, Beliefnets and Decision Analysis*, chapter 3. John Wiley & Sons Ltd. New York.

[10] Pearl, J.; and Paz, A. 1989. Graphoids: A graph-based logic for reasoning about relevance relations. In B. Du Boulay et al. (eds.), *Advances in Artificial Intelligence-II*, pp. 357–363. North Holland, Amsterdam.

[11] Pearl, J.; and Verma, S. T. 1987. The logic of representing dependencies by directed acyclic graphs. In *AAAI*, pp. 347–379, Seattle Washington.

[12] Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems.* Morgan-Kaufman, San Mateo.

[13] Pearl, J.; and Tarsi, M. 1986. Structuring causal trees. *Journal of Complexity*, 2:60–77.

[14] Shoham, Y. 1987. *Reasoning About Change.* MIT Press, Boston MA.

[15] Simon, H. 1954. Spurious correlations: A causal interpretation. *Journal American Statistical Association*, 49:469–492.

[16] Spirtes, P.; Glymour, C.; and Scheines, R. 1989. Causality from probability. Technical Report CMU-LCL-89-4, Department of Philosophy Carnegie-Mellon University.

[17] Spohn, W. 1990. Direct and indirect causes. *Topoi*, 9.

[18] Suppes, P. 1970. *A Probabilistic Theory of Causation.* North Holland, Amsterdam.

[19] Verma, T. S.; and Pearl J. 1988. Causal networks: Semantics and expressiveness. In *Uncertainty in Artificial Intelligence 4*, Shachter R. D.; Levitt T.S.; Kanal L.N.; and Lemmer J.F. (Editors). Elsevier Science Publishers B.V. (North-Holland), pp. 69–76. 1990.

[20] Verma, T. S; and Pearl J. 1990. UCLA Cognitive Systems Laboratory Technical Report (R-150), in *proceedings*, sixth conference on Uncertainty in AI, Cambridge, Mass. July 27-29, 1990, pp. 220-227.

Appendix: Proofs

Lemma 8 below follows directly from the definition of simple dags and from Theorem 1.

Lemma 8 *: Let D be a simple Bayesian network representing a distribution P over a set of variables U and let a and b be two nodes not connected with link in D. If a and b have a common direct child, then $I(a, \emptyset, b)$ holds in P. And if they do not have a common direct child, then $I(a, U \setminus \{a, b\}, b)$ holds in P.*

Proof: If a and b have a common direct child then, since D is simple, every trail t between a and b contains a head-to-head node with respect to t. Consequently, by Theorem 1, $I(a, \emptyset, b)$ holds in P. If a and b do not have a common direct child then, since D is simple, every trail t between a and b contains a node that is not a head-to-head node with respect to t. Consequently, by Theorem 1, $I(a, U \setminus \{a, b\}, b)$ holds in P. □

Definition A dag is said to be *non-triangular* if every two nodes a and b that are connected with a direct link do not have a common direct child c (a, b and c are distinct nodes).

Theorem 2 *Let D be a non-triangular Bayesian network that represents a strictly-positive distribution P. Then, for every link $a - b$ in D, $I(a, U \setminus \{a, b\}, b)$ does not hold in P.*[7]

Proof: We shall use the notation $I_D(X, Z, Y)$ to stand for X and Y are graphically independent given Z and the notation $I_P(X, Z, Y)$ to stand for X and Y are conditionally independent given Z.

Let $a_1 \ldots a_n$ be an ordering of the vertices of D. Let $a_i \rightarrow a_j$ be a link in D. If $j = n$ then $I_P(a_i, U \setminus \{a_i, a_n\}, a_n)$ does not hold, for otherwise, the parent set of node a_n is not minimal. Assume that $i < j < n$ and, by contradiction, that $I_P(a_i, U \setminus \{a_i, a_j\}, a_j)$ holds. We will show that D cannot represent P. Nodes a_i and a_j cannot be both parents of a_n since this would imply the configuration $a_i \rightarrow a_n \leftarrow a_j$ with a_i connected to a_j in D contrary to its non-triangularity. Thus either $I_D(a_i, U \setminus \{a_i, a_n\}, a_n)$ or $I_D(a_j, U \setminus \{a_j, a_n\}, a_n)$ hold. Consequently, by Theorem 1, either $I_P(a_i, U \setminus \{a_i, a_n\}, a_n)$ or $I_P(a_j, U \setminus \{a_j, a_n\}, a_n)$ must hold which together with $I_P(a_i, U \setminus \{a_i, a_j\}, a_j)$ imply that $I_P(a_i, U \setminus \{a_i, a_j, a_n\}, a_j)$ holds as well because strictly positive distributions

[7] Notice, that we prove this theorem for the class of non-triangular dags which includes simple dags as a special case.

satisfy the following property:[8]

$$[I_P(a_i, U \setminus \{a_i, a_n\}, a_n) \vee I_P(a_j, U \setminus \{a_j, a_n\}, a_n)] \wedge$$

$$I_P(a_i, U \setminus \{a_i, a_j\}, a_j) \rightarrow I_P(a_i, U \setminus \{a_i, a_j, a_n\}, a_j)$$

Similarly, a_{n-1} can not be a son of both a_i and a_j. Thus either $I_D(a_i, U \setminus \{a_i, a_n, a_{n-1}\}, a_{n-1})$ or $I_D(a_j, U \setminus \{a_j, a_n, a_{n-1}\}, a_{n-1})$ hold which together with $I_P(a_i, U \setminus \{a_i, a_j, a_n\}, a_j)$ (which is derived in the previous step) imply that $I_P(a_i, U \setminus \{a_i, a_j, a_{n-1}, a_n\}, a_j)$ must hold. Continuing this way, by descending induction we get that the $I_P(a_i, R_{ij}, a_j)$ holds where R_{ij} are all vertices in D with indices less than j not including a_i. The link $a_i \rightarrow a_j$ is therefore redundant (See Ex. 3.11 in [12]). Thus the parent set of node a_j is not minimal, contradiction. □

Theorem 3 *Let P be a strictly-positive distribution. If P can be well-represented by a simple Bayesian network D, then the* skeleton*of D is equal to the graph constructed in step 3.*

Proof: Denote with G_2 the undirected graph constructed in step 2 (by removing every link for which $I(a, U \setminus \{a, b\}, b)$ holds in P). Denote with G_3 the undirected graph constructed in step 3 (which is obtained from G_2 by removing every link for which $I(a, \emptyset, b)$ holds in P). Let $a - b$ be a link in the skeleton of D. We show that $a - b$ must be a link in G_3. Since D is simple, by Theorem 2, the link $a - b$ is part of G_2. Since D represents P well, $I(a, \emptyset, b)$ does not hold in P, hence the link $a - b$ is not removed from G_2 and is therefore a link in G_3.

That the converse holds, namely, a link in G_3 must be a link in the skeleton of D, is shown as follows. Let a and b be two nodes not connected with a link in D. We show that $a - b$ is not a link in G_3. By Lemma 8 either $I(a, \emptyset, b)$ or $I(a, U \setminus \{a, b\}, b)$ hold in P. Consequently, the link between a and b is removed at Step 2 or at Step 3 and therefore it is not a link in G_3. □

Theorem 4 *Let P be a strictly-positive distribution. If P is well-represented by a simple Bayesian network D, then no link would be oriented both ways by step 4.*

[8] This property follows directly from the graphoid axioms [10] and can also be proven from the definition of conditional independence. It does not hold without assuming strict-positiveness.

Proof: By Theorem 3, the skeleton of D equals G_3. Assume, by contradiction, that there exists a link $a - b$ in G_3 that can be oriented both ways. Then, there exist a neighbor q of b for which $I(a, \emptyset, q)$ holds in P that induces an orientation from a into b and there exists another node p, neighbor of a, for which $I(b, \emptyset, p)$ holds in P that induces the reverse orientation. Thus, G_3 must contain the chain $p - a - b - q$. Clearly, either a or b are not head-to-head nodes wrt this trail. Consequently either a and q are connected with a trek or p and b are connected with a trek. In both cases D does not represent P well because $I(a, \emptyset, q)$ and $I(b, \emptyset, p)$ hold in P. Contradiction. □.

Theorem 5 *Let P be a strictly-positive distribution. If D is a simple Bayesian network that represents P well, and $a - c - b$ is a chain in the skeleton of D, then the trail $a \rightarrow c \leftarrow b$ is part of D if and only if $I(a, \emptyset, b)$ holds in P.*

Proof of Theorem 5: If $a \rightarrow c \leftarrow b$ is part of D, then by Lemma 8 $I(a, \emptyset, b)$ holds in P. And if it is not part of D but is part of the skeleton of D, then a and b are connected via a trek $a - c - b$ and therefore, since D represents P well, $I(a, \emptyset, b)$ does not hold in P. □