Heuristic Greedy Search Algorithms for Latent Variable Models

Peter Spirtes (Department of Philosophy, Carnegie Mellon University, ps7z@andrew.cmu.edu), Thomas Richardson(Department of Statistics, University of Washington) , and Chris Meek (Microsoft)

I. Introduction

There has recently been significant progress in the development of algorithms for learning the directed acyclic graph (DAG) part of a Bayesian network without latent variables from data and optional background knowledge. However, the problem of learning the DAG part of a Bayesian network with latent (unmeasured) variables is much more difficult for two reasons: first the number of possible models is infinite, and second, calculating scores for latent variables models is generally much slower than calculating scores for models without latent variables.

In this paper we will describe how to extend search algorithms developed for non-latent variable DAG models to the case of DAG models with latent variables. We will introduce two generalizations of DAGs, called mixed ancestor graphs (or MAGs) and partial ancestor graphs (or PAGs), and briefly describe how they can be used to search for latent variable DAG models, to classify, and to predict the effects of interventions in causal systems.

II. Directed Acyclic Graphs (DAGs)

A Bayesian network consists of two distinct parts: a directed acyclic graph (DAG or belief-network structure) and a set of parameters for the DAG. Under the statistical interpretation of a DAG, a DAG with a set of vertices **V** represents a set of probability measures over **V**. (We place sets of variables and defined terms in boldface.) Following the terminology of Lauritzen *et al.* (1990) say that a probability measure over a set of variables **V** satisfies the **local directed Markov property** for a directed acyclic graph (or DAG) G with vertices **V** if and only if for every W in **V**, W is independent of **V**\(**Descendants**(W) ∪ **Parents**(W)) given **Parents**(W), where **Parents**(W) is the set of parents of W in G, and **Descendants**(W) is the set of descendants of W in **G**. (Note that a vertex is its own ancestor and descendant, although not its own parent or child.) A DAG G **represents** the set of probability measures which satisfy the local directed Markov property for G. Variants of probabilistic DAG models were introduced in the 1980's in Pearl (1988) among others. Many familiar parametric models, such as non-recursive structural equation models with uncorrelated errors, factor analytic models, item response models, etc. are special cases of parameterized DAGs. (See Pearl 1988 for references.)

Under the causal interpretation, a DAG represents the causal relations in a given population with a set of vertices **V** when there is an edge from A to B if and only if A is a direct cause of B relative to **V**. The use of DAGs to simultaneously represent a set of causal hypotheses and a family of probability distributions extends back to the path diagrams introduced by Sewell Wright (1934). For the class of models considered in this paper we make two assumptions relating causal DAGs to probability distributions.

Causal Independence Assumption: If A does not cause B, and B does not cause A, and there is no third variable that causes both A and B, then A and B are independent.

Causal Faithfulness Assumption: If a causal DAG M correctly describes the causal structure in a population with probability distribution P, then each conditional independence true in in P is entailed by M.

These assumptions linking the statistical and causal interpretations of DAGs are defended in Spirtes, Glymour and Scheines (1993).

III. Partial Ancestral Graphs (PAGs)

In some cases, not all of the variables in a DAG can be measured. We call those variables whose values are measured the observed variables, and all other variables in the DAG latent variables. For a given division of the variables in a DAG G into observed and latent, we write G(**O**,**L**) where **O** is the set of observed variables and **L** is the set of latent variables.

A DAG G **entails a conditional independence relation** if and only if it is true in every probability measure satisfying the local directed Markov property for G. Two directed graphs G_1(**O**,**L**) and G_2(**O'**,**L'**) are **conditional independence equivalent** if and only if **O** = **O'**, and for all **X**, **Y** and **Z** included in **O**, G_1(**O**,**L**) entails **X** and **Y** are independent conditional on **Z** if and only if G_2(**O**,**L**) entails **X** and **Y** are independent conditional on **Z**. We denote the set of directed acyclic graphs that are conditional independence equivalent to *G*(**O**,**L**) as **Equiv**(*G*(**O**,**L**)).

A partial ancestral graph (PAG) can be used to represent any subset of **Equiv**(*G*(**O**,**L**)). A PAG is an extended graph consisting of a set of vertices **O**, and a set of edges between vertices, where there may be the following kinds of edges: A ↔ B, A o—o B, A o→ B, A ←o B, A → B or A ← B. We say that the A endpoint of A → B is "–"; the A endpoint of an A ↔ B, A ←o B, or A ← B edge is "<"; and the A endpoint of a A o—o B or A o→ B is "o". The conventions for the B endpoints are analogous. In addition pairs of edge endpoints may be connected by underlining (interpreted below). A partial ancestral graph for a set of directed acyclic graphs **G** each sharing the same set of observed variables **O**, contains partial information about the ancestor relations in **G**, namely only those ancestor relations common to all members of **G**. (If we allow **G** to contain directed cyclic graphs as well as directed acyclic graphs then several extra types of edges are needed in the PAG. (See Richardson, 1996) In the following definition, which provides a semantics for PAGs we use "*" as a meta-symbol indicating the presence of any one of {o, –, >}, e.g. A *→ B represents either A → B, A ↔ B, or A o→ B.

Partial Ancestral Graphs (PAGs)

If **G** is a set of directed acyclic graphs included in **Equiv**(*G*(**O**, **L**)), Ψ (with vertices **O**) is a PAG for **G** if and only if

(i) There is an edge between A and B in Ψ if and only if every DAG in **G** does not entail that A and B are independent conditional on any subset of **O**\{A,B}.

(ii) If there is an edge in Ψ out of A, i.e. A → B, then A is an ancestor of B in every graph in **G**.

(iii) If there is an edge in Ψ into B, i.e. A*→ B, then in every DAG in **G**, B is **not** an ancestor of A.

(iv) If there is an underlining A*—$\underline{*B*}$—*C in Ψ then B is an ancestor of (at least one of) A or C in every DAG in **G**.

(v) Any edge endpoint not marked in one of the above ways is left with a small circle thus: o—*.

Some examples of PAGs are shown in Figure 1, where **O** = {A,B,C,D}. In cases where the distinction between latent variables and measured variables is important, we enclose latent variables in ovals. (The MAGs in Figure 1 are defined in the next section.)

The requirement that **G** is included in **Equiv**(*G*(**O**, **L**)) guarantees that if one directed acyclic graph in **Equiv**(*G*(**O**, **L**)) does not entail that A and B are independent conditional on any subset of **O**\{A,B}, then all directed acyclic graphs in **Equiv**(*G*(**O**, **L**)) do not entail that A and B are independent conditional on any subset of **O**\{A,B}.

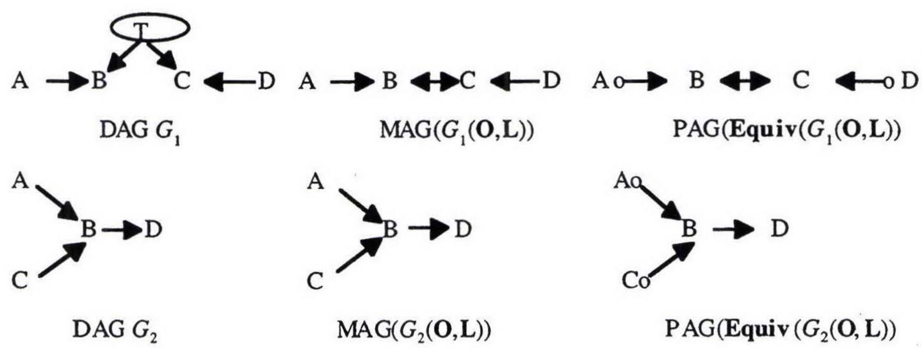


Figure 1

Note that only condition (i) gives necessary and sufficient conditions about features of the PAG. All of the other conditions are merely necessary conditions. That means that there can be more than one PAG representing a given set **G**; two such PAGs have the same adjacencies, but one may contain a "o" endpoint where the other contains a "–" or "> " endpoint. There are PAGs for **Equiv**(*G*(**O**, **L**)) with enough orientation information to determine whether or not each DAG in **Equiv**(*G*(**O**, **L**)) entails that **A** and **B** are independent conditional on any subset included in $\mathbf{O}\backslash(\mathbf{A} \cup \mathbf{B})$; we will say that any such PAG that has enough orientations to do this is "weakly complete" for $\mathbf{Equiv}(G(\mathbf{O}, \mathbf{L}))$. (Weak completeness does *not* entail that every ancestor relation common to every member of $\mathbf{Equiv}(G(\mathbf{O}, \mathbf{L}))$ is explicitly represented in the PAG.)

Thus a PAG can be used to represent both the ancestor relations among the members of $\mathbf{O}$ common to members of $\mathbf{G}$, and the set of conditional independence relations among the members of $\mathbf{O}$ in $\mathbf{G}$. Some PAGs (e.g. PAG($\mathbf{Equiv}(G_1(\mathbf{O},\mathbf{L}))$) in Figure 1) represent a set of conditional independence relations not entailed by any DAG $G(\mathbf{O},\mathbf{L})$ where $\mathbf{L} = \varnothing$.

PAGs have two distinct uses. Just as DAGs can be used by algorithms to perform fast conditionalizations, PAGs can be used in a similar way. And just as, given a causal interpretation, DAGs can be used to calculate the effects of any ideal intervention upon a system, PAGs, given a causal interpretation, can be used to calculate the effects of *some* ideal interventions upon a system. (See Spirtes et al. 1993, where PAGs are called POIPGs.)

While it would generally be preferable to know the true causal DAG $G(\mathbf{O},\mathbf{L})$ rather than a PAG representing $\mathbf{Equiv}(G(\mathbf{O}, \mathbf{L}))$, there are several reasons why it may be easier to find a PAG representing $\mathbf{Equiv}(G(\mathbf{O}, \mathbf{L}))$ than it is to find $G(\mathbf{O},\mathbf{L})$ itself. First the space of PAGs is finite, while the space of DAGs with latent variables is infinite. Second, for a variety of scores for models (such as BIC, posterior probability, etc.) there may be many different DAGs which receive the same score, but represent different causal theories and make different predictions about the effects of interventions upon a system. The data alone does not allow one to distinguish between these models, so even with population data, one cannot be sure which is the correct causal model. Nevertheless, for some (but not all) equivalence classes of causal models, and some (but not all) ideal interventions, it is possible to use a PAG to consistently estimate the effect of the intervention, even without knowing which causal model represented by the PAG is the correct model. Note that this strategy is not useful in instances where every pair of measured variables has some strong latent common cause; in that case the PAG that represents $\mathbf{Equiv}(G(\mathbf{O}, \mathbf{L}))$ is completely connected, and cannot be used to predict the effects of any ideal interventions on the system.

Is it possible to find a PAG from data and background knowledge? The FCI algorithm, under a set of assumptions described in Spirtes et al. 1993, is guaranteed in the large sample limit to find a weakly complete correct PAG for a given distribution. It uses a series of conditional independence tests to construct a PAG that represents a given distribution. The algorithm is exponential in the number of vertices in the PAG in the worst case (as is any algorithm based upon conditional independence tests.) However, the large sample reliability does not guarantee reliability on realistic sample sizes, and if the power of the conditional independence tests is low, the results of the tests are not compatible with any single PAG. For these reasons, it would be desirable to have a search that was not based upon conditional independence tests, or could be used to supplement an algorithm based upon conditional independence tests by using the output of the FCI algorithm as a starting point for a search.

Recently, a number of algorithms for searching for DAGs without latent variables have been developed that do not rely on conditional independence tests. (Chickering et al. 1995, Spirtes and Meek 1995) Instead, these are heuristic searches that attempt to maximize a score. We will describe here a heuristic PAG search that attempts to find a PAG with the highest score. One problem with this approach is that because a PAG represents a set of DAG models which may receive different scores (either Bayes Information Criterion, posterior probability, etc.) a PAG cannot be assigned a score by setting its score equal to an arbitrarily chosen DAG that it represents. In the next section we will show how to indirectly assign a score to a PAG.

IV. Mixed Ancestral Graphs (MAGs)

A MAG (or mixed ancestral graph) is a completely oriented PAG for a set of graphs which consists of a single directed acyclic graph $G(\mathbf{O},\mathbf{L})$. (By completely oriented we mean that there are no "o" endpoints on any edge). Some examples of MAGs are shown in Figure 1, where $\mathbf{O} = \{A,B,C,D\}$.

A MAG can also be considered a representation of a set of conditional independence relations among variables in $\mathbf{O}$ (which in some cases cannot be represented by any DAG containing just variables in $\mathbf{O}$; e.g. MAG($G_1(\mathbf{O},\mathbf{L})$) in Figure 1.) A MAG imposes no restrictions on the set of distributions it represents other than the conditional independence relations that it entails. (The class of MAGs is neither a subset nor a superset of other generalizations of DAGs such as chain graphs, cyclic directed graphs, or cyclic chain graphs.)

MAGs have the following useful features:

- DAG G_1 in Figure 1 is an example of a DAG such that as the sample size increases without limit, the difference between the Bayes Information Criterion (BIC) of MAG(G_1,**O**) and the BIC of any DAG G' that contains only variables in **O** increases without limit almost surely. Hence in some cases a maximum likelihood estimate of the MAG parameters is a better estimator of some of the population parameters than the maximum likelihood estimate of any DAG parameters.

- In the large sample limit, for multi-variate normal or discrete distributions, any (possibly latent variable) DAG with a maximum BIC score is represented by the MAG with the highest BIC score among all MAGs.

- There is a three place graphical relation among disjoint sets of vertices (**A** is d-separated from B given **C**) which holds if and only if the MAG entails that **A** is independent of **B** conditional on **C**. D-separation in MAGs is a simple extension of Pearl's d-separation relation (Pearl 1988) defined over DAGs.

If a PAG Ψ represents **Equiv**(G(**O**,**L**)), we say that any MAG that represents graph G(**O**,**L**) is represented by Ψ. For every PAG, there is some MAG that it represents, and every MAG represented by a PAG receives the same BIC score. Thus a PAG can be assigned a score by finding some MAG that it represents, scoring the MAG, and assigning that score to the PAG. It is possible that a PAG represents some non-MAG model that receives a higher BIC score than any MAG represented by the PAG. However, assigning a MAG score to a PAG that represents it has the following desirable property. For any distribution P(**O**), if there is some DAG G that contains **O**, such that for any three disjoint sets of variables $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \subseteq \mathbf{O}$, **X** is independent of **Y** given **Z** if and only if **X** is d-separated from **Y** given **Z** in G, then **P(O)** is said to be **faithful** to G over **O**. For any multi-variate normal distribution P(**O**), if P(**O**) is faithful to some DAG G over **O**, then in the large sample limit the PAG that represents G receives the highest BIC score among all PAGs.

A. *Parameterizing MAGs*

We will describe how a parameterization of a MAG in the multi-variate normal case is an extension of a parameterization of a DAG corresponding to a "structural equation model". (Parameterization and estimation of parameters in the case of discrete variables is somewhat more difficult.)

The variables in a linear structural equation model (SEM) can be divided into two sets, the "error variables" or "error terms," and the substantive variables. Corresponding to each substantive variable X_i is a linear equation with X_i on the left hand side of the equation, and the direct causes of X_i plus the error term ε_i on the right hand side of the equation. Since we have no interest in first moments, without loss of generality each variable can be expressed as a deviation from its mean.

Consider, for example, two SEMs S_1 and S_2 over $\mathbf{X} = \{X_1, X_2, X_3\}$, where in both SEMs X_1 is a direct cause of X_2. The structural equations in Figure 2 are common to both S_1 and S_2:

$$X_1 = \varepsilon_1$$
$$X_2 = \beta_{21} X_1 + \varepsilon_2$$
$$X_3 = \varepsilon_3$$

Figure 2: Structural Equations for SEMs S_1 and S_2

where β_{21} is a free parameters ranging over real values, and $\varepsilon_1, \varepsilon_2$ and ε_3 are error terms. In addition suppose that $\varepsilon_1, \varepsilon_2$ and ε_3 are distributed as multivariate normal. In S_1 we will assume that the correlation between each pair of distinct error terms is fixed at zero. The free parameters of S_1 are $\theta = \langle\beta, \mathbf{P}\rangle$, where β is the set of linear coefficients $\{\beta_{21}\}$ and **P** is the set of variances of the error terms. We will use $\Sigma_{S_1(\theta_1)}$ to denote the covariance matrix parameterized by the vector θ_1 for model S_1. If all the pairs of error terms in a SEM S are uncorrelated, we say S is a SEM with **uncorrelated errors**.

S_2 contains the same structural equations as S_1, but in S_2 we will allow the errors between X_2 and X_3 to be correlated, i.e., we make the correlation between the errors of X_2 and X_3 a free parameter, instead of fixing it at zero, as in S_1. In S_2 the free parameters are $\theta = \langle\beta, \mathbf{P'}\rangle$, where β is the set of linear coefficients $\{\beta_{21}\}$ and **P'** is the set of variances of the error terms and the correlation between ε_2 and ε_3. If the correlations between any of the error terms in a SEM are not fixed at zero, we will call it a SEM with **correlated errors**.

It is possible to associate with each SEM with uncorrelated errors a directed graph that represents the causal structure of the model and the form of the linear equations. For example, the directed graph associated with the substantive variables in S_1 is $X_1 \rightarrow X_2 \quad X_3$, because X_1 is the only substantive variable that occurs on the right hand side of the equation for X_2.

It is generally accepted that correlation is to be explained by some form of causal connection. Accordingly if ε_2 and ε_3 are correlated we will assume that either ε_2 causes ε_3, ε_3 causes ε_2, some latent variable causes both ε_2 and ε_3, or some combination of these. We represent the correlated error between ε_2 and ε_3 by introducing a latent variable T that is a common cause of X_2 and X_3. If $\mathbf{O} = \{X_1,X_2,X_3\}$, the MAG for the directed graph associated with S_2 is $X_1 \rightarrow X_2 \leftrightarrow X_3$. The statistical justification for this is provided in Spirtes et al. (1996). It turns out that the set of MAGs is a subset of the set of recursive structural equation models with correlated errors. Hence, there are well known techniques(Bollen, 1992) for estimating and performing statistical tests upon MAG models such as S_2.

B. *The Bayes Information Criterion (BIC) Score of a MAG*

As the sample size increases without limit, the Bayes Information Criterion is an O(1) approximation of a function of the posterior distribution. In the case of a multi-variate normal structural equation model, for a given sample

$$\text{BIC}(M\text{ , sample}) = \text{L}(\Sigma_{M(\theta_{\max})}\text{,sample}) - \ln(\text{samplesize} * \text{number of variables}) * \text{df}_M\text{, where}$$

- θ_{max} is the maximum likelihood estimate of the parameters for model *M* from sample,
- $\Sigma_{M(\theta_{\max})}$ is the covariance matrix for *M* when Θ takes on its maximum likelihood value θ_{max},
- $\text{L}(\Sigma_{M(\theta_{\max})}$,sample) is the likelihood ratio test statistic of $\Sigma_{M(\theta_{\max})}$,
- df_M is the degrees of freedom of the MAG *M*.

(See Raftery, 1993).

C. *Greedy BIC MAG Search*

A greedy search among MAGs is given as input a MAG to start with (possibly a MAG with no edges). At each stage, the algorithm takes the MAG M it has constructed thus far and calculates the score of each MAG resulting from one-step modifications such as an edge addition (directed or bi-directed) to *M*, removal of one edge (directed or bi-directed) from *M*, or reversal of one directed edge in *M*. If none of these changes improves the BIC score of *M*, the algorithm halts and outputs *M*. Otherwise the change that most improves the BIC score is made to *M* and the process is repeated.

Even at large sample sizes, this search suffers from the following problem. At each stage, there may be many MAGs that receive the same BIC score, and the algorithm arbitrarily chooses one of them. While two MAGs M and M' may receive the same BIC score, the one edge modification to M may receive a much higher BIC score than the same one edge modification to M'. Thus if the search halts when it cannot improve the score, it may halt at M', which is a local rather that a global maximum.

V. Greedy BIC PAG Search

A greedy PAG search solves some of the problems associated with a greedy MAG search. First, there are many fewer PAGs than MAGs. Second, MAGs that are score-equivalent in the sense that they receive the same BIC score for every data set will all be represented by the same PAG, and no two PAGs are score-equivalent. Hence the search does not suffer from the problem of having to choose arbitrarily from many score-equivalent alternatives. The search is described below and illustrated in Figure 3. It is basically a latent variable version of a search devised by C. Meek and described in Spirtes and Meek (1995).

```
procedure GBPS(PAG; data);
begin
 MAG:=PAG-to-MAG(PAG);
 current-score:=score(MAG,data);
 max-score:=current-score;
 while max-score <= current-score do
 begin
  new-PAG:=add-best-edge-to-PAG(PAG);
  MAG:=PAG-to-MAG(new-PAG);
  current-score:=score(MAG,data);
  if current-score > max-score then
   begin
    max-score:=current-score;
     PAG:=new-PAG;
  end;
 end;
 current-score:=max-score;
 while max-score <= current-score do
 begin
  new-PAG:=remove-worst-edge-in-PAG(PAG);
  MAG:=PAG-to-MAG(new-PAG);
  current-score:=score(MAG,data);
  if current-score > max-score then
   begin
    max-score:=current-score;
     PAG:=new-PAG;
  end;
 end;
 return(PAG);
end;
```

The search starts with some initial PAG. This could come from background knowledge, another search procedure such as FCI, or could simply be a PAG with no edges. The PAG is then turned into a MAG in order to assign a score to it. The search then looks for the single best edge to add to the initial PAG. We consider all one edge extensions of MAG M which entail a proper subset of the conditional independence relations entailed by M. In the example of Figure 3 there are four such single edge PAG extensions of the initial PAG. Each of these four extensions is turned into a MAG in order to score it. The MAG with the best score is chosen, and turned back into its corresponding PAG. These steps are then repeated until the score cannot be improved. At this stage the search then removes edges until the score can no longer be improved.

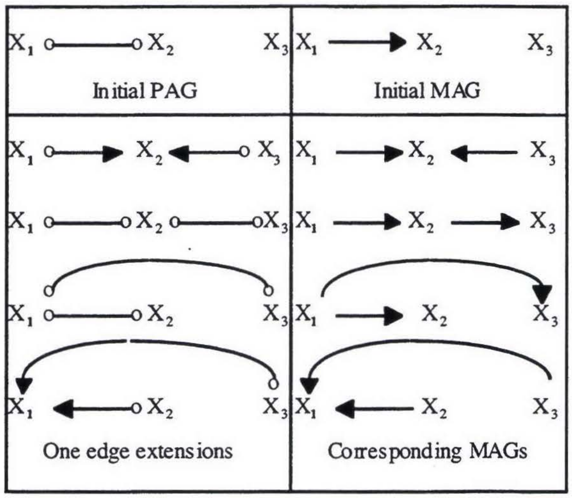


Figure 3

We conjecture that this search is asymptotically correct, as long as the distribution from which the sample data are drawn is the marginal of a distribution faithful to some directed acyclic graph. It is worst case exponential in the number of vertices because of the remove-worst-edge-in-PAG step. In addition, we do not know the complexity of the PAG-to-MAG step, because we do not know how much back-tracking may be needed in order to turn a PAG into a MAG. We do not currently know the number of variables that this kind of search can feasibly be performed on. The current implementation is not practical for 30 variables, but could be greatly speeded up.

VI. Simulation Study

As a preliminary simulation study, we chose two graphs G_1 and G_2 with latent variables (Figure 4), and for sample sizes 2500, 1000, 500, and 250 generated 5 pseudo-rendom samples from them. The error variables were standard normal, and the linear coefficients were between 0.5 and 1.5, and did not vary with sample size or sample. The input to the algorithm is the data, and the output is a PAG. Because determining whether X is an ancestor of Y is important for predicting the effects of interventions on X, we measure the performance of the algorithm by counting for how many ordered pairs of variables <X,Y> the output PAG implies that X is an ancestor of Y (#a in Table 1, averaged over the 5 samples at a given sample size), and what percentage of the time the ancestor implication is correct in the graph that generated the data (%ac, averaged over the 5 samples at a given sample size). We construct similar measures for non-ancestor relations (#na and %nac respectively). (It is important to realize it may not be possible to correctly infer all of the ancestor and non-ancestor relations in the true DAG, even with population data, because there may be DAGs that entail the same conditional independence relations as the true DAG,but that orient the edges differently. In G_1, for example only three of the ancestor relations among the measured variables can be reliably inferred) In Table 1, size represents the sample size. In G_1, in 20% of the ordered pairs of distinct measured variables <X,Y>, X is an ancestor of Y; in G_2, in 30% of the ordered pairs of distinct measured variables <X,Y>, X is an ancestor of Y. In the case of large sample sizes and sparse graphs, with perfectly normal data, and only a few latent variables, the algorithm performs quite well (see Table 1). However, we expect the algorithm's performance to be a function of the edge coefficients, how many vertices each vertex in the graph is adjacent to, the sample size, the number and strength of latent variables, the amount of selection bias, and deviations from normality. In order to evaluate the algorithm's performance more simulation tests are needed, as well as applications to real data.

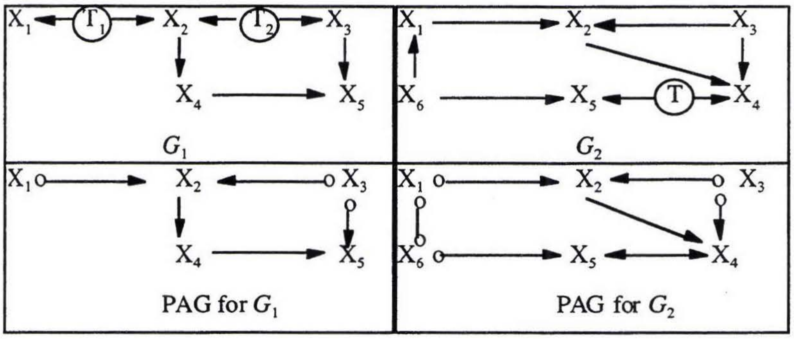


Figure 4

	G_1					G_2				
size	2500	1000	500	250	100	2500	1000	500	250	100
#a	3	3	3	3	3	1	1	1	.6	.6
%ac	100	100	100	100	100	100	100	20	100	100
#na	11	11	11	11	11	19	19	19	12.4	7.4
%nac	100	100	100	100	100	100	100	100	98.3	94.6

Table 1

Bibliography

Bollen, K. (1989). Structural Equations with Latent Variables. Wiley, New York.

Chickering, D. and Geiger, D. and Heckerman, D. (1995). Learning Bayesian networks: Search methods and experimental results. Preliminary papers of the fifth international workshop on Artificial Intelligence and Statistics, Fort Lauderdale, FL, pp. 112-128.

Lauritzen, S., Dawid, A., Larsen, B., Leimer, H., (1990) Independence properties of directed Markov fields, Networks, 20, 491-505.

Pearl, J., (1988). Probabilistic Reasoning in Intelligent Systems, Morgan Kaufman: San Mateo, CA.

Pearl, J. (1995) Causal diagrams for empirical research, Biometrika, 82.

Raftery, A. (1993) Bayesian Model Selection in Structural Equation Models, in Testing Structural Equation Models, ed. by K. Bollen and S. Long, Sage Publications.

Richardson, T. (1996). A discovery algorithm for directed cyclic graphs. Uncertainty in Artificial Intelligence, Proceedings, 12th Conference, Morgan Kaufman, CA.

Spirtes, P., Glymour, C., and Scheines, R., (1993) Causation, Prediction, and Search, (Springer-Verlag Lecture Notes in Statistics 81, New York).

Spirtes, P., Richardson, T., Meek, C., Scheines, R., and Glymour, C. (1996). Using D-separation to Calculate Zero Partial Correlations in Linear Models with Correlated Errors, Carnegie Mellon University Technical Report Phil-72.

Spirtes, P., and Meek, C. (1995). Learning Bayesian Networks with Discrete Variables from Data", in Proceedings of The First International Conference on Knowledge Discovery and Data Mining, ed. by Usama M. Fayyad and Ramasamy Uthurusamy, AAI Press, pp. 294-299.

Wright, S. (1934). The method of path coefficients. Annals of Mathematical Statistics, 5, 161-215.

A Polynomial Time Algorithm For Determining DAG Equivalence in the Presence of Latent Variables and Selection Bias

by Peter Spirtes (Department of Philosophy, Carnegie Mellon University, ps7z@andrew.cmu.edu)
and Thomas Richardson (Department of Statistics, University of Washington)

Following the terminology of Lauritzen et. al. (1990) say that a probability measure over a set of variables **V** satisfies the **local directed Markov property** for a directed acyclic graph (DAG) G with vertices **V** if and only if for every W in **V**, W is independent of the set of all its non-descendants conditional on the set of its parents. One natural question that arises with respect to DAGs is when two DAGs are "statistically equivalent". One interesting sense of "statistical equivalence" is "conditional independence equivalence" which holds when two DAGs entail the same set of conditional independence relations. In the case of DAGs, conditional independence equivalence also corresponds to a variety of other natural senses of statistical equivalence (such as representing the same set of distributions). Theorems characterizing conditional independence equivalence for directed acyclic graphs and that can be used as the basis for polynomial time algorithms for checking conditional independence equivalence were provided by Verma and Pearl (1990), and Frydenberg (1990). The question we will examine is how to extend these results to cases where a DAG may have latent (unmeasured) variables or selection bias (i.e. some of the variables in the DAG have been conditioned on.) Conditional independence equivalence is of interest in part because there are algorithms for constructing DAGs with latent variables and selection bias that are based on observed conditional independence relations. For this class of algorithms, it is impossible to determine which of two conditional independence equivalent causal structures generated a given probability distribution, given only the set of conditional independence and dependence relations true of the observed distribution. We will describe a polynomial (in the number of vertices) time algorithm for determining when two DAGs which may have latent variables or selection bias are conditional independence equivalent.

A DAG G **entails a conditional independence relation R** if and only if R is true in every probability measure satisfying the local directed Markov property for G. (We place definitions and sets of variables in boldface.) Pearl, Geiger, and Verma (Pearl 1988) have shown that there is a graphical relation, d-separation, that holds among three disjoint sets of variable **A**, and **B**, and **C** in DAG G if and only if G entails that **A** is independent of **B** given **C**. A vertex Y is a **collider** on an undirected path U if U contains a subpath $X \rightarrow Y \leftarrow Z$. Say that a vertex V on an undirected path U between X and Y is **active** on U given **Z** (**Z** not containing X and Y) if and only if either V is not a collider on U and not in **Z**, or V is a collider on U and is an ancestor of **Z**. For three disjoint sets of variables **A**, **B**, and **C**, **A** is **d-connected** to **B** given **C** in graph G, if and only if there is an undirected path from some member of **A** to a member of **B** such that every vertex on U is active given **C**; for three disjoint sets of variables **A**, **B**, and **C**, **A** is **d-separated** from **B** given **C** in graph G, if and only **A** is not d-connected to **B** given **C**.

Two DAGs are **conditional independence equivalent** if and only if they have the same vertices and entail the same set of conditional independence relations . If two DAGs G_1 and G_2 are conditional independence equivalent, the set of distributions that satisfy the local directed Markov property for G_1 equals the set of distribution that satisfy the local directed Markov property for G_2. Theorems that provide the basis for polynomial time algorithms for testing conditional independence equivalence for DAGs were given in Verma and Pearl (1990), for cyclic directed graphs in Richardson (1994), and for directed acyclic graphs with latent variables in Spirtes and Verma (1992).

DAGs are also used to represent causal processes. Under this interpretation, a directed edge from A to B means that A is a direct cause of B relative to the variables in the DAG. Suppose a causal process represented by DAG G generates some population with a given distribution $P(\mathbf{V})$ that satisfies the local directed Markov property for G. If some of the variables in **V** are unmeasured, and some have been conditioned on (due to those variables being causally related to the sampling mechanism) then the set of conditional independence relations entailed for the subset of measured variables in the subpopulation from which the sample is drawn is not necessarily equal to the set of conditional independence relations entailed by any DAG (without latent variables or selection bias). Assume then that the variables in **V** can be partitioned into **O** (observed), **L** (latent), and **S** (selected, or conditioned on.) In that case instead of observing $P(\mathbf{V})$, we may be able to observe only $P(\mathbf{O}|\mathbf{S})$, that is the marginal distribution over the observed variables in the selected subpopulation. Let us call $P(\mathbf{O}|\mathbf{S})$ the "observed" distribution. There are algorithms which, under some plausible assumptions relating probability distributions to causal processes, are correct in the large sample limit, and that can construct a representation of the class of DAGs (that may have latent variables and variables conditioned on) that are compatible with the observed conditional independence relations. See Spirtes et al. 1993 for the latent variable case without selection bias, and Spirtes et al. 1995.

For a given DAG G, and a partition of the variable set $\mathbf{V}$ of G into observed ($\mathbf{O}$), selection ($\mathbf{S}$), and latent ($\mathbf{L}$) variables, we will write $G(\mathbf{O},\mathbf{S},\mathbf{L})$. Let us now extend the definition of conditional independence equivalence to the case where there may be latent variables and selection bias. Two directed graphs $G_1(\mathbf{O},\mathbf{L},\mathbf{S})$ and $G_2(\mathbf{O}',\mathbf{L}',\mathbf{S}')$ are **conditional independence equivalent** if and only if $\mathbf{O} = \mathbf{O}'$, and for all $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ included in $\mathbf{O}$, $G_1(\mathbf{O},\mathbf{L},\mathbf{S})$ entails $\mathbf{X}$ and $\mathbf{Y}$ are independent conditional on $\mathbf{Z} \cup \mathbf{S}$ if and only if $G_2(\mathbf{O}',\mathbf{L}',\mathbf{S}')$ entails $\mathbf{X}$ and $\mathbf{Y}$ are independent conditional on $\mathbf{Z} \cup \mathbf{S}'$. Intuitively, the conditional independence relations true in the observed distribution could have been generated either by the causal DAG $G_1(\mathbf{O},\mathbf{L},\mathbf{S})$ or by $G_2(\mathbf{O}',\mathbf{L}',\mathbf{S}')$. Information just about the observed conditional independence relations cannot distinguish any two DAGs which are conditional independence equivalent.

In order to state necessary and sufficient conditions for conditional independence equivalence, we will need the following concept. A mixed ancestral graph (MAG) is an extended graph consisting of a set of vertices $\mathbf{V}$, and a set of edges between vertices, where there may be the following kinds of edges: A $\leftrightarrow$ B, A o—o B, A o$\rightarrow$ B, A $\leftarrow$o B, A $\rightarrow$ B, or A $\leftarrow$ B. (A MAG may be considered a special case of a PAG that represents a single graph. See Richardson 1996.) We say that the A endpoint of an A $\rightarrow$ B edge is "–"; the A endpoint of an A $\leftrightarrow$ B, A $\leftarrow$o B, or. A $\leftarrow$ B edge is "<"; and the A endpoint of an A o—o B or A o$\rightarrow$ B edge is "o". The conventions for the B endpoints are analogous. A mixed ancestral graph for a directed acyclic graph $G(\mathbf{O},\mathbf{S},\mathbf{L})$ represents some of the ancestor relations in $G(\mathbf{O},\mathbf{S},\mathbf{L})$. In the following definition, which provides a semantics for MAGs we use "*" as a meta-symbol indicating the presence of any one of {o, –, >}, e.g. A *$\rightarrow$ B represents either A $\rightarrow$ B, or A $\leftrightarrow$ B.

Mixed Ancestral Graphs (MAGs)

A MAG represents directed acyclic graph $G(\mathbf{O},\mathbf{S},\mathbf{L})$ (in which case we write MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$)) if:

(i) If A and B are in $\mathbf{O}$, there is an edge between A and B in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) if and only for any subset $\mathbf{W} \subseteq \mathbf{O}\backslash\{A,B\}$, A and B are d-connected given $\mathbf{W} \cup \mathbf{S}$ in $G(\mathbf{O},\mathbf{S},\mathbf{L})$.

(ii) There is an edge A $\rightarrow$ B (or B $\leftarrow$ A) in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) if and only if A is an ancestor of B but not $\mathbf{S}$ in $G(\mathbf{O},\mathbf{S},\mathbf{L})$;

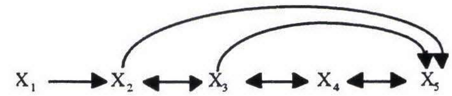
(iii) There is an edge A $\leftarrow$* B (or B *$\rightarrow$ A) in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) if and only if A is **not** an ancestor of B or $\mathbf{S}$ in $G(\mathbf{O},\mathbf{S},\mathbf{L})$;

(iv) There is an edge A o—* B (or B *—o A) in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) if and only if A is an ancestor of $\mathbf{S}$ in $G(\mathbf{O},\mathbf{S},\mathbf{L})$.

The definition of "d-separation" given for DAGs can be applied directly to MAGs, as long as such concepts as "undirected path", "collider", etc., are given their obvious extensions to MAGs. We include in the Appendix the definitions of terms such as "undirected path" etc. which apply both to directed graphs and MAGs.

The first step in forming a MAG for a graph is to form the ancestor matrix for the graph. Let n be the number of vertices in $\mathbf{O} \cup \mathbf{S} \cup \mathbf{L}$ and m the number of vertices in $\mathbf{O}$. Aho, Hopcroft, and Ullman (1974) describes a transitive closure algorithm for filling in such a matrix that is $O(n^3)$. Then each pair of vertices X and Y in $\mathbf{O}$ ($O(m^2)$) is adjacent in MAG($G_1(\mathbf{O},\mathbf{S},\mathbf{L})$) if and only if they are not d-separated ($O(n^2)$) given (**Ancestors**({X,Y} $\cup$ $\mathbf{S}$) $\cap$ $\mathbf{O}$) in $G_1(\mathbf{O},\mathbf{S},\mathbf{L})$ (where **Ancestors**($\mathbf{Z}$) is the set of vertices which are ancestors of vertices in $\mathbf{Z}$; see Lemma 5.) The orientation of each edge in the MAG ($O(m^2)$) can then be determined by examining the ancestor matrix. So forming a MAG is $O(n^3m)$.

If U is an acyclic undirected path containing X and B, and X is before B on U, then U(X,B) represents the unique subpath of U between X and B. If B is before X on U, by definition U(X,B) = U(B,X). In MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$), U is a **discriminating path** for B if and only if U is an undirected path between X and Y with at least three edges, U contains B, B $\neq$ X, B adjacent to Y on U, X is not adjacent to Y, and for every vertex Q on U(X,B) except for the endpoints Q is a collider on U(X,B) and there is an edge Q $\rightarrow$ Y in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$). See Figure 1.



< X_1,X_2,X_3,X_4,X_5> is discriminating path for X_4

Figure 1

If Y is adjacent to X and Z on a path *U*, and X and Z are not adjacent in thegraph, then Y is **unshielded** on *U*. MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the **same basic colliders** if and only if they have (i) the same adjacencies; (ii) the same unshielded colliders (iii) if *U* is a discriminating path for X in MAG(G_1(**O**,**S**,**L**)), and the corresponding path *U'* in MAG(G_2(**O**,**S'**,**L'**)) is a discriminating path for X, then X is a collider on *U* in MAG(G_1(**O**,**S**,**L**)) if and only if X is a collider on *U'* in MAG(G_2(**O**,**S'**,**L'**)).

Theorem 1: DAGs G_1(**O**,**S**,L) and G_2(**O**,**S'**,L') are conditional independence equivalent if and only if MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders.

Theorem 1 is the basis of an $O(n^3m^2)$ algorithm for determining conditional independence equivalence, where n is the maximum number of vertices in G_1(**O**,**S**,L) and G_2(**O**,**S'**,**L'**), and m is the number of vertices in **O**. The first step in determining conditional independence equivalence is to form MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)), which is $O(n^3m)$. Checking that the two MAGs have the same unshielded colliders is $O(m^3)$, and for each triple of vertices all of which are adjacent to each other, there is a simple algorithm that determines whether there is a discriminating path that examines each edge ($O(m^2)$) in the MAG at most once. Hence, overall the algorithm is $O(n^3m^2)$.

Appendix

For our purposes we need to represent a variety of marks attached to the ends of edges. In general, we allow that the end of an edge can be marked out of by "—", or can be marked with ">", or can be marked with an "o". In order to specify completely the type of an edge, therefore, we need to specify the variables and **marks** at each end. For example, the left end of "A o→ B" can be represented as the ordered pair [A, o] and the right end can be represented as the ordered pair [B, >]. We will also call [A, o] the A end of the edge between A and B. The first member of the ordered pair is called an endpoint of an edge, e.g. in [A, o] the endpoint is A. The entire edge is a set of ordered pairs representing the endpoints, e.g. {[A, o], [B, >]}. Note that the edge {[B, >],[A, o]} is the same as {[A, o],[B, >]} since it doesn't matter which end of the edge is listed first. Note that a directed edge such as A → B has a mark "—" at the A end.

We say a **graph** is an ordered triple <**V**,**M**,**E**> where **V** is a non-empty set of vertices, **M** is a non-empty set of marks, and **E** is a set of sets of ordered pairs of the form {[V_1,M_1],[V_2,M_2]}, where V_1 and V_2 are in **V**, $V_1 \neq V_2$, and M_2 and M_2 are in **M**. If *G* = <**V**,**M**,**E**> we say that *G* is **over V**. (Directed graphs and MAGs are both special cases of graphs.)

In a graph, for a directed edge A → B, the edge is **out of** A, and A is **parent** of B and B is a **child** of A. An edge A ← B, A↔B, or A ←o B is **into** A. A sequence of edges <E_1,...,E_n> in *G* is an **undirected path** if and only if there exists a sequence of vertices <V_1,...,V_{n+1}> such that for $1 \leq i \leq n$ E_i, has endpoints V_i and V_{i+1}, and $E_i \neq E_{i+1}$. A path *U* is **acyclic** if no vertex appears more than once in the corresponding sequence of vertices. We will assume that an undirected path is acyclic unless specifically mentioned otherwise. A sequence of edges <E_1,...,E_n> in *G* is a **directed path** *D* **from** V_1 to V_n if and only if there exists a sequence of vertices <V_1,..., V_{n+1}> such that for $1 \leq i \leq n$, there is a directed edge $V_i \rightarrow V_{i+1}$ on D. If there is an acyclic directed path from A to B or B = A then A is an **ancestor** of B, and B is a **descendant** of A. If **Z** is a set of variables, A is an **ancestor** of **Z** if and only if it is an ancestor of a member of **Z**, and similarly for **descendant**. If **X** is a set of vertices in *G*, let **Ancestors(X)** be the set of all ancestors of members of **X** in *G*(**O**,**S**,**L**). A vertex V is a **collider** on an undirected path *U* if and only if *U* contains a pair of distinct edges adjacent on the path and into V. The **orientation** of an acyclic undirected path between A and B is the set consisting of the A end of the edge on *U* that contains A, and the B end of the edge on *U* that contains B. Say that a vertex V on an undirected path *U* between X and Y is **active** on *U* given **Z** (**Z** not containing X and Y) if and only if either V is not a collider on *U* and not in **Z**, or V is a collider on *U* and is an ancestor of **Z**. For three disjoint sets of variables **A**, **B**, and **C**, **A** is **d-connected** to **B** given **C** in graph *G*, if and only if there is an undirected path from some member of **A** to a member of **B** such that every vertex on U is active given **C**; for three disjoint sets of variables **A**, **B**, and **C**, **A** is **d-separated** from **B** given **C** in graph G, if and only **A** is not d-connected to **B** given **C**.

In a **directed graph**, all of the edges are directed edges. A directed graph is **acyclic** if and only if it contains no directed cyclic paths. Lemma 1 is a simple generalization of Lemma 3.3.1 in Spirtes *et al.* (1993).

Lemma 1: In a directed acyclic graph G over a set of vertices **V**, if the following conditions hold:

(a) R is a sequence of vertices in **V** from A to B, $R \equiv \langle A \equiv X_0, \ldots X_{n+1} \equiv B \rangle$, such that $\forall i, 0 \leq i \leq n$, $X_i \neq X_{i+1}$ (the X_i are only *pairwise distinct* , i.e. not necessarily distinct),

(b) $\mathbf{Z} \subseteq \mathbf{V} \backslash \{A,B\}$,

(c) $\mathcal{T}$ is a set of undirected paths such that

(i) for each pair of consecutive vertices in R, X_i and X_{i+1}, there is a unique undirected path in $\mathcal{T}$ that d-connects X_i and X_{i+1} given $\mathbf{Z} \backslash \{X_i, X_{i+1}\}$,

(ii) if some vertex X_k in R is in **Z**, then the paths in $\mathcal{T}$ that contain X_k as an endpoint collide at X_k, (i.e. all such paths are directed into X_k)

(iii) if for three vertices X_{k-1}, X_k, X_{k+1} occurring in R, the d-connecting paths in $\mathcal{T}$ between X_{k-1} and X_k, and X_k and X_{k+1}, collide at X_k then X_k has a descendant in **Z**,

then there is a path U in G that d-connects $A \equiv X_0$ and $B \equiv X_{n+1}$ given **Z** that contains only edges occurring in $\mathcal{T}$.

U is an **inducing path** between X and Y in G(**O**,**S**,**L**), if and only U is an acyclic undirected path such that every member of $\mathbf{O} \cup \mathbf{S}$ on U is a collider on U, and every collider on U is an ancestor of $\{X,Y\} \cup \mathbf{S}$. (This is a generalization of the concept of inducing path that was introduced in Verma and Pearl 1990). The following sequence of lemmas state that for every subset **W** of **O**, X and Y are d-connected given $\mathbf{W} \cup \mathbf{S}$ in G(**O**,**S**,**L**) if and only there is an inducing path between X and Y in G(**O**,**S**,**L**). For space reasons we do not present the proofs here, but they are simple modifications of the proofs that appear in Spirtes et al. (1993), in which the case of latent variables without selection bias is considered. (There is no analog of Lemma 4 in Spirtes *et al.* 1993, but the proof is very similar to that of Lemma 2 and Lemma 3.)

Lemma 2: In directed graph G(**O**,**S**,**L**), if there is an inducing path between A and B that is out of A and into B, then for any subset **Z** of $\mathbf{O} \backslash \{A,B\}$ there is an undirected path C that d-connects A and B given $\mathbf{Z} \cup \mathbf{S}$ that is out of A and into B.

Lemma 3: If G(**O**,**S**,**L**) is a directed acyclic graph, and there is an inducing path U between A and B that is into A and into B then for every subset **Z** of $\mathbf{O} \backslash \{A,B\}$ there is an undirected path C that d-connects A and B given $\mathbf{Z} \cup \mathbf{S}$ that is into A and into B.

Lemma 4: If G(**O**,**S**,**L**) is a directed acyclic graph, and there is an inducing path U between A and B that is out of A and out of B then for every subset **Z** of $\mathbf{O} \backslash \{A,B\}$ there is an undirected path C that d-connects A and B given $\mathbf{Z} \cup \mathbf{S}$.

Lemma 5: If G(**O**,**S**,**L**) is a directed acyclic graph and an undirected path U in G(**O**,**S**,**L**) d-connects A and B given $((\mathbf{Ancestors}(\{A,B\} \cup \mathbf{S}) \cap \mathbf{O}) \cup \mathbf{S}) \backslash \{A,B\}$ then U is an inducing path between A and B.

The following lemma follows from a simple application of d-separation to discriminating paths.

Lemma 6: In MAG(G(**O**,**S**,**L**)), if U is a discriminating path for B between X and Y, and B is a collider on U then B is no set that d-separates X and Y, and if B is not a collider on U, B is in every set that d-separates X and Y.

Note that it follows directly from the definition of MAG(G(**O**,**S**,**L**)) that there are no edges A o—B in MAG(G(**O**,**S**,**L**)), and if there is an edge A o—* B in MAG((G(**O**,**S**,**L**)), then the A endpoint of every edge in MAG(G(**O**,**S**,**L**)) is "o". Hence if A is a collider on any path in MAG(G(**O**,**S**,**L**)), the A endpoint of no edge in MAG(G(**O**,**S**,**L**)) is a "o".

Let $l(U,C_i,\mathbf{Z})$ be the length of a shortest directed path from collider C_i on U to a member of **Z**. Let U be a **minimal d-connecting path** between X and Y given **Z** if and only if U is a d-connecting path between X and Y given **Z**, and there is no other path V d-connecting X and Y given **Z** such that either V has fewer edges than U, or V has the same number of edges as U and the sum over j of $l(V,D_j,\mathbf{Z})$ is less than the sum over i of $l(U,C_i,\mathbf{Z})$. Say that two undirected paths U and U' which contain a vertex C **disagree** at C if C is a collider on U but not on U', or vice-versa.

Lemma 7: If U is a minimal d-connecting path between A and B given **R** in MAG(G(**O**,**S**,**L**)), and E is an edge between C and D in U, but C and D are not adjacent on U, and U' is the result of substituting E in for U(C,D) in U, then either U(C,D) is into C and E is out of C, or U(C,D) is into D, and E is out of D.

Proof. If C and D are both active on U', then U' d-connects A and B given **R**, and is shorter than U, contradicting the assumption. Hence either C is not active on U', or D is not active on U'. If U' agrees with U at C and D, then C and D are both active on U'. Hence U' disagrees with U at C or D.

If C is a collider on U', but not on U, it follows that E is an edge D $*\!\!\rightarrow$ C, there is an edge M $*\!\!\rightarrow$ C on U, and U(C,D) is out of C. If there is no collider on U(C,D) then C is either an ancestor of D, or an ancestor of a vertex with a "o" endpoint. If C is an ancestor of a vertex with a "o" endpoint, then C is an ancestor of **S** in G(**O**,**S**,**L**), and hence there cannot be a D $*\!\!\rightarrow$ C edge in MAG(G(**O**,**S**,**L**)). If C is an ancestor of D, this contradicts the D $*\!\!\rightarrow$ C edge. It follows that there is a collider on U(C,D) and hence C is an ancestor of the first collider on U(C,D). It follows that C is an ancestor of **R**. Hence C is active on U'.

Similarly, if D is a collider on U' but not on U, D is active on U'. It follows that either C is a collider on U but not on U', or D is a collider on U but not on U'. Hence either U(C,D) is into C and E is out of C, or U(C,D) is into D, and E is out of D. $\therefore$

Lemma 8: If U is a minimal d-connecting path between A and B given **R** in MAG(G(**O**,**S**,**L**)), U contains C $*\!\!-\!\!*$ F $*\!\!-\!\!*$ D, and C and D are adjacent in MAG(G(**O**,**S**,**L**)), then MAG(G(**O**,**S**,**L**)) contains one of the following subgraphs:

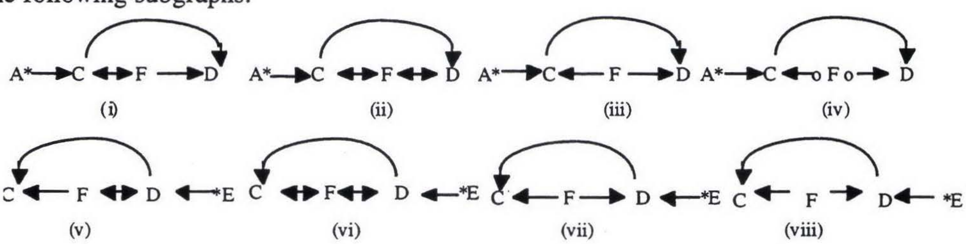


Figure 2

Proof. Let E be the edge between C and D, and U' be the result of substituting E in for U(C,D). By Lemma 7, either U(C,D) is into C and E is out of C, or U(C,D) is into D, and E is out of D. Suppose first that E is out of C and U(C,D) is into C. Then MAG(G(**O**,**S**,**L**)) contains either (i), (ii), (iii) or (iv) of Figure 2 or one of the following subgraphs (ix), (x), or (xi) of Figure 3:

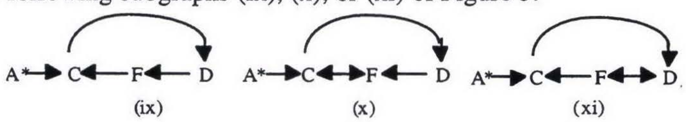


Figure 3

However (ix) contains a cycle. (x) is impossible because C $\leftrightarrow$ F implies C is not an ancestor of F in G(**O**,**S**,**L**), but there is a path C $\rightarrow$ D $\rightarrow$ F in MAG(G(**O**,**S**,**L**)), and hence a directed path from C to F in G(**O**,**S**,**L**). (xi) is impossible because F $\leftrightarrow$ D implies F is not an ancestor of D in G(**O**,**S**,**L**), but there is a path F $\rightarrow$ C $\rightarrow$ D in MAG(G(**O**,**S**,**L**)), and hence a directed path from F to D in G(**O**,**S**,**L**).

Similarly, it can be shown that if there is an edge D $\rightarrow$ C, the only possible subgraphs are (v), (vi), (vii), and (viii). $\therefore$

Lemma 9: If U is a minimal d-connecting path between X and Y given **R** in MAG(G(**O**,**S**,**L**)), U contains the subpath A $*\!\!\rightarrow$ B $\leftarrow\!\!*$ D $*\!\!\rightarrow$ C, and MAG(G(**O**,**S**,**L**)) contains the edge B $\rightarrow$ C, then U contains a unique subpath U(F,C) that is a discriminating path for D.

Proof. We will show that for each $n \geq 1$, if U contains a vertex M such that U(M,D) is of length n, for every vertex Q on U(M,D) except for the endpoints Q is a collider on U(M,D), and for each vertex Q on U(M,D) except possibly for D there is an edge Q $\rightarrow$ C in MAG(G(**O**,**S**,**L**)), then U contains a vertex F such that U(F,D) is of length n + 1, and either U(F,D) is a discriminating path for D, or U contains an edge F $\leftrightarrow$ M, and MAG(G(**O**,**S**,**L**)) contains an edge F $\rightarrow$ C.

By hypothesis, there is a path U(B,D) of length 1 such that every vertex Q between B and D is a collider on U and there is an edge B $\rightarrow$ C in MAG(G(**O**,**S**,**L**)).

Suppose that U contains a vertex M such that U(M,D) is of length n, and for every vertex Q on U(M,D) except for the endpoints Q is a collider on U(M,D) and for each vertex Q on U(M,D) except possibly for D there is an edge Q $\rightarrow$ C in MAG(G(**O**,**S**,**L**)). Because U is a minimal d-connecting path, by Lemma 7 there is an edge F *$\rightarrow$ M on U. The edge between F and M is either F $\rightarrow$ M, F o$\rightarrow$ M, or F $\leftrightarrow$ M. If U(F,C) is not a discriminating path for D then there is an edge between F and C in MAG(G(**O**,**S**,**L**)). By Lemma 7, there are two cases: (i) the edge between F and C is out of C (and hence C $\rightarrow$ F), or (ii) the edge between F and C is F $\rightarrow$ C, and the edge between F and M is F $\leftrightarrow$ M.

If (i), there is no edge F o$\rightarrow$ M because C $\rightarrow$ F is into F. If (i) and the edge between F and M is F $\rightarrow$ M, then there is a cycle F $\rightarrow$ M $\rightarrow$ C and C $\rightarrow$ F. If the edge between F and M is F $\leftrightarrow$ M then there is a contradiction because M is not an ancestor of F but there is a path M $\rightarrow$ C $\rightarrow$ F. It follows that case (ii) holds. Hence if U does not contains a discriminating subpath, then for every subpath of U there is a longer subpath of U. Because U is of finite length, it follows that it contains a discriminating subpath U(M,C) for D.

We will now show that U is unique. Because the edge between B and C is oriented as B $\rightarrow$ C, M lies between X and C. No subpath of U(M,C) is a discriminating path for D because all of the vertices on U(M,C) except for M are adjacent to C. No path containing U(M,C) is a discriminating path for D because M is not adjacent to C. $\therefore$

In MAG(G(**O**,**S**,**L**)), a vertex V is a **hidden vertex** on a discriminating path U if and only if there are vertices X and Y on U such that V is adjacent to X and Y on U, and X and Y are adjacent in MAG(G(**O**,**S**,**L**)).

Lemma 10: In MAG(G(**O**,**S**,**L**)), if U is a minimal d-connecting path between X and Y given **Z**, then there is no pair of distinct vertices B, J such that B is a hidden vertex on the discriminating path U(I,K) for J, and J is a hidden vertex on the discriminating path U(A,C) for B.

Proof. Suppose on the contrary that B is a hidden vertex on U(I,K), and J is a hidden vertex on U(A,C). Because B is hidden on U(I,K), C lies on U(I,K). C $\neq$ K because B lies on U(I,K), the only vertex adjacent to K on U(I,K) is J, and B $\neq$ J. C $\neq$ J because otherwise J is not a hidden vertex on U(A,C). C $\neq$ I because otherwise J, which is on U(A,C) but not equal to B or A is an ancestor of C = I, and hence by repeated applications of Lemma 1 through Lemma 5 there is an inducing path between I and K in G(**O**,**S**,**L**). But by definition of discriminating path I and K are not adjacent in MAG(G(**O**,**S**,**L**)). Hence C $\neq$ I. Because C is on U(I,K) but is not equal to I, J, or K, there is a directed path from C to K. Similarly, there is a directed path from K to C. Hence, there is a directed cycle in MAG(G(**O**,**S**,**L**)), which is a contradiction. $\therefore$

Lemma 11: In a MAG(G(**O**,**S**,**L**)), if U is a minimal d-connecting path between X and Y given **Z**, then there is no triple of distinct hidden vertices X, Y, Z on U such that X is a hidden vertex on the discriminating path for Y on U, Y is a hidden vertex on the discriminating path for Z on U, and Z is between Y and X on U.

Proof. Let U_Y be the discriminating path for Y on U, and similarly for U_Z. Because X is a hidden vertex on the discriminating path for Y on U, every vertex between X and Y is on the discriminating path for Y on U. Hence Z is on the discriminating path for Y on U. Because Z is between Y and X, and neither Y nor X is an endpoint of U_Y, Z is not an endpoint of U_Y; it follows that each of the vertices adjacent to Z on U are also on U_Y. Because Z is a hidden vertex on U, and both of the vertices adjacent to Z are also on the discriminating path for Y on U, Z is a hidden vertex on U_Y. By hypothesis, Y is a hidden vertex on U_Z. But this contradicts Lemma 10.

Lemma 12: In a MAG(G(**O**,**S**,**L**)), if U is a minimal d-connecting path between X and Y given **Z**, then there is no quadruple of distinct hidden vertices A_i, A_{r+1}, A_{i+1}, A_r in that order on U such that A_i is a hidden vertex on the discriminating path for A_{i+1} on U, and A_r is a hidden vertex on the discriminating path for A_{r+1} on U.

Proof. Let U_{i+1} be the discriminating path for A_{i+1} on U, and similarly for U_{r+1}. Suppose contrary to the hypothesis there is a quadruple of distinct hidden vertices A_i, A_{r+1}, A_{i+1}, A_r in that order on U such that A_i is a hidden vertex on the discriminating path for A_{i+1} on U, and A_r is a hidden vertex on the discriminating path for A_{r+1} on U. A_{r+1} is on U_{i+1} because it is between A_i and A_{i+1}, and A_i is on U_{i+1}. A_{r+1} is a hidden vertex on U, and both of the vertices adjacent to A_{r+1} on U are also on U_{i+1}, because A_{r+1} is not an endpoint of U_{i+1}. Hence A_{r+1} is a hidden vertex on U_{i+1}. Similarly, A_{i+1} is a hidden vertex on U_{r+1}. But this contradicts Lemma 10.

Lemma 13: In a MAG(G(**O**,**S**,**L**)), if U is a minimal d-connecting path between X and Y given **Z**, then there is no sequence of length greater than 1 of distinct vertices $\langle A_1,A_2,...,A_n\rangle$ such that for each pair of vertices A_i, A_{i+1} that are adjacent in the sequence, A_i is a hidden vertex on the discriminating path of A_{i+1} on U, and A_n is a hidden vertex on the discriminating path of A_1 on U. (Note that the subscripts of the vertices do not necessarily reflect the order in which they occur on U.)

Proof. Suppose without loss of generality that n is greater than 1, and A_1 is to the right of A_n on U. Let r be the highest index such that A_r is to the right of A_1 if such a vertex exists; otherwise let r = 1. We will now show that A_{r+1} is to the left of A_n. If r = 1, every vertex except A_1 is to the left of A_1, so A_2 is to the left of A_1, and $A_2 \neq A_n$ by Lemma 10. By Lemma 11 then A_2 is not between A_1 and A_n, so it is to the left of A_n. If r ≠ 1, then $A_{r+1} \neq A_n$ by Lemma 11, and A_{r+1} is not between A_1 and A_n by Lemma 12. Hence A_{r+1} is to the left of A_n.

We will now show that some vertex A_{s+1} whose index is greater than r+1 is to the right of A_r. A_{r+2} is not between A_{r+1} and A_r by Lemma 11, and hence not equal to A_n. There are two cases. If A_{r+2} is to the right of A_r, then we are done. Suppose then that A_{r+2} is to the left of A_{r+1}. It follows that there is some vertex with index greater than r+2 (e.g. A_n) on the other side of A_{r+1}. It follows that for some s > r such that A_s and A_{s+1} are on opposite sides of A_{r+1} (where A_s is to the left of A_{r+1}.) A_{s+1} is not between A_{r+1} and A_r by Lemma 12, and hence $A_{s+1} \neq A_n$. So A_{s+1} is to the right of A_r. But this is a contradiction, because r is the highest index such that A_r is to the right of A_1

We will now recursively define the order of a discriminating path for a hidden variable on a minimal d-connecting path. If U is a minimal d-connecting path between X and Y given **Z**, and W is a hidden variable on U such that the discriminating path for W on U contains no hidden variables other than W, then W is a **0-order hidden variable** on U. If U is a minimal d-connecting path between X and Y given **Z**, and W is a hidden variable on U such that the maximum order of any other hidden variable on the discriminating path for W on U is n−1, then W is an **nth-order hidden variable** on U.

Lemma 13 guarantees that this recursive definition is sound, because it guarantees that if U is a minimal d-connecting path between X and Y given **Z** that contains hidden variables, then there is a 0 order hidden variable on U and also that the definition of the order of any hidden variable W on U is not defined in terms of the order of W.

Lemma 14: If there is an edge A *→ B in MAG(G(**O**,**S**,**L**)), then in G(**O**,**S**,**L**) there is an inducing path between A and B that is into B.

Proof. By the definition of a MAG and Lemma 5 there is an inducing path U between A and B in G(**O**,**S**,**L**), and B is not an ancestor of A or **S** in G(**O**,**S**,**L**). Suppose that U is out of B. If there are no colliders on U, then U is a directed path from B to A, and B is an ancestor of A, which is a contradiction. If there is a collider on U, let C be the closest collider to B; C it is an ancestor of B, A, or **S**. If C is an ancestor of B then there is a cycle in G(**O**,**S**,**L**) which is a contradiction. If C is an ancestor of A or **S**, then B is an ancestor of A or **S** which is a contradiction. Hence U is into B. ∴

Lemma 15: If MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, U is a discriminating path between X_1 and Y for F in MAG(G_1(**O**,**S**,**L**)), U' is the path corresponding to U in MAG(G_2(**O**,**S'**,**L'**)), and every vertex (except for the endpoints and possibly F) is a collider on U', then U' is a discriminating path for F in MAG(G_2(**O**,**S'**,**L'**)).

Proof. Suppose that the vertices on U preceding F are X_1, ..., X_n. By definition, in MAG(G_1(**O**,**S**,**L**)) X_1 is not adjacent to Y, X_2 is a collider on U, and X_2 is an unshielded non-collider on the concatenation of $U(X_1,X_2)$ and the edge $X_2 \rightarrow Y$. Hence in MAG(G_2(**O**,**S'**,**L'**)), X_1 is not adjacent to Y, X_2 is a collider on

U', and X_2 is an unshielded non-collider on the concatenation of $U'(X_1,X_2)$ and the edge between X_2 and Y. It follows that the edge between X_2 and Y is oriented as $X_2 \rightarrow Y$ in MAG(G_2(**O**,**S'**,**L'**)).

Suppose for each X_i, $2 \leq i \leq m-1$, in MAG(G_2(**O**,**S'**,**L'**)) X_i is a collider on U', and the edge between X_i and Y is oriented as $X_i \rightarrow Y$. In MAG(G_2(**O**,**S'**,**L'**)), let V' be the concatenation of $U'(X_1,X_{m-1})$ and the edge between X_{m-1} and Y. Every vertex on V' between X_1 and X_m is a collider by hypothesis, and for each X_i between X_1 and X_m, there is an edge $X_i \rightarrow Y$ by hypothesis. Hence V' is a discriminating path for X_m. If V is the corresponding path in MAG(G_1(**O**,**S**,**L**)), V is a discriminating path, and X_m is a non-collider on V. Hence X_m is a non-collider on V'. It follows that the edge between X_m and Y is oriented as $X_m \rightarrow Y$. By induction, U' is a discriminating path for F in MAG(G_2(**O**,**S'**,**L'**)). $\therefore$

Lemma 16: If MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, U is a minimal d-connecting path between X and Y given **Z** in MAG(G_1(**O**,**S**,**L**)), U' is the corresponding path in MAG(G_2(**O**,**S'**,**L'**)), then F is a collider on U if and only if F is a collider on U'.

Proof. If F is not a hidden vertex on U, then because MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, F is not a hidden vertex on U', and by definition F is a collider on U if and only if F is a collider on U'.

Suppose F is a hidden vertex on U. By Lemma 8, MAG(G(**O**,**S**,**L**)) contains one of the subgraphs of type (i) through (viii) in Figure 2. By Lemma 9, U contains a discriminating path U(M,N) for F.

Suppose first that F is a zero order hidden vertex on U. Then all of the vertices on U(M,F) except for the endpoints are unshielded colliders in MAG(G_1(**O**,**S**,**L**)). Because MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, all of the vertices on U(M,F) are unshielded colliders. Hence by Lemma 15, U'(A,B) is a discriminating path in MAG(G_2(**O**,**S'**,**L'**)). It follows that F is a collider on U if and only if F is a collider on U'.

Suppose that for all $0 \leq i < n$, the i^{th} order hidden vertices on U are oriented the same way on U'. Now consider an n^{th} order hidden vertex on U. There is a subpath U(M,N) that is a discriminating path for F. By the induction hypothesis, all of the colliders on U(M,N) are colliders on U'(M,N). Hence by Lemma 15, U'(A,B) is a discriminating path in MAG(G_2(**O**,**S'**,**L'**)). It follows that F is a collider on U if and only if F is a collider on U'. $\therefore$

Lemma 17: If A and B are d-connected given **R** in MAG(G(**O**,**S**,**L**)), then A and B are d-connected given **R** $\cup$ **S** in G(**O**,**S**,**L**).

Proof. Suppose that in MAG(G(**O**,**S**,**L**)) A and B are d-connected given **R** by a minimal d-connecting path U. Then each vertex on U is active given **R**. For each edge X *—* Y in MAG(G(**O**,**S**,**L**)), there is an inducing path between X and Y in G(**O**,**S**,**L**). By Lemma 2, Lemma 3, and Lemma 4, there is a path that d-connects X and Y given **R** $\cup$ **S**\{X,Y} in G(**O**,**S**,**L**). Choose all such d-connecting path for each pair of vertices X and Y adjacent on U; call this collection of d-connecting paths T. If a vertex X is on U, say that X is **active in T given R** $\cup$ **S** whenever either (i) there are vertices C and D on U adjacent to X, there is a path in T between X and C that is into X, there is a path in T between X and D that is into X, and X is an ancestor of **R** $\cup$ **S** in G(**O**,**S**,**L**), or (ii) there is a d-connecting path in T containing X as an endpoint that is out of X, and X is not in **R**. Consider the following three cases.

U contains a subpath C *—o F o—* D, and F is active on U given **R**. Hence there is a path X_1 in T that d-connects C and F given (**R** $\cup$ **S**)\{C,F}, and a path X_2 in T that d-connects F and D given (**R** $\cup$ **S**)\{F,D}. F is not in **R** because F is active on U given **R**, and F is not a collider on U. F is active in T given **R** $\cup$ **S** if X_1 and X_2 collide at F because it is an ancestor of **S** in G(**O**,**S**,**L**), and is active in T if X_1 and X_2 do not collide at F because it is not in **R** $\cup$ **S**, so F is active in T given **R** $\cup$ **S**.

U contains a subpath C *$\rightarrow$ F $\leftarrow$* D, and F is active on U given **R**. It follows that in MAG(G(**O**,**S**,**L**)) F has a descendant in **R**. Hence F has a descendant in **R** in G(**O**,**S**,**L**). By Lemma 14 there is an inducing path between C and F that is into F, and an inducing path between D and F that is into F. It follows from Lemma 2 and Lemma 3 that there is a path X_1 in T that d-connects C and F given (**R** $\cup$ **S**)\{C,F} that is into F, and a path X_2 in T that d-connects F and D given (**R** $\cup$ **S**)\{F,D} that is into F. F is active in T given **R** $\cup$ **S** because X_1 and X_2 collide at F in G(**O**,**S**,**L**), and F is an ancestor of **R** in G(**O**,**S**,**L**).

U contains a subpath C *—* F $\rightarrow$ D, and F is active on U given **R**. (The case where U contains a subpath C $\leftarrow$ F *—* D is analogous.) Because F is active on U given **R**, F is not in **R**. There is a directed path from F to D in $G(\mathbf{O},\mathbf{S},\mathbf{L})$ that does not contain any vertices in **S**. There are two cases. If the directed path contains a member of **R**, then F is an ancestor of **R**, and hence F is active in T given $\mathbf{R} \cup \mathbf{S}$ regardless of whether or not the d-connecting paths collide at F. If the directed path does not contain a member of **R**, the directed path d-connects F and D given $\mathbf{R} \cup \mathbf{S}$ and is out of F. It follows that F is active in T given $\mathbf{R} \cup \mathbf{S}$. It follows from Lemma 1 that there is a path in $G(\mathbf{O},\mathbf{S},\mathbf{L})$ that d-connects A and B given $\mathbf{R} \cup \mathbf{S}$. ∴

Lemma 18: If X and Y are d-connected given $\mathbf{Z} \cup \mathbf{S}$ in $G(\mathbf{O},\mathbf{S},\mathbf{L})$, then X and Y are d-connected given **Z** in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$).

Proof. Suppose that U is a minimal d-connecting path between X and Y given $\mathbf{Z} \cup \mathbf{S}$ in $G(\mathbf{O},\mathbf{S},\mathbf{L})$. We will perform a series of operation which show how to construct a path U' in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) which d-connects A and B given **Z** in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$). The operations are illustrated with Figure 4 ($\mathbf{Z} = O_2,O_3$).

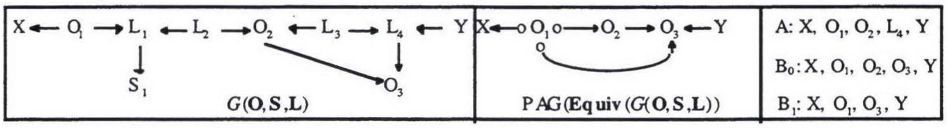


Figure 4

First form the following sequence of vertices. A(0) = X. If A(n−1) ≠ Y, then A(n) is the first vertex on U after A(n−1) such that either it is a non-collider on U and is in **O**, or it is a collider on U and not an ancestor of **S**. The last vertex in the sequence is Y because Y is in **O** and not a collider on U. Suppose the length of the sequence is n, i.e. A(n) = Y.

Note that for $1 \le i \le n$, if A(i) is a collider on U then it is an ancestor of **Z**, because U d-connects X and Y given $\mathbf{Z} \cup \mathbf{S}$.

Suppose A(i) is in **O**, but not a collider on U. Then for $1 < i \le n-1$, either U(A(i−1),A(i)) or U(A(i),A(i+1)) is out A(i). Suppose without loss of generality that U(A(i−1),A(i)) is out of A(i). Then A(i) is an ancestor of **S** or A(i−1) because either U(A(i−1),A(i)) contains no colliders in which case A(i) is an ancestor of A(i−1), or it does contain a collider, in which case the first collider is an ancestor of a member of **S**, and A(i−1) is an ancestor of the first collider. Similarly, if U(A(i),A(i+1)) is out A(i) then A(i) is an ancestor of A(i+1) or **S**. So if A(i) is in **O**, but not a collider on U, then A(i) is an ancestor of A(i−1), A(i+1) or **S**.

Now form the sequence of vertices where for $1 \le i \le n$, $B_0(i)$ = A(i) if A(i) in **O**, and otherwise $B_0(i) = O_i$, where O_i is the first vertex in **O** on a shortest path from A(i) to **Z**. (Such a path exists because U d-connects X and Y given $\mathbf{Z} \cup \mathbf{S}$, and no A(i) that is a collider on U is an ancestor of **S**.) We will now show that for $1 \le i \le n$, there is an inducing path between $B_0(i)$ and $B_0(i+1)$.The path between A(i) and A(i+1) d-connects A(i) and A(i+1) given (**Ancestors**($\{B_0(i),B_0(i+1)\} \cup \mathbf{S}) \cap \mathbf{O}) \cup \mathbf{S})\backslash\{A(i),A(i+1)\}$, because every collider on U(A(i),A(i+1)) is an ancestor of **S**, and no non-collider on U(A(i),A(i+1)) (except for the endpoints) is in $\mathbf{O} \cup \mathbf{S}$. The path from A(i) to O_i (if there is one) d-connects A(i) and O_i given (**Ancestors**($\{B_0(i),B_0(i+1)\} \cup \mathbf{S}) \cap \mathbf{O}) \cup \mathbf{S})\backslash\{A(i),O_i\}$ because by construction it is a directed path that contains no member of **O** except for O_i, and no member of **S**. Similarly, the path from A(i+1) to O_{i+1} (if there is one) d-connects A(i+1) and O_{i+1} given (**Ancestors**($\{B_0(i),B_0(i+1)\} \cup \mathbf{S}) \cap \mathbf{O}) \cup \mathbf{S})\backslash\{A(i+1),O_{i+1}\}$. By Lemma 1 $B_0(i)$ and $B_0(i+1)$ are d-connected given **Ancestors**($\{B_0(i),B_0(i+1)\} \cup \mathbf{S}) \cap \mathbf{O}) \cup \mathbf{S})\backslash \{B_0(i),B_0(i+1)$, and by Lemma 5 there is an inducing path between $B_0(i)$ and $B_0(i+1)$. Because for $1 \le i \le n-1$ there is an inducing path between $B_0(i)$ and $B_0(i+1)$, there is a path B_0 in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) on which $B_0(i)$ is the i^{th} vertex.

If $B_0(i)$ is not a non−collider on U, any edge that contains $B_0(i)$ on any of the paths used to construct the inducing path between $B_0(i)$ and $B_0(i+1)$ or $B_0(i-1)$, is into $B_0(i)$. Hence the inducing path between $B_0(i)$ and $B_0(i+1)$ and the inducing path between $B_0(i)$ and $B_0(i-1)$ are into $B_0(i)$.

If $B_0(i)$ is on U but not a collider on U, then $B_0(i)$ is not in $\mathbf{Z} \cup \mathbf{S}$ because U d-connects X and Y given $\mathbf{Z} \cup \mathbf{S}$. In addition, either $B_0(i)$ is an ancestor of $B_0(i-1)$, $B_0(i+1)$ or **S**, because $B_0(i)$ = A(i), A(i) is an ancestor of A(i−1), A(i+1) or **S**, and A(i−1) is an ancestor of $B_0(i-1)$, and A(i+1) is an ancestor of $B_0(i+1)$. It follows that $B_0(i)$ is not a collider on B_0, and is not in $\mathbf{Z} \cup \mathbf{S}$.

By construction, if $B_0(i)$ is not a non-collider on U, $B_0(i)$ is in **O**, an ancestor of **Z**, and not an ancestor of **S**. However, $B_0(i)$ may be in **Z** but not be a collider on B_0 in MAG($G(\mathbf{O},\mathbf{S},\mathbf{L})$) (if in $G(\mathbf{O},\mathbf{S},\mathbf{L})$ it is an ancestor of either its predecessor or successor on *U*). The following algorithm removes all non-colliders on B_0 which are in **Z**.

k = 0;

Repeat

If there is a triple of vertices $B_k(i-1)$, $B_k(i)$, $B_k(i+1)$ such that the inducing paths between $B_k(i-1)$ and $B_k(i)$, and $B_k(i)$ and $B_k(i+1)$ collide at $B_k(i)$, but $B_k(i)$ is in **Z** and an ancestor of $B_k(i-1)$ or $B_k(i+1)$, form sequence B_{k+1} by removing $B_k(i)$ from the sequence (i.e. for $1 \le j < i$, set $B_{k+1}(j) = B_k(j)$, and for $i \le j \le n-1$, set $B_{k+1}(j) = B_k(j+1)$);

k := k + 1;

until there is no such triple of vertices in the sequence B_k.

At each stage of the algorithm, if there is a triple of vertices $B_k(i-1)$, $B_k(i)$, $B_k(i+1)$ such that the inducing paths between $B_k(i-1)$ and $B_k(i)$, and $B_k(i)$ and $B_k(i+1)$ collide at $B_k(i)$, but $B_k(i)$ is an ancestor of $B_k(i-1)$ or $B_k(i+1)$, then by Lemma 1 through Lemma 5 there is an inducing path between $B_k(i-1)$ and $B_k(i+1)$. Hence for every k and each i, $1 \le i \le$ length of sequence B_k, there is an inducing path between $B_k(i)$ and $B_k(i+1)$. It follows that there is path B_k in MAG(*G*(**O**,**S**,**L**)) such that the i^{th} vertex on the path is $B_k(i)$.

Suppose first that $B_k(i)$ is a non-collider on *U*. We will show $B_k(i)$ is a non-collider on B_k in MAG(*G*(**O**,**S**,**L**)), and not in $\mathbf{Z} \cup \mathbf{S}$. We have already shown this for B_0. In addition, we have shown that if $B_0(i)$ is a non-collider on *U* either $B_0(i)$ is an ancestor of $B_0(i-1)$, $B_0(i+1)$ or **S**. Suppose for $1 \le i \le$ length(B_{k-1}), if $B_{k-1}(i)$ is a non-collider on *U* either $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i-1)$, $B_{k-1}(i+1)$ or **S.** It is an ancestor of $B_k(i-1)$, $B_k(i+1)$ or **S**, unless $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i-1)$, and $B_{k-1}(i-1)$ was removed at the k^{th} step of the algorithm, or $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i+1)$, and $B_{k-1}(i+1)$ was removed at the k^{th} step of the algorithm. Suppose without loss of generality that $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i+1)$, and $B_{k-1}(i+1)$ was removed at the k^{th} step of the algorithm. It follows that $B_{k-1}(i+1)$ is an ancestor of $B_{k-1}(i+2)$, $B_{k-1}(i)$ or **S**. $B_{k-1}(i+1)$ is not an ancestor of $B_{k-1}(i)$ because *G*(**O**,**S**,**L**) is acyclic. Hence, it follows that it is an ancestor of either $B_{k-1}(i+2)$ or **S**. It follows that $B_k(i) \equiv B_{k-1}(i)$ is an ancestor of **S** or $B_{k-1}(i+2) \equiv B_k(i+1)$. Hence $B_k(i)$ is not a collider on B_k, and not in $\mathbf{Z} \cup \mathbf{S}$.

Suppose that $B_k(i)$ is not a non-collider on *U*. If k = 0, then $B_k(i)$ is in **O** and an ancestor of **Z**, and for every other value of k, B_k is a subsequence of B_0, so $B_k(i)$ is in in **O** and an ancestor of **Z** if it is not a non-collider on *U*. We will now show that the inducing paths between $B_k(i)$ and $B_k(i-1)$, and $B_k(i)$ and $B_k(i+1)$ are both into $B_k(i)$. We have already shown that the inducing path between $B_0(i)$ and $B_0(i+1)$ and the inducing path between $B_0(i)$ and $B_0(i-1)$ are into $B_0(i)$. Suppose the same is true for each $B_{k-1}(i)$ that is not a non-collider on *U*. This will also be true for each $B_k(i)$ unless $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i-1)$, and the vertex $B_{k-1}(i-1)$ was removed at the k^{th} step of the algorithm, or $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i+1)$, and the vertex $B_{k-1}(i+1)$ was removed at the k^{th} step of the algorithm. Suppose without loss of generality that $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i+1)$, and $B_{k-1}(i)$ was removed at the k^{th} step of the algorithm. The edge on the inducing path between $B_{k-1}(i)$ and $B_{k-1}(i+1)$ that contains $B_{k-1}(i)$ is into $B_{k-1}(i)$ and so is every edge on the inducing path between $B_{k-1}(i+1)$ and $B_{k-1}(i+2)$ that contains $B_{k-1}(i)$ (because it is in **O**.) $B_{k-1}(i)$ is not on any directed path from $B_{k-1}(i+1)$ to $B_{k-1}(i+2)$ because by hypothesis, $B_{k-1}(i)$ is an ancestor of $B_{k-1}(i+1)$. Since all of the edges that contain $B_{k-1}(i)$ on the paths used to construct the inducing path between $B_{k-1}(i)$ and the vertex $B_{k-1}(i+2)$ are into $B_{k-1}(i)$, the inducing path between $B_{k-1}(i) \equiv B_k(i)$ and $B_{k-1}(i+2) \equiv B_k(i+1)$ is into $B_k(i)$.

If $B_k(i)$ is not a non-collider on *U*, then the inducing paths between $B_k(i)$ and $B_k(i-1)$, and $B_k(i)$ and $B_k(i+1)$ are both into $B_k(i)$. It follows that if the algorithm exits at stage k, each $B_k(i)$ in **Z** that is not a non-collider on *U* is not an ancestor of either $B_k(i-1)$ or $B_k(i+1)$. Hence it is a collider on B_k, and is a member of **O** that is an ancestor of **Z**.

Hence each vertex on B_k is active, and B_k d-connects X and Y given **Z** in MAG(*G*(**O**,**S**,**L**)). $\therefore$

Lemma 19: If MAG(*G*(**O**,**S**,**L**)) contains A *→ B → H, and an edge A *—* H, then (i) the edge between A and H is into H, and (ii) if A *—* H has a different orientation at A than A *→ B, then the edges are oriented as A → H and A ↔ B.

Proof. Because there is an edge into H, A *—* H is not oriented as A *—o H. Because there is an edge A *→ B, B is not an ancestor of A. If the edge between A and H is oriented as A ← H, then B is an ancestor of A, which is a contradiction. Hence the edge between A and H is into H. If the edge A *→ H has a different orientation at A than A *→ B, then either A ↔ B and A → H, or A → B and A ↔ H. If A → B and A ↔ H, then A is an ancest or of H (A → B → H) which contradicts A ↔ H. ∴

Lemma 20: If MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, *U* is a minimal d-connecting path between X and Y given **Z** in MAG(G_1(**O**,**S**,**L**)), F is a collider on *U*, H is an ancestor of **Z** and there is an F → H edge in MAG(G_1(**O**,**S**,**L**)), then there is an F → H edge in MAG(G_2(**O**,**S'**,**L'**)).

Proof. If F is a collider on *U*, by Lemma 16 both MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) contain X_0 *→ F ←* Y_0. If there is no edge between X_0 and H in MAG(G_1(**O**,**S**,**L**)), then F is an unshielded non-collider on X_0 *→ F *—* H in MAG(G_1(**O**,**S**,**L**)), and hence F is an unshielded non-collider on X_0 *→ F *—* H in MAG(G_2(**O**,**S'**,**L'**)). It follows that F *—* H is oriented as F o→ H or F → H in MAG(G_2(**O**,**S'**,**L'**)). It is not oriented as F o→ H because F is a collider on U and hence not an ancestor of **S**. It follows that the edge is oriented as F → H. Similarly, if there is no edge between Y_0 and H, F *—* H is oriented as F → H in MAG(G_2(**O**,**S'**,**L'**)). Suppose then that X_0 and Y_0 are both adjacent to H.

There is a vertex N on *U* between X and F that is either (i) not adjacent to H, or (ii) the edge between N and H is not into H, or (iii) *U* agrees with the concatenation of *U*(X,N) and the edge between N and H at N (since X itself trivially satisifes condition (iii) if it is adjacent to H.) Similarly, there is a vertex M on U between Y and F that is either (i) not adjacent to H, or (ii) the edge between M and H is not into H, or (iii) *U* agrees with the concatenation of *U*(Y,M) and the edge between M and H at M. Let X_{n+1} be the closest such vertex on *U* to F, and Y_{o+1} be the closest such vertex on *U* to F. Let X_n through X_0 be the be the vertices on *U* between X_{n+1} and F.

We will now show by induction that for $0 \le i \le n$, the edge between X_i and its successor on *U* is into X_i, and there is an edge X_i → H in MAG(G_1(**O**,**S**,**L**)). If $X_0 = X_{n+1}$, then it is trivially true (because there are no vertices between X_{n+1} and F). Suppose X_0 is between X_{n+1} and F. We have already shown that there is an edge X_0 *→ F, and an edge F → H. By definition of X_{n+1}, the edge between X_0 and H disagrees with the edge between X_0 and F at X_0. By Lemma 19 it follows that the edge between X_0 and F is X_0 → F, and the edge between X_0 and F is into X_0. Suppose for $0 \le i \le m-1$ that the edge between X_i and its successor on *U* is into X_i, and the edge between X_i and H is oriented as X_i → H. If X_m is between X_{n+1} and H, by definition of X_{n+1}, X_m is adjacent to H and *U* disagrees with the concatenation of *U*(X,X_{m-1}) and the edge between X_{m-1} and H at X_{m-1}; hence the edge between X_m and X_{m-1} is X_m *→ X_{m-1}. By lemma 19, the edge between X_m and X_{m-1} is into X_m, and there is an edge X_m → H. Hence every vertex X_i between X_{n+1} and F is a collider on *U*, and there is an edge X_i → H. Similarly, every vertex Y_i between Y_{o+1} and F is a collider on *U*, and there is an edge Y_i → H.

If X_{n+1} is adjacent to H, then by Lemma 19 X_{n+1} *—* is into H, and by definition of X_{n+1}, *U* agrees with the concatenation of *U*(X,X_{n+1}) and the edge between X_{n+1} and H at X_{n+1}. Similarly, if If Y_{o+1} is adjacent to H, then by Lemma 19 Y_{o+1} *—* is into H, and by definition of Y_{o+1}, *U* agrees with the concatenation of *U*(Y_{o+1},Y) and the edge between Y_{o+1} and H at Y_{o+1}. In that case, the concatenation of *U*(X,X_{n+1}), the edge between X_{n+1} and H, the edge between H and Y_{o+1}, and U(Y_{o+1},Y) d-connects X and Y given **Z**, and *U* is not minimal. This is a contradiction. It follows that either X_{n+1} or Y_{o+1} is not adjacent to H. Suppose without loss of generality that it is the former.

If X_{n+1} is not adjacent to H there is a path *V* between X_{n+1} and H consisting of the concatenation of *U*(X_{n+1},F) with the edge between F and H. Let *V'* be the corresponding path in MAG(G_2(**O**,**S'**,**L'**)). By definition, *V* is a discriminating path for F in MAG(G_1(**O**,**S**,**L**)), and F is a non-collider on the path. Because all of the colliders on *V* are also colliders on *U* which is minimal, by Lemma 16 they are all colliders on *V'*. By Lemma 15, *V'* is a discriminating path for F. Hence F is a non-collider on *V'* in MAG(G_2(**O**,**S'**,**L'**)), and the edge between F and H is oriented as F → H in MAG(G_2(**O**,**S'**,**L'**)). ∴

Lemma 21: If MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, *U* is a minimal d-connecting path between X and Y given **Z** in MAG(G_1(**O**,**S**,**L**)), A is a collider on *U* and B is a member of **Z** that is the endpoint of a shortest path D from A to **Z** in MAG(G_1(**O**,**S**,**L**)), then B is a descendant of A in MAG(G_2(**O**,**S'**,**L'**)).

Proof. Let D' be the path corresponding to D in MAG(G_2(**O**,**S'**,**L'**)), and U' be the path corresponding to U in MAG(G_2(**O**,**S'**,**L'**)). By Lemma 16, A is a collider on U'. By Lemma 20, the first edge on D' is out of A. If D' is not a directed path in MAG(G_2(**O**,**S'**,**L'**)) then it contains a collider F. Because D does not contain a collider, and MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, F is a shielded collider in MAG(G_2(**O**,**S'**,**L'**)). It follows then that in MAG(G_1(**O**,**S**,**L**)) there is a vertex E and D contains a subpath E → F → H, and an edge between E and H. The edge between E and H is not oriented as H → E, else MAG(G_1(**O**,**S**,**L**)) contains a cycle; it is nor oriented as E ↔ H because E is an ancestor of H in MAG(G_1(**O**,**S**,**L**)); it is neither E *—o H nor E o—* H, because A then would be an ancestor of **S** in G_1(**O**,**S**,**L**) and hence not a collider on U. Hence it is oriented as E → H. But then D is not a shortest directed path from A to a member of **Z**, contrary to our assumption. ∴

Theorem 1: DAGs G_1(**O**,**S**,**L**) and G_2(**O**,**S'**,**L'**) are conditional independence equivalent if and only if MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders.

Proof. If X and Y are adjacent in MAG(G_1(**O**,**S**,**L**)) but not in MAG(G_2(**O**,**S'**,**L'**)), then for some subset **V** of **O**, X and Y are d-separated given **V** ∪ **S** in G_2(**O**,**S'**,**L'**), but not d-separated given **V** ∪ **S'** in G_2(**O**,**S'**,**L'**). If C *→ F ←* D is an unshielded collider in MAG(G_1(**O**,**S**,**L**)) but not in MAG(G_2(**O**,**S'**,**L'**)), then every set that d-separates C and D in MAG(G_1(**O**,**S**,**L**)) does not contain F, but every set that d-separates C and D in MAG(G_2(**O**,**S'**,**L'**)) does contain F. If U is a discriminating path between X and Y for F in MAG(G_1(**O**,**S**,**L**)) and the corresponding path U' is a discriminating path for F in MAG(G_2(**O**,**S'**,**L'**)), and F is a collider on U but not on U', then by Lemma 6 there is a set **Z** that contains F that d-separates X and Y in MAG(G_2(**O**,**S'**,**L'**)) but not in MAG(G_1(**O**,**S**,**L**)).

If MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same basic colliders, then by Lemma 16 and Lemma 21, MAG(G_1(**O**,**S**,**L**)) and MAG(G_2(**O**,**S'**,**L'**)) have the same d-separation relations. By Lemma 17 and Lemma 18, X and Y are d-separated given **R** in MAG(G_1(**O**,**S**,**L**)) if and only if X and Y are d-separated given **R** ∪ **S** in G_1(**O**,**S**,**L**). Similarly, X and Y are d-separated given **R** in MAG(G_2(**O**,**S'**,**L'**)) if and only if X and Y are d-separated given **R** ∪ **S'** in G_2(**O**,**S'**,**L'**). It follows that G_1(**O**,**S**,**L**) and G_2(**O**,**S'**,**L'**) are conditional independence equivalent. ∴

Bibliography

Aho, A., Hopcroft, J., Ullman D., (1974) The Design and Analysis of Computer Algorithms, Addison-Wesley: Reading, MA.

Frydenberg, M., (1990), The Chain Graph Markov Property, Scandinavian Journal of Statistics, 17, 333-353.

Lauritzen, S., Dawid, A., Larsen, B., Leimer, H., (1990) Independence properties of directed Markov fields, Networks, 20, 491-505.

Pearl, J., (1988). Probabilistic Reasoning in Intelligent Systems, Morgan Kaufman: San Mateo, CA.

Richardson, T. (1994b). Properties of Cyclic Graphical Models. MS Thesis, Carnegie Mellon University.

Richardson, T. (1996). A discovery algorithm for directed cyclic graphs. Uncertainty in Artificial Intelligence, Proceedings, 12th Conference, Morgan Kaufman, CA.

Spirtes, P., Verma, T. (1992) Equivalence of Causal Models with Latent Variables. Technical Report CMU-PHIL-33, Department of Philosophy, Carnegie Mellon University, October, 1992.

Spirtes, P., Glymour, C., and Scheines, R., (1993) Causation, Prediction, and Search, (Springer-Verlag Lecture Notes in Statistics 81, New York).

Spirtes, P., Meek, C., and Richardson, T. (1995). Causal Inference in the Presence of Latent Variables and Selection Bias, in Proceedings of the Eleventh Conference on Uncertainty in *Artificial Intelligence*, ed. by Philippe Besnard and Steve Hanks, Morgan Kaufmann Publishers, Inc., San Mateo.

Verma, T. and Pearl, J. (1990), Equivalence and synthesis of causal models, in Proc. Sixth Conference on Uncertainty in AI. Association for Uncertainty in AI, Inc., Mountain View, CA.