Automated Discovery of Linear Feedback Models[1]

by

Thomas Richardson and Peter Spirtes

1. Introduction

The introduction of statistical models represented by directed acyclic graphs (DAGs) has proved fruitful in the construction of expert systems, in allowing efficient updating algorithms that take advantage of conditional independence relations (Pearl, 1988, Lauritzen *et al.* 1993), and in inferring causal structure from conditional independence relations (Spirtes and Glymour, 1991, Spirtes, Glymour and Scheines, 1993, Pearl and Verma, 1991, Cooper, 1992). As a framework for representing the combination of causal and statistical hypotheses, DAG models have shed light on a number of issues in statistics ranging from Simpson's Paradox to experimental design (Spirtes, Glymour and Scheines, 1993). The relations of DAGs with statistical constraints, and the equivalence and distinguishability properties of DAG models, are now well understood, and their characterization and computation involves three properties connecting graphical structure and probability distributions: (i) a local directed Markov property, (ii) a global directed Markov property, (iii) and factorizations of joint densities according to the structure of a graph (Lauritzen, *et al.*, 1990).

Recursive structural equation models are one kind of DAG model. However, non-recursive structural equation models are not DAG models, and are instead naturally represented by directed *cyclic* graphs in which a finite series of edges representing influence leads from a vertex representing a variable back to that same vertex. Such graphs have been used to model feedback systems in electrical engineering (Mason, 1953, 1956), and to represent economic processes (Haavelmo, 1943, Goldberger, 1973). In contrast to the acyclic case, almost nothing general is known about how directed cyclic graphs (DCGs) represent conditional independence constraints, or about their equivalence or identifiability properties, or about characterizing classes of DCGs from conditional

[1] Research for this paper was supported by the National Science Foundation through grant 9102169 and the Navy Personnel Research and Development Center and the Office of Naval Research through contract number N00014-93-1-0568. We are indebted to Clark Glymour, Richard Scheines, Christopher Meek, and Marek Druzdel for helpful conversations.

independence relations or other statistical constraints. This paper addresses all of these issues. The issues turn on how the relations among properties (i), (ii) and (iii) essential to the acyclic case generalize—or fail to generalize—to directed cyclic graphs and associated families of distributions. It will be shown that when DCGs are interpreted by analogy with DAGs as representing functional dependencies with independently distributed noises or "error variables," the equivalence of the fundamental global and local Markov conditions characteristc of DAGs no longer holds, even in linear systems. For linear systems associated with DCGs with independent errors or noises, a characterisation of conditional independence constraints is obtained, and it is shown that the result generalizes in a natural way to systems in which the error variables or noises are statistically dependent.

We also present a correct polynomial time (on sparse graphs) discovery algorithm for linear cyclic models that contain no latent variables. This algorithm outputs a representation of a class of non-recursive linear structural equation models given observational data as input. Under the assumption that all conditional independencies found in the observational data are true for structural reasons rather than because of particular parameter values, the algorithm discovers causal features of the structure which generated the data. A simple modification of the algorithm can be used as a decision procedure (whose runtime is polynomial in the number of vertices) for determining when two directed graphs (cyclic or acyclic) entail the same set of conditional independence relations.

The remainder of this paper is organized as follows: Section 2 defines relevant mathematical ideas and gives some necessary technical results on DAGs and DCGs. Section 3 obtains results for non-recursive linear structural equations models. Section 4 describes a discovery algorithm. Section 5 describes an algorithm for deciding when two graphs (cyclic or acyclic) entail the same set of conditional indepenence relations (or in the linear case entail the same zero partial correlations), and Section 6 describes some open research problems. All proofs are in Section 7.

2. Directed Graphs and Probability Distributions

A **directed acyclic graph** (DAG) *G* with a set of vertices **V** can be given two distinct interpretations. (We place sets of variables and defined terms in boldface.) On the one hand, such graphs can be used to represent causal relations between variables, where an edge from A to B in *G* means that A is a direct cause of B relative to **V**. A **causal graph** is a DAG given such an interpretation.

On the other hand, a DAG with a set of vertices **V** can also represent a set of probability measures over **V**. 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. (A glossary of graph theoretic terminology is given in Section 7. 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*.

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). Variants of probabilistic DAG models were introduced in the 1980's in Wermuth (1980), Wermuth and Lauritzen (1983), Kiiveri, Speed, and Carlin (1984), Kiiveri and Speed (1982), and Pearl (1988). [2] In Section 4 we will present assumptions which link the two interpretations of directed graphs.

Pearl(1988) defines a global directed Markov property that has been shown to be equivalent to the local directed Markov property for DAGs, and can be used to calculate the consequence of the local directed Markov property. (See e.g. Lauritzen *et al.* 1990.[3]) Several preliminary notions are required. Vertex X is a **collider** on an acyclic undirected path *U* in directed graph *G* if and only if there are two adjacent edges on *U* directed into X (e.g. A → X ← B). Every other vertex on *U* is a **non-collider** on *U*. In a directed graph *G*, if X and Y are not in **Z**, then an acyclic undirected path *U* **d-connects** X and Y given **Z** if and only if every collider on *U* has a descendant in **Z**, and no non-collider on *U* is in **Z**. For three disjoint sets **X**, **Y**, and **Z**, **X** and **Y** are **d-connected** given **Z** in *G* if and only if there is a path *U* that d-connects some X in **X** to some Y in **Y** given **Z**. For three disjoint sets **X**, **Y**, and **Z**, **X** and **Y** are **d-separated** given **Z** in *G* if and only of **X** is not d-connected to **Y** given **Z**. A probability distribution P satisfies the global directed Markov property for directed graph *G* if and only if for any three disjoint sets of variables

---

[2]It is often the case that some further restrictions are placed on the set of distributions represented by a DAG. For example, one could also require the Minimality Condition, i.e. that for any distribution P represented by *G*, P does not satisfy the local directed Markov Condition for any proper subgraph of *G*. This condition, and others are discussed in Pearl(1988) and Spirtes, Glymour, and Scheines(1993).
[3] However, in Section 3 we show that the local and global directed Markov properties are not equivalent for cyclic directed graphs.

X, **Y**, and **Z**, if **X** is d-separated from **Y** given **Z** in *G*, then **X** is independent of **Y** given **Z** in P.

The following theorems relate the global directed Markov property to factorizations of a density function. Denote a density function over **V** by *f*(**V**), where for any subset **X** of **V**, *f*(**X**) denotes the marginal of *f*(**V**). If *f*(**V**) is the density function for a probability measure over a set of variables **V** and An(**X**) is the set of ancestors of members of **X** in directed graph *G*, say that *f*(**V**) **factors according to directed graph** *G* with vertices **V** if and only if for every subset **X** of **V**,

$$f(\mathrm{An}(\mathbf{X})) = \prod_{\mathbf{V}\in\mathrm{An}(\mathbf{X})} g_{\mathrm{V}}(\mathrm{V}, \mathrm{Parents}(\mathrm{V}))$$

where g_{V} is a non-negative function. The following result was proved in Lauritzen *et al.* (1990).

Theorem 1: If **V** is a set of random variables with a probability measure P that has a density function *f*(**V**), then *f*(**V**) factors according to DAG *G* if and only if P satisfies the global directed Markov property for *G*.

As in the case of acyclic graphs, the existence of a factorization according to a cyclic directed graph *G* does entail that a measure satisfies the global directed Markov property for *G*. The proof given in Lauritzen *et al.* (1990) for the acyclic case carries over essentially unchanged to the cyclic case. (Lauritzen *et al.* use a different definition of d-separation that in Section 7 is shown to be equivalent to Pearl's in both the cyclic and the acylic case.)

Theorem 2: If **V** is a set of random variables with a probability measure P that has a density function *f*(**V**) and *f*(**V**) factors according to directed (cyclic or acyclic) graph *G*, then P satisfies the global directed Markov property for *G*.

However, unlike the case of acyclic graphs, if a probability measure over a set of variable **V** satisfies the global directed Markov property for cyclic graph *G* and has a density function *f*(**V**), it does not follow that *f*(**V**) factors according to *G*.

The following weaker result relating factorization of densities and the global directed Markov property does hold for both cyclic and acyclic directed graphs.

Theorem 3: If **V** is a set of random variables with a probability measure P that has a positive density function *f*(**V**), and P satisfies the global directed Markov property for directed (cyclic or acyclic) graph *G*, then *f*(**V**) factors according to *G*.

3. Non-recursive Linear Structural Equation Models

The problem considered in this section is to investigate the generalization of the Markov properties to linear, non-recursive structural equation models. First we must relate the social scientific terminology to graphical representations, and clarify the questions.

The variables in a structual equation model (SEM) can be divided into two sets, the "error" variables and the "substantive" variables. Corresponding to each substantive variable X_i is an equation expressing X_i as a *linear* function of the direct causes of X_i plus a unique error variable ε_i. (We will not consider non-linear models.) 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 and X_2 is a direct cause of X_3. The structural equations[4] in Figure 1 are common to both S_1 and S_2.

$$X_1 = \varepsilon_1$$

$$X_2 = \beta_1 X_1 + \varepsilon_2$$

$$X_3 = \beta_2 X_2 + \varepsilon_3$$

Figure 1: Structural Equations for SEMs S_1 and S_2

where β_1 and β_2 are free parameters ranging over real values, and ε_1, ε_2 and ε_3 are error variables. In addition suppose that ε_1, ε_2 and ε_3 are distributed as multivariate normal. In S_1 we will assume that the correlation between each pair of distinct error variables 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 $\{\beta_1, \beta_2\}$ and $\mathbf{P}$ is the set of variances of the error variables. We will use $\Sigma_{\mathbf{S1}}(\theta_1)$ to denote the covariance matrix parameterized by the vector θ_1 for model S_1, and

[4] We realize that it is slightly unconventional to write the trivial equation for the exogenous variable X_1 in terms of its error, but this serves to give the error variables a unified and special status as providing all the exogenous sources of variation for the system.

occasionally leave out the model subscript if the context makes it clear which model is being referred to. If all the pairs of error variables 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 $\theta = <\beta$, **P'**>, where β is the set of linear coefficients $\{\beta_1,\beta_2\}$ and **P'** is the set of variances of the error variables and the correlation between ε_2 and ε_3. If the correlations between any of the error variables in a SEM are not fixed at zero, we will call it a SEM with **correlated errors**.[5]

If the coefficients in the linear equations are such that the substantive variables are a unique linear function of the error variables alone, the set of equations is said to have a **reduced form**. A linear SEM with a reduced form also determines a joint distribution over the substantive variables. We will consider only linear SEMs which have coefficients for which there is a reduced form, all variances and partial variances among the substantive variables are finite and positive, and all partial correlations among the substantive variables are well defined (e.g. not infinite).

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 $X_1 \rightarrow X_2 \rightarrow X_3$, because X_1 is the only substantive variable that occurs on the right hand side of the equation for X_2, and X_2 is the only substantive variable that appears on the right hand side of the equation for X_3. We generally do not include error variables in the causal graph associated with a SEM unless the errors are correlated. When the distinction is relevant to the discussion, we enclose measured variables in boxes, latent variables in circles, and leave error variables unenclosed.

[5]We do not consider SEMs with other sorts of constraints on the parameters, e.g., equality constraints.

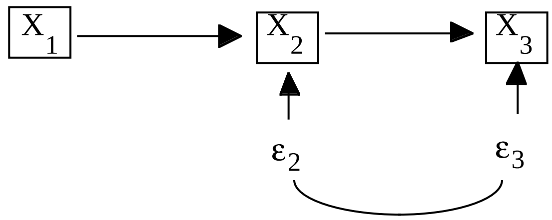

Figure 2. SEM S_2 with correlated errors

The typical path diagram that would be given for S_2 is shown in Figure 2. This is not strictly a directed graph because of the curved line between error variables ε_2 and ε_3, which indicates that ε_2 and ε_3 are correlated. It is generally accepted that correlation is to be explained by some form of causal connection (see section 4.1 below), and accordingly we assume that curved lines are an ambiguous representation of a causal connection.

Interpreted causally, the error variable for a variable X represents all causes of X other than the substantive variables explicitly included in the model. Thus the error variable is really an additive amalgamation of perhaps hundreds of other variables that are left unmeasured. To say that two error variable ε_2 and ε_3 are correlated and have an unknown causal connection is to say that one or more of the variables V_i that make up ε_2 is identical with, or causally connected to one or more of the variables W_j that make up ε_2. Whether V_i is a cause of W_j, or W_j is a cause of V_i, or there is a latent variable C that is a cause of both V_i and W_j (Figure 3), or some combination of these, in all cases there is a latent common cause of the substantive variables X_2 and X_3 for which ε_2 and $\varepsilon_3$are the error variables. Thus we will convert SEMs with correlated errors into SEMs without correlated errors by adding a latent common cause (which possibly represents an amalgam of many latents) of the appropriate substantive variables and replacing the previously correlated error variables with uncorrelated ones. (This method of handling correlated errors is also used in Koster 1995.) We illustrate this process in Figure 3.

Let ε_2 be a linear combination of the set of variables **V**, and ε_3 be a linear combination of the set of variables **W**. (We do not preclude the possibility that **V** and **W** overlap.) In Figure 3, we indicate that ε_2 is a function of the variables in **V** with braces. The variables in **V** and **W** can be re-partitioned into three disjoint subsets (one each for ε_2’, ε_3’, and T) such that no variable in any of the subsets is causally connected to a variable in either of the other subsets. By the Causal Independence assumption (introduced in section 4.1) ε_2’, ε_3’, and T are independent.

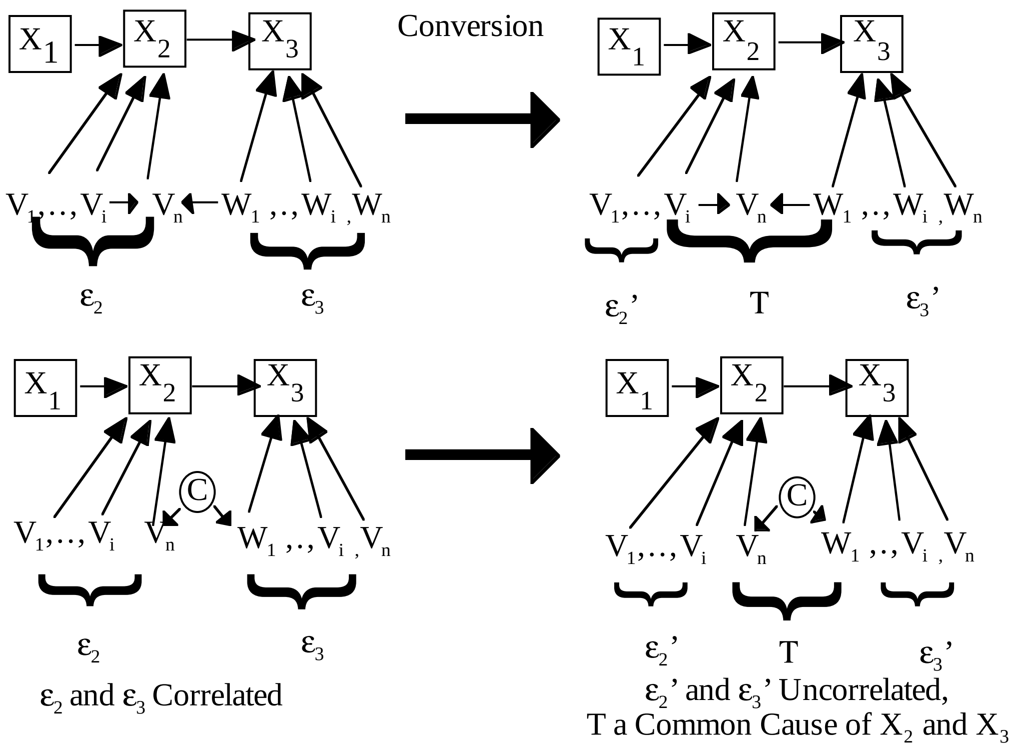

Figure 3. Correlated to Uncorrelated Errors

Applying this procedure to S_2 (Figure 2) results in a model S_2' which we can associate with a directed graph (Figure 4). S_2 and S_2' represent the same causal relations, and for every covariance matrix $\Sigma_{\mathbf{s2}}(\theta)$, there is a covariance matrix $\Sigma_{\mathbf{s2'}}(\theta') = \Sigma_{\mathbf{s2}}(\theta)$, and vice-versa. Henceforth, we will consider only SEMs with uncorrelated errors. However the theorems in this section can be applied to SEMs with correlated errors by using this transformation technique. A SEM is said to be **recursive** (an RSEM) if its directed graph is acyclic; otherwise it is **non-recursive**.[6]

[6] Note that this use of cyclic directed graphs to represent feedback processes represents an extension of the causal interpretation of directed graphs.

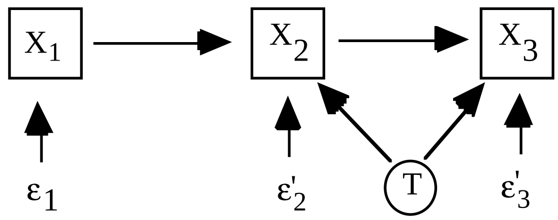

Figure 4. S_2': Correlated Errors Replaced by Latent Common Cause

A linear SEM (or its corresponding DAG) containing disjoint sets of variables **X**, **Y**, and **Z** **linearly entails** that **X** is independent of **Y** given **Z** if and only if **X** is independent of **Y** given **Z** for all values of the linear coefficients not fixed at zero, and all distributions of the exogenous variables in which they are jointly independent. Let $\rho_{XY.\mathbf{Z}}$ be the partial correlation of X and Y given **Z**. A linear SEM (or its corresponding DAG) containing X, Y, and **Z**, where $X \neq Y$ and X and Y are not in **Z**, **linearly entails** that $\rho_{XY.\mathbf{Z}} = 0$ if and only if $\rho_{XY.\mathbf{Z}} = 0$ for all values of the linear coefficients not fixed at zero and all distributions of the exogenous variables in which they are jointly independent. It follows from Kiiveri and Speed (1982) that if the error variables are jointly independent, then any distribution that forms a linear, recursive SEM with a directed graph *G* satisfies the local directed Markov property for *G*. One can therefore apply d-separation to the DAG in a linear, recursive SEM to compute the conditional independencies and zero partial correlations it linearly entails. The d-separation relation provides a polynomial (in the number of vertices) time algorithm for deciding whether a given vanishing partial correlation is linearly entailed by a SEM with a given DAG.

Linear non-recursive structural equation models (linear SEMs) are commonly used in the econometrics literature to represent feedback processes that have reached equilibrium.[7] Corresponding to a set of non-recursive linear equations is a cyclic graph, as the following example from Whittaker (1990) illustrates.

$$X_1 = \varepsilon_{X1}$$
$$X_2 = \varepsilon_{X2}$$
$$X_3 = \beta_{31}X_1 + \beta_{34}X_4 + \varepsilon_{X3}$$
$$X_4 = \beta_{42}X_2 + \beta_{43}X_3 + \varepsilon_{X4}$$

ε_{X1}, ε_{X2}, ε_{X3}, ε_{X4} are jointly independent and normally distributed

---

[7]Cox and Wermuth (1993), Wermuth and Lauritzen(1990) and (indirectly) Frydenberg(1990) consider a class of linear models they call *block recursive*. The block recursive models overlap the class of SEMs, but they are neither properly included in that class, nor properly include it. Frydenberg (1990) presents necessary and sufficient conditions for the equivalence of two block recursive models.

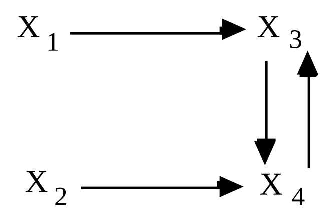

Figure 5: Example of Non-recursive SEM

Theorem 4, Theorem 5, and Theorem 6 state that the set of conditional independence relations and zero partial correlations entailed by a SEM can be read off of the d-separation relations in the associated graph, even in the case of cyclic graphs. (Theorem 4 was independently proved by J. Koster in Koster (1995).)

Theorem 4: The probability measure P over the substantive variables of a linear SEM *L* (recursive or non-recursive) with jointly independent error variables satisfies the global directed Markov property for the directed (cyclic or acyclic) graph *G* of *L*, i.e. if **X**, **Y**, and **Z** are disjoint sets of variables in *G* and **X** is d-separated from **Y** given **Z** in *G*, then **X** and **Y** are independent given **Z** in P.

Theorem 5: In a linear SEM *L* with jointly independent error variables and directed (cyclic or acyclic) graph *G* containing disjoint sets of variables **X**, **Y** and **Z**, if **X** is not d-separated from **Y** given **Z** in *G* then *L* does not linearly entail that **X** is independent of **Y** given **Z**.

Applying Theorem 4 and Theorem 5 to a linear SEM with the directed graph in Figure 5, the conditional independence relations linearly entailed are just X_1 is independent of X_2, and X_1 is independent of X_2 given X_3 and X_4.

Theorem 6: In a linear SEM *L* with jointly independent error variables and (cyclic or acyclic) directed graph *G* containing substantive variables X, Y and **Z**, where X ≠ Y and **Z** does not contain X or Y, X is d-separated from Y given **Z** in *G* if and only if *L* linearly entails that $\rho_{XY.\mathbf{Z}} = 0$.

As in the acyclic case, d-separation provides a polynomial time procedure for deciding whether a linear SEM with a cyclic graph linearly entails a conditional independence or vanishing partial correlation.

In DAGs the global directed Markov property entails the local directed Markov property, because a variable V is d-separated from its non-parental non-descendants given its parents. However, this is not always the case in cyclic graphs. For example, in Figure 5,

X_4 is not d-separated from its non-parental non-descendant X_1 given its parents X_2 and X_3, so the local directed Markov property does not hold.[8]

4. The Discovery Problem

Suppose that we are given data sampled from a population whose causal structure is correctly described by some non-recursive structural equation model **M**. Is it possible to discover the causal graph of **M** from the data, or at least recover some features of the causal graph from the data? In Spirtes *et al.* (1995) the problem of discovering features of the causal graph is considered under the assumption that it is acyclic, but that there may be latent variables (i.e. there may be unmeasured variables that are the direct cause of at least two measured variables.) Here we will consider the problem of discovering features of the causal graph under the assumption that it may be cyclic, but there are no latent variables. Future research is needed on the problem of discovering the causal graph when it may be cyclic *and* there may be latent variables.

In order to make inferences about causal relations from a sample distribution, it is necessary to introduce some axioms that relate probability distributions to causal relations. The two assumptions that we will make are the Causal Independence and Causal Faithfulness Assumptions, described in the next two subsections.

4.1. The Causal Independence Assumption

The most fundamental assumption relating causality and probability that we will make is the following:

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 uncorrelated.

This assumption allows us to draw a *causal* conclusion from *statistical* data and lies at the foundation of the theory of randomized experiments. If the value of A is randomized, the experimenter knows that the randomizing device is the sole cause of A. Hence the experimenter knows B did not cause A, and that there is no third variable which causes both A and B. This leaves only two alternatives: either A causes B or it does not. If A and

---

[8] We are indebted to C. Glymour for pointing out that the local Markov condition fails in Whittaker's model. Indeed, there is *no* acyclic graph (even with additional variables) that linearly entails all and only conditional independence relations linearly entailed by Figure 5, although the directed cyclic graph of Figure 5 is equivalent to one in which the edges from X_1 to X_3 and X_2 to X_4 are replaced, respectively, by edges from X_1 to X_4 and from X_2 to X_3.

B are correlated in the experimental population, the experimenter concludes that A does cause B, which is an application of the Causal Independence Assumption.

The Causal Independence Assumption entails that if two error variables, such as ε_2 and ε_3 in Figure 2 are correlated there is a latent common cause of X_2 and X_3 responsible for the correlation. In other words, when X_2 and X_3 have correlated errors, we assume that the distribution over X_2 and X_3 is the marginal of some other distribution including a finite number of latent causes of X_2 and X_3 in which the error variables are uncorrelated, as in Figure 4. Since we are making the assumption that there are no latent variables, it follows that the error variables of the causal graph are uncorrelated. The correctness of the d-separation criterion for deciding which partial correlations are linearly entailed to be zero by a SEM with an associated graph *G* then follows necessarily from Theorem 6.

4.2. The Faithfulness Assumption

In addition to the zero partial correlations that are entailed for *all* linear parameterizations of a graph, there may be zero partial correlations that hold only for some *particular* parameterizations of a graph. For example, suppose Figure 6 is the directed graph of a SEM that describes the relations among Tax Rate, the Economy, and Tax Revenues.

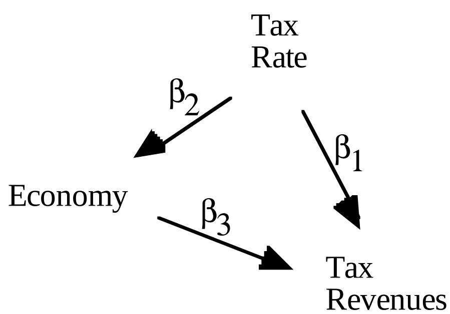

Figure 6. Economic Model

In this case there are no vanishing partial correlation constraints entailed for all values of the free parameters. But if $\beta_1 = -(\beta_2 \times \beta_3)$, then Tax Rate and Tax Revenues are uncorrelated. The SEM postulates a direct effect of Tax Rate on Revenue (β_1), and an indirect effect through the Economy ($\beta_2 \times \beta_3$). The parameter constraint indicates that these effects *exactly* offset each other, leaving no total effect whatsoever. In such a case we say that the population is **unfaithful** to the graph of the causal structure that generated it. A distribution is **faithful** to a directed graph *G* if each partial correlation that is zero in the distribution is entailed to be zero by *G*.

Causal Faithfulness Assumption: If a distribution P is generated by a SEM with associated graph *G*, then **A** is independent of **B** given **C** in P only if *G* linearly entails that **A** is independent of **B** given **C**.

The faithfulness assumption limits the SEMs considered to those in which population constraints are entailed by structure, not by particular values of the parameters. If one assumes faithfulness, then if A and B are *not* d-separated given **C**, then $\rho_{A,B.\mathbf{C}} \neq 0$, (because it is not linearly entailed to equal zero for all values of the free parameters.) Faithfulness should not be assumed when there are deterministic relationships among variables, or equality constraints upon free parameters, since either of these can lead to violations of the assumption. Some form of the assumption of faithfulness is used in every science, and amounts to no more that the belief that an improbable and unstable cancellation of parameters does not hide real causal influences. When a theory cannot explain an empirical regularity save by invoking a special parameterization, most scientists are uneasy with the theory and look for an alternative.

It is also possible to give a personalist Bayesian argument for assuming faithfulness. For any graph, the set of linear parameterizations of the graph that lead to violations of linear faithfulness are Lebesgue measure zero. Hence any Bayesian whose prior over the parameters is absolutely continuous with Lebesgue measure, assigns a zero prior probability to violations of faithfulness. Of course, this argument is not relevant to those Bayesians who place a prior over the parameters that is not absolutely continuous with Lebesgue measure and assigns a non-zero probability to violations of faithfulness.

The assumption of faithfulness guarantees the asymptotic correctness of the Cyclic Causal Discovery (CCD) algorithm described in Section 4.4. It does *not* guarantee that on samples of finite size this algorithm is reliable.

Given the Causal Independence Assumption, an assumption of no latent variables, a linearity assumption, and the Causal Faithfulness assumption, it follows that in a distribution P generated by a causal structure represented by a directed graph *G*, $\rho_{XY.Z} = 0$ if and only if X is d-separated from Y given Z in *G*. So if we can perform statistical tests of zero partial correlations then we can use that information to draw conclusions about the d-separation relations in *G*, and then to reconstruct as much information about *G* as possible. Henceforth we will speak of reconstructing features of *G* from d-separation relations, and from zero partial correlation interchangeably, since given our assumptions, these are equivalent. (The algorithm is correct for any distribution for which a d-separation oracle is available, but the only case we know of in which such an oracle is available is the linear case.)

Of course the number of distinct d-separation relations grows exponentially with the number of variables in the graph. Therefore it is important to discover the features of *G* from a subset of the set of all d-separation relations. The CCD algorithm that we describe below chooses the subset of d-separation relations that it needs to reconstruct features of *G* as it goes along. Therefore we assume that it has access to a **d-separation oracle** that correctly answers questions about d-separation relations in *G*. In practice, of course, the oracle is some kind of statistical test of the hypothesis that a particular partial correlation is zero in a population that satisfies the global Markov and faithfulness properties for *G*.

4.3. Output Representation – Partial Ancestral Graphs (PAGs)

In general, it is not possible to reconstruct a unique graph *G* given information only about its d-separation relations, because there may be more than one graph that contains exactly the same set of d-separation relations. Two directed graphs G, G^* are said to be d-separation **equivalent** if they both have the same set of d-separation relations. The set of directed graphs d-separation equivalent to a given graph *G* is denoted by **Equiv**(*G*). Richardson(1994b) and Richardson(1995) present a polynomial-time algorithm for determining when two graphs are d-separation equivalent to each other; a simpler algorithm is presented in Section 5. (Note that there is a stronger sense of equivalence, which we will call linear statistical equivalence between two graphs which holds when every distribution described by a linear parameterization of one graph can also be described by a linear parameterization of the other graph, and vice-versa. In the acyclic case it is known that d-separation equivalence is equivalent to linear statistical equivalence, but it is not known if this is so for cyclic graphs.)

The members of **Equiv**(*G*) always have certain features in common. We now introduce the formalism with which we will represent features common to all graphs in **Equiv**(*G*) for some fixed *G*. A partial ancestral graph (PAG) is an extended graph, consisting of a set of vertices **V**, a set of edges between vertices, and a set of edge-endpoints, two for each edge, drawn from the set {o, –, >}. In addition pairs of edge endpoints may be connected by underlining, or dotted underlining. A partial ancestral graph for *G* contains partial information about the ancestor relations in *G*, namely only those ancestor relations common to all members of **Equiv**(*G*). In the following definition, which provides a semantics for PAGs we use '*' as a meta-symbol indicating the presence of any one of {o, –, >}.

Partial Ancestral Graphs (PAGs)[9]

Ψ is a PAG for Directed Cyclic Graph *G* with vertex set **V**, if and only if

(i) There is an edge between A and B in *G* if and only if A and B are d-connected in **Ψ** given any subset $\mathbf{W} \subseteq \mathbf{V} \backslash \{A,B\}$.

(ii) If there is an edge in **Ψ** out of A (not necessarily into B), i.e. A–* B, then A is an ancestor of B in every graph in **Equiv**(*G*).

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

(iv) If there is an underlining A*—*B*—*C in **Ψ** then B is an ancestor of (at least one of) A or C in every graph in **Equiv**(*G*).

(v) If there is an edge from A to B, and from C to B, (A—> B<— C), then the arrow heads at B are joined by dotted underlining, thus A—>B<—C, only if in every graph in **Equiv**(*G*) B is not a descendant of a common child of A and C.

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

Observe that condition (i) differs from the other five conditions in providing necessary *and* sufficient condtions on **Equiv**(*G*) for a given symbol, in this case an edge, to appear in a PAG. The other five conditions merely state necessary conditions. For this reason there are in fact many different PAGs for a graph *G*. Although they all have the same edges, they do not necessarily have the same endpoints. Some of the PAGs provide more information than others about causal structure, e.g. they have fewer 'o's at the end of edges.[10]

If **Ψ** is a PAG for Directed Cyclic Graph *G*, we also say that **Ψ** represents *G*. Since every clause in the definition refers only to **Equiv**(*G*), it follows that if **Ψ** is a PAG for Directed Cyclic Graph *G*, and $G^* \in$ **Equiv**(*G*), then **Ψ** is also a PAG for G^*. This is not surprising since, as the output of the discovery algorithm we present below, the PAG is designed to

[9] The extended graphs which we introduce here - Partial Ancestral Graphs - use a superset of the set of symbols used by Partially Oriented Inducing Path Graphs (POIPGs) descaribed in Spirtes *et al.* (1993) but the *graphical* interpretation of the orientations given to edges is different. However, it has been shown in Sprites *et al.* (1996) that a POIPG can be interpreted directly as a PAG. A direct corollary is that PAGs can be used to represent the d-separation equivalence class for directed *acyclic* graphs with *latent* variables. It is an open question whether or not the set of symbols is sufficiently rich to allow us to represent d-separation classes of cyclic graphs with latent variables.

[10] If one PAG for a graph *G* has a '>' at the end of an edge, then every other PAG for the same graph either has a '>' or a 'o' in that location. Similarly if one PAG for a graph *G* has a '–' at the end of an edge then every other PAG for the same graph either has a '–' or an 'o' in that location.

represent features common to all graphs in the d-separation equivalence class. However, some PAGs (providing less information) may also represent graphs from different d-separation equivalence classes. However, any PAG output by the discovery algorithm we present provides sufficient information so as to ensure that all of the graphs that it represents lie in one d-separation equivalence class; hence any PAG output by the discovery algorithm can be thought of as representing a unique d-separation equivalence class.

The set of features described by a PAG is rich enough to enable us to distinguish between any two d-separation equivalence classes, i.e. there is some feature common to all graphs in one d-separation equivalence class that is not true of all graphs in the other d-separation equivalence class, and this difference can be expressed by a difference in the PAGs representing those d-separation equivalence classes.

Example:
Suppose *G* is as follows:

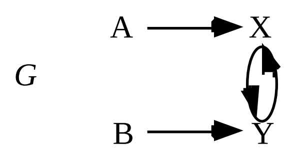

Figure 7

In this case it can be shown that **Equiv**(*G*) contains (only) two graphs:

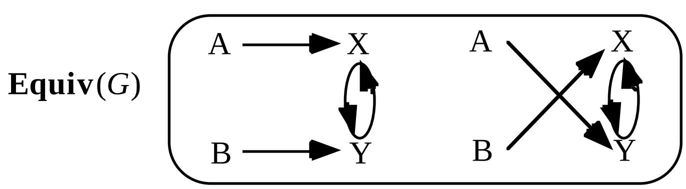

Figure 8

The PAG which the CCD algorithm outputs given as input an oracle for deciding conditional independence facts in *G*, is:

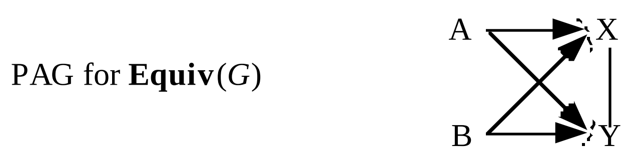

Figure 9

Observe that the PAG tells us the following facts about **Equiv**(*G*):[11]

(a) X is an ancestor of Y, and Y is an ancestor of X in every graph in **Equiv**(*G*).
(b) In no graph in **Equiv**(*G*) is X or Y an ancestor of A or B.
(c) In every graph in **Equiv**(*G*) both A and B are ancestors of X and Y.

Note that not every edge in the PAG appears in every graph in **Equiv**(*G*). This is because an edge in the PAG indicates only that the two variables connected by the edge are d-connected given any subset of the other variables. In fact it is possible to show something stronger, namely that if there is an edge between two vertices in a PAG, then there is some graph represented by the PAG in which that edge is present.[12]

This example is atypical in that the PAG given by the algorithm contains no 'o' endpoints; however it shows how much information a PAG may provide. Notice that the following are also PAGs for *G* though they are less informative.

Other PAGs for **Equiv**(*G*)

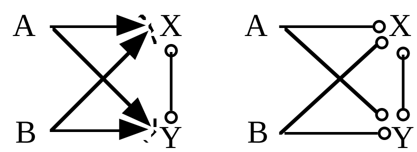

Figure 10

The CCD algorithm we describe does not always give the most informative PAG for a given graph *G* in that there may be features common to all graphs in the d-separation equivalence class which are not captured by the PAG that the algorithm outputs. In this sense the algorithm is not complete. However, the algorithm is 'd-separation complete' in the sense that if the d-separation oracles for two different graphs cause the algorithm to produce the same PAG as output then the two graphs are d-separation equivalent.

Two definitions are required to state the algorithm. Two vertices, X and Y in a PAG are **p-adjacent** if there is an edge between them, X*—*Y in the PAG.[13] For PAG **Ψ**, **Adjacencies**(Ψ,X) is a function giving the set of variables Y s.t. there is an edge X*—*Y in **Ψ**. **Ψ** is a dynamic object in the algorithm that changes as the algorithm progresses, and hence **Adjacencies**(Ψ,X) also changes as the algorithm progresses. A trace of the algorithm on a simple example is given in section 4.6.

---

[11]This is not an exhaustive list. For example, the presence of the dotted line connecting the arrowheads on the A —>X, and B—>X edges, tells us that in no graph in Equiv(*G*) are both of these edges present. Likewise with the dotted line connecting the arrowheads of the B—>Y, and A—>Y edges.
[12]See footnote 10.
[13]Here as elsewhere '*' as a meta-symbol indicating any of the three ends -, o, >.

4.4. The Cyclic Causal Discovery (CCD) Algorithm

The overall strategy for discovery is shown in Figure 11.

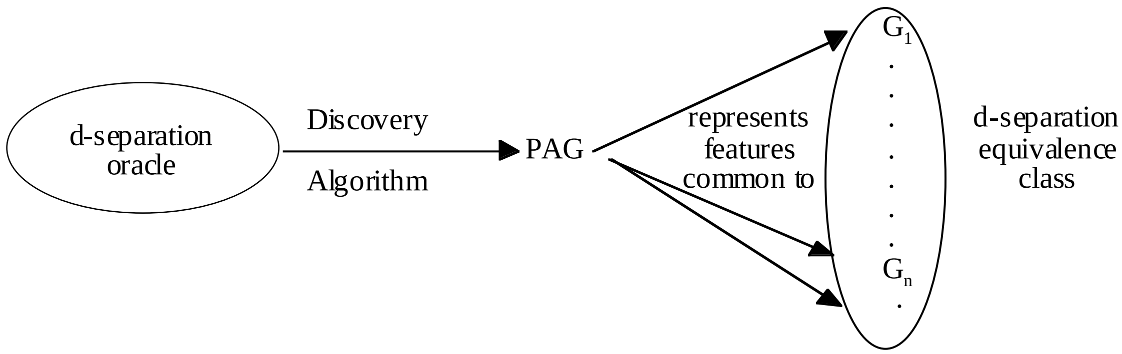

Figure 11

CCD Algorithm

Input: An oracle for answering questions of the form: "Is X d-separated from Y given set **Z**, (X,Y$\notin$ **Z**) in graph *G*?"

Output: A PAG for *G*.

¶**A** a) Form the complete undirected PAG Ψ, i.e. for each pair of variables A and B, Ψ contains the edge A o—o B.

b) n = 0.
repeat
  repeat
    select an ordered pair of variables X and Y that are p-adjacent in Ψ such that the number of vertices in **Adjacencies**(Ψ,X)\{Y} is greater than or equal to n;
    repeat
      select a subset **S** of **Adjacencies**(Ψ,X)\{Y} with n vertices;
      if X and Y are d-separated given **S** delete edge X o—o Y from Ψ and set **Sepset**(X,Y) = **S** and **Sepset**(Y,X) = **S**;
    until every subset **S** of **Adjacencies**(Ψ,X)\{Y} with n vertices has been selected or some subset **S** has been found for which X and Y are d-separated given **S**;
  until all ordered pairs of p-adjacent vertices X and Y such that **Adjacencies**(Ψ,X)\{Y} has greater than or equal to n vertices have been selected;
  n = n + 1;
until for each ordered pair of p-adjacent vertices X, Y, **Adjacencies**(Ψ,X)\{Y} has less than n vertices.

¶B. For each triple of vertices A,B,C such that the pair A,B and the pair B,C are each p-adjacent in Ψ but the pair A, C are not p-adjacent in Ψ, orient A∗—∗B∗—∗C as A—>B<—C if and only if B is not in **Sepset**<A,C>; orient A∗—∗B∗—∗C as A∗—∗B∗—∗C if and only if B is in **Sepset**<A,C>.

¶C. For each triple of vertices <A,X,Y> in Ψ such that

(a) A is not p-adjacent to X or Y in Ψ,

(b) X and Y are p-adjacent in Ψ,

(c) X ∉ **Sepset**<A,Y>

(i) If **Sepset** < A, Y > $\subsetneq$ **Sepset** < A, X > then orient X o–∗Y as X<—Y

(ii) Else if **Sepset**<A,X> is not a subset of **Sepset**<A,Y>, then orient X o–∗Y as X<—Y if A and X are d-connected given **Sepset**<A,Y>

¶D. For each vertex V in Ψ form the following set: X∈ **Local**(Ψ,V) if and only if X is p-adjacent to V in Ψ, or there is some vertex Y such that X—>Y<—V in Ψ. (Local(Ψ,V) is calculated once for each vertex V and does not change as the algorithm progresses.)

m = 1.

repeat

repeat

select an ordered triple <A,B,C> such that A—>B<—C but A and C are not p-adjacent, and **Local**(Ψ,A)\(**Sepset**<A,C> ∪ {B,C}) has greater than or equal to m vertices.

repeat

select a set **T** ⊆ **Local**(Ψ,A)\(**Sepset**<A,C> ∪ {B,C}) with m vertices, and test if A and C are d-separated given **T** ∪ **Sepset**<A,C> ∪ {B} then orient the triple A—>B<—C as A—>B<—C, and record **T** ∪ **Sepset**<A,C> ∪ {B} in **SupSepset**<A,B,C>.

until every subset **T** ⊆ **Local**(Ψ,A)\(**Sepset**<A,C> ∪ {B,C}) with m vertices has been selected or a d-separating set for A and C has been recorded in **SupSepset**<A,B,C>.

until all triples such that A—>B<—C, (i.e. not A—>B<—C), A and C are not p-adjacent, and **Local**(Ψ,A)\(**Sepset**<A,C> ∪ {B,C}) have greater than or equal to m vertices have been selected.

m = m +1.

until each ordered triple <A,B,C> such that A—>B<—C but A and C are not p-adjacent, is such that **Local**(Ψ,A)\(**Sepset**<A,C> ∪ {B}) has fewer than m vertices.

¶E. If there is a quadruple <A,B,C,D> of distinct vertices such that

(i) A—>B<—C in Ψ

(ii) A—>D<—C or A—>D<—C in Ψ

(iii) B and D are p-adjacent in Ψ

then orient B∗–oD as B—>D in Ψ if D is not in **SupSepset**<A,B,C>
else orient B∗–oD as B∗—D in Ψ if D is in **SupSepset**<A,B,C>

¶F. For each quadruple <A,B,C,D> of distinct vertices such that

(i) A—>B<—C in Ψ

(ii) D is not p-adjacent to both A and C in Ψ

if A and D are d-connected given **SupSepset**<A,B,C> ∪ {D}, then orient B∗—oD as B —>D in Ψ

4.5. Soundness and Completeness

Theorem 7: (Soundness) Given as input an oracle for d-separation relations in the (cyclic or acyclic) directed graph *G*, then the output is a PAG Ψ for *G*.

Theorem 7 is proved by showing that each section of the algorithm makes correct inferences from the answers of the d-separation oracle applied to *G.*

In practice, an approximation to a d-separation oracle can be implemented as a statistical test that the corresponding partial correlation vanishes. As the sample size increases without limit, if the significance level of the statistical test is systematically lowered, then the probabilities of both Type I and Type II error for the test approach zero, so that the statistical test is correct with probability one. Of course, this does not guarantee that the CCD algorithm as implemented is reliable on realistic sample sizes. The reliability of the algorithm depends upon the following factors:

1. Whether the Causal Independence Assumption holds (i.e. there are no latent variables).
2. Whether the Causal Faithfulness Assumption holds.
3. Whether the distributional assumptions made by the statistical tests hold.
4. The power of the statistical tests against alternatives.
5. The significance level used in the statistical tests.

In the future, we will test the sensitivity of the algorithm to these factors on simulated data.

Theorem 8: (d-separation Completeness) If the CCD algorithm, when given as input d-separation oracles for the graphs G_1, G_2 produces as output PAGs Ψ_1, Ψ_2 respectively, then Ψ_1 is identical to Ψ_2 if and only if G_1 and G_2 are d-separation equivalent, i.e. $G_2 \in$ **Equiv**(G_1) and vice versa.

The proof is based on the characterization of d-separation equivalence in Richardson (1994b). (It follows directly from Theorem 7 that if G_1 and G_2 are d-separation equivalent then Ψ_1 is identical to Ψ_2.)

4.6. Trace of CCD Algorithm

The following illustrates the operation of the algorithm given as input a d-separation oracle for the following graph:

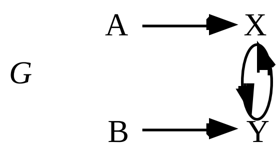

Figure 12

Initial PAG Ψ:

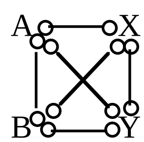

Figure 13

Section ¶A:

Since A and B are d-separated given the empty set, the algorithm removes the edge between A and B and records **Sepset**<A,B> = **Sepset**<B,A> = ∅. This is the only pair of vertices that are d-separated given a subset of the other variables. Hence after ¶A Ψ is as follows:

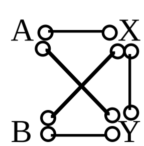

Figure 14

Section ¶B

Since X∉ **Sepset**<A,B> and Y∉ **Sepset**<A,B>, Ao–oXo–oB and Ao–oYo–oB are oriented respectively as A–>X<–B and A–>Y<–B. The state of Ψ at the end of ¶B is shown in Figure 15.

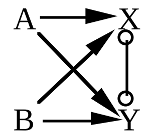

Figure 15

Section ¶C No orientations are performed in this case.

Section ¶D

Since A and B are d-separated given {X,Y}, the algorithm records **SupSepset**<A,X,B> = **SupSepset**<A,Y,B> = {X,Y}, and it orients A–>X<–B as A–>X<–B, and A–>Y<–B as A–>Y<–B. The state of PAG Ψ after ¶D is shown in Figure 16.

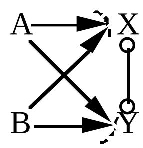

Figure 16

Section ¶E

The quadruple <A,B,X,Y> is such that (i) A–>X<–B, (ii) A–>Y<–B, (iii) X and Y are p-adjacent, thus it satisfies the conditions in section ¶E. Since Y∈ **SupSepset**<A,X,B>, the edge Xo–oY is oriented as Y—oX. Since X∈ **SupSepset**<A,Y,B>, this edge is further oriented as Y—X.

Section ¶F – Performs no orientations in this case, hence the PAG that is output is:

PAG Ψ

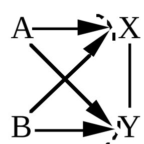

Figure 17

4.7. Complexity of CCD Algorithm

Let $\text{MaxDegree}(G) = \underset{Y \in V}{\text{Max}} \left| \{ X \mid Y \leftarrow X, \text{ or } X \leftarrow Y \text{ in } G \} \right|$,
and $\text{MaxAdj}(G) = \underset{Y \in V}{\text{Max}} \left| \{ X \mid X \text{ is p-adjacent to Y in any PAG for } G \} \right|$

The number of d-separation tests performed by Step ¶A of the CCD algorithm will, in a worst case, be bounded as follows:

$$\begin{array}{c}\text{Total number of}\\ \text{oracle consultations in ¶A}\end{array} \leq 2 \cdot \binom{n}{2} \sum_{i=0}^{k} \binom{n-2}{i} \leq \frac{(k+1)n^2(n-2)^{k+1}}{k!}.$$

where n = number of vertices in G, k = MaxAdj(G). Since MaxAdj(G) $\leq$ (MaxDegree(G))^2, with MaxDegree(G) = r this step is $O(n^{r^2+3})$. It should be stressed that even as a worst case complexity bound this is a very loose one; the bound presumes that there is a graph in which every pair of non-adjacent vertices in the graph are only d-separated given all vertices adjacent to one of them.

Step ¶B performs no additional tests of d-separation.

Step ¶C performs at most one d-separation test for each triple satisfying the conditions given. Thus this step is O(n^3).

In a worst case the number of tests of d-separation that Step ¶D performs is bounded by

$$\begin{array}{c}\text{Total number of}\\ \text{oracle consultations in ¶D}\end{array} \leq \binom{n}{3}\sum_{i=0}^{m}\binom{n-3}{i} \leq \frac{(m+1)n^3(n-3)^{m+1}}{m!}$$

where m = $\operatorname{Max}_{Y\in\mathbf{V}}|\{X \mid \text{Local}(\Psi,X)\}$ in ¶D. Since m $\leq$ (MaxDegree(G))^2, this step is $O(n^{r^2+4})$. Again this is a loose bound.

Step ¶E performs no tests of d-separation, while step ¶F performs at most one test for each quadruple satisfying the conditions. Hence this step is O(n^4), (though in many graphs there may be very few quadruples satisfying all four conditions).

5. d-separation Equivalence

Since the CCD algorithm is d-separation complete, the orientation rules in the algorithm may be used to construct a d-separation equivalence algorithm. We present an algorithm that, given as input a Directed Cyclic or Acyclic graph *G* will produce as output the same PAG that the CCD algorithm outputs given only a d-separation oracle for *G*. However, this algorithm, unlike the CCD algorithm, runs in time polynomial in the number of vertices, even if MaxDegree(*G*) is not kept fixed. Thus this algorithm can be used to test for d-separation equivalence of two graphs in polynomial time. Let Children(**X**) be the set of children of members of **X** in *G*.

Cyclic PAG-from-Graph Algorithm

Input: Directed Cyclic or Acyclic graph *G*

Output: The CCD PAG Ψ for *G*.

¶a Form the complete undirected PAG Ψ, whch has an edge o–o between every pair of vertices in the vertex set **V.**

For each ordered pair of vertices <A,B> form the following sets:

$\mathbf{S_{A,B}}$ = Children(A) $\cap$ An({A,B})
$\mathbf{T_{A,B}}$ **=** (Parents($\mathbf{S_{A,B}}$ $\cup$ {A}) $\cup$ $\mathbf{S_{A,B}}$)\(Descendants(Children(A)$\cap$Children(B)) $\cup$ {A,B})
For each ordered pair <A,B>:
If A and B are d-separated given $\mathbf{T_{A,B}}$ then record $\mathbf{T_{A,B}}$ in **Sepset**<A,B> and **Sepset** <B,A> and remove the edge Ao–oB from $\boldsymbol{\Psi}$.
else if A and B are d-separated given $\mathbf{T_{B,A}}$ then record $\mathbf{T_{B,A}}$ in **Sepset**<A,B> and **Sepset** <B,A> and remove the edge Ao–oB from $\boldsymbol{\Psi}$.

¶b For each triple of vertices A,B,C such that the pair A,B and the pair B,C are each p-adjacent in $\boldsymbol{\Psi}$ but the pair A, C are not p-adjacent in $\boldsymbol{\Psi}$, orient A∗—∗B∗—∗C as A—>B<—C if and only if B is not in **Sepset**<A,C>; orient A∗—∗B∗—∗C as A∗—$\underline{*B*}$—∗C if and only if B is in **Sepset**<A,C>.

¶c For each triple of vertices <A,X,Y> in $\boldsymbol{\Psi}$ such that

(a) A is not p-adjacent to X or Y in $\boldsymbol{\Psi}$

(b) X and Y are p-adjacent in $\boldsymbol{\Psi}$

(c) X $\notin$ **Sepset**<A,Y>

Orient X o–∗Y as X<—Y if A and X are d-connected given **Sepset**<A,Y>

¶d For each triple <A,B,C> or <C,B,A> such that A—>B<—C, A and C are not p-adjacent, form the following set:

$\mathbf{Q_{A,B,C}}$ = Children(A) $\cap$ An({A,B,C})
$\mathbf{R_{A,B,C}}$ **=** (Parents($\mathbf{Q_{A,B,C}}$ $\cup$ {A}) $\cup$ $\mathbf{Q_{A,B,C}}$)\(Descendants(Children(A)$\cap$Children(C)) $\cup$ {A,C})
If A and C are d-separated given $\mathbf{R_{A,B,C}}$ $\cup$ {B} then orient A—>B<—C as A—$\underline{>B<}$—C, and record $\mathbf{R_{A,B,C}}$ $\cup$ {B} in **SupSepset**<A,B,C>.

¶e If there is a quadruple <A,B,C,D> of distinct vertices such that

(i) A—$\underline{>B<}$—C in $\boldsymbol{\Psi}$,

(ii) A—>D<—C or A—$\underline{>D<}$—C in $\boldsymbol{\Psi}$,

(iii) B and D are p-adjacent in $\boldsymbol{\Psi}$,

then orient B∗–oD as B—>D in $\boldsymbol{\Psi}$ if D is not in **SupSepset**<A,B,C>
else orient B∗–oD as B∗—D in $\boldsymbol{\Psi}$ if D is in **SupSepset**<A,B,C>.

¶f For each quadruple <A,B,C,D> of distinct vertices such that

(i) A—>B<—C in Ψ

(ii) D is not p-adjacent to both A and C in Ψ, and

if A and D are d-connected given **SupSepset**<A,B,C> ∪ {D}, then orient B*—oD as B —>D in Ψ.

We do not include the proof that this algorithm is correct, but it is very similar to the proof that the CCD algorithm itself is correct. The main difference between the two algorithms lies in the fact the CCD algorithm must search for the Sepset and SupSepset sets, testing many different candidates, whereas the PAG from graph algorithm is faced with the much simpler task of constructing these sets, given the graph itself.

Since, by Theorem 8, given two graphs G_1, G_2 as input, the CCD algorithm will produce the same PAG as output if and only if G_1 and G_2 are d-separation equivalent, the algorithm given above provides an algorithm for deciding the d-separation equivalence of two directed graphs. Moreover the algorithm is of complexity $O(n^7)$ where n is the number of vertices in the graph. In this respect this algorithm is significantly faster than the procedure presented in Richardson (1994b) which was $O(n^9)$.

In addition, if a directed cyclic graph G is provided as input to the PAG-from-graph algorithm, then it is also possible to tell from the execution of the algorithm, whether or not there is an directed acyclic graph that is d-separation equivalent to G: If steps ¶c–¶f perform no orientations then there is a directed acyclic graph d-separation equivalent to G. This follows from the fact that the combination of d-separation relations that the rules in ¶c–¶f require are not entailed by any directed acyclic graph. (See Richardson 1994, 1994b).

6. Conclusion

These results raise a number of interesting questions whose answers may be of practical importance. Are there other parameterizations of directed cyclic graphs which entail the global Markov condition? Is there a polynomial algorithm for determining when two arbitrary directed graphs (cyclic or acyclic) linearly entail the same set of conditional independence relations over a common subset of variables **O**? As we have seen there are correct, polynomial time algorithms for inferring features of sparse directed graphs (cyclic or acyclic) from a probability distribution when there are no latent common causes. There are similarly correct, but not polynomial time, algorithms for inferring features of directed acyclic graphs from a probability distribution even when there may be latent common causes (see Spirtes, 1992, Spirtes, Glymour and Scheines, 1993, and Spirtes, Meek, and Richardson 1995). Are there comparable algorithms for inferring features of directed graphs (cyclic or acyclic) from a probability distribution even when there may be latent common causes?

7. Proofs

A **directed graph** is an ordered pair of a finite set of vertices **V**, and a set of directed edges **E**. A directed edge from A to B is an ordered pair of distinct vertices <A,B> in **V** (depicted as A $\rightarrow$ B) in which A is the **tail** of the edge and B is the **head**; the edge is **out of** A and **into** B, and A is **parent** of B and B is a **child** of A; also A and B are **adjacent**. A sequence of edges $<E_1,...,E_n>$ in a directed graph G is an **undirected path** if and only if there exists a sequence of vertices $<V_1,...,V_{n+1}>$ such that for $1 \leq i \leq n$ either $<V_i,V_{i+1}> = E_i$ or $<V_{i+1},V_i> = E_i$ and $E_i \neq E_{i+1}$. A sequence of edges $<E_1,...,E_n>$ in a directed graph G is a **directed path** if and only if there exists a sequence of vertices $<V_1,...,V_{n+1}>$ such that for $1 \leq i \leq n$ $<V_i,V_{i+1}> = E_i$. A (directed or undirected) path U is **acyclic** if no vertex occurring on an edge in the path occurs more than once. 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. A directed graph is **acyclic** if and only if it contains no directed cyclic paths.[14]

7.1. Proof of Theorem 3

Some of the proofs are simplified by using a graphical relation (which we will call "Lauritzen d-separation") shown in Lauritzen *et al.* (1990) to be equivalent to Pearl's d-separation relation defined in Section 2. Several preliminary definitions are needed to define Lauritzen d-separation. An **undirected graph** is an ordered pair of a finite set of vertices **V**, and a set of undirected edges **E**. An undirected edge between A and B is an unordered pair of distinct vertices {A,B} in **V**. A sequence of edges $<E_1,...,E_n>$ in an undirected graph H is an **undirected path** if and only if there exists a sequence of vertices $<V_1,...,V_{n+1}>$ such that for $1 \leq i \leq n$ $\{V_i,V_{i+1}\} = E_i$ and $E_i \neq E_{i+1}$. Let $G(\mathbf{X})$ be the subgraph of directed graph G that contains only vertices in **X**, with an edge from A to B in $G(\mathbf{X})$ if and only if there is an edge from A to B in G. Moral(G) **moralizes** a directed graph G if and only if Moral(G) is an undirected graph with the same vertices as G, and a pair of vertices X and Y are adjacent in Moral(G) if and only if either X and Y are adjacent in G, or they have a common child in G. In an undirected graph H, if **X**, **Y**, and **Z** are disjoint sets of vertices, then **X** is separated from **Y** given **Z** if and only if every undirected path between a member of **X** and a member of **Y** contains a member of **Z**. If **X**, **Y** and **Z** are disjoint sets of variables, **X** and **Y** are **Lauritzen d-separated** given **Z** in

[14] An undirected path is often defined as a sequence of vertices rather than a sequence of edges. The two definitions are essentially equivalent for acyclic directed graphs, because a pair of vertices can be identified with a unique edge in the graph. However, a cyclic graph may contain more than one edge between a pair of vertices. In that case it is no longer possible to identify a pair of vertices with a unique edge.

a directed graph G just when **X** and **Y** are separated given **Z** in Moral(G(An(**X** $\cup$ **Y** $\cup$ **Z**))). We will show that Pearl d-separation and Lauritzen d-separation are equivalent even in cyclic graphs.

Lemma 1 states conditions under which a set of 'short' d-connecting paths may be put together to form a single d-connecting path. Since some of the vertices in the proofs are defined as satisfying certain properties in the graph, if A and B are vertices, we write A≡ B if and only if A and B are different names for the same vertex. If there is an undirected path U containing vertices A and B in directed graph G, and there is only one subpath of U between A and B, then U(A,B) is the subpath of U between A and B.

Lemma 1: (Richardson 1994b)

In a directed (cyclic or 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 $\forall i$, $0 \leq i \leq n$, $X_i \neq X_{i+1}$ i.e. 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. both 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**.

Proof: Let U' be the concatenation of all of the paths in $\mathcal{T}$ in the order of the sequence R. U' may not be acyclic because it may contain vertices more than once. In particular it may contain the endpoints A and B more than once. Let U^* be a subsection of U' which begins with A, and ends with B, and has no occurences of A or B in between.

We now form the path U by removing all (undirected) cycles from U^*. The process here is simply as follows: If $Y_i = Y_j$, then we remove the subpath associated with the sequence of vertices $\langle Y_i, \ldots Y_j \rangle$. Carry out this process until there are no repetitions among the Y_i. Let us then rename the sequence of vertices associated with U as $\langle V_0 \ldots V_p \rangle$.

We will call an edge in U containing a given vertex V_i, an **endpoint edge**, if V_i is in the sequence R and the edge containing V_i is either the first or last edge on a path in $\mathcal{T}$ between V_i and its predecessor or successor in R; otherwise the edge is an **internal edge**.

It now remains to show that U is a d-connecting path.

First we prove that if $V_i \in \mathbf{Z}$ then V_i is a collider on U:

Any edge in U is either an internal edge or an endpoint edge. If V_i occurrs on a path between two vertices in R, then since the path is d-connecting given $\mathbf{Z}\backslash$\{the endpoints\}, V_i is a collider on this path. Hence any internal edge between V_i and some other vertex is into V_i.

If V_i occurs in R, then by condition c(ii) any paths in $\mathcal{T}$ which contain V_i as an endpoint collide there. Hence any endpoint edges containing V_i are into V_i.

Since the edges in U are a subset of the internal and endpoint edges in U', and all the edges in U' that contained V_i are into V_i it follows that V_i is a collider on U.

Next we prove that every collider V_j on U has some descendant in $\mathbf{Z}$:

Since the edges in U are a subset of those in U', it follows that if V_i is a collider on U then there was some edge $Y_{h-1} \rightarrow Y_h$ where $V_{i-1} \equiv Y_{h-1}$ and $V_i \equiv Y_h$. Similarly, there was some edge $Y_k \leftarrow Y_{k+1}$ where $V_i \equiv Y_k$ and $V_{i+1} \equiv Y_{k+1}$ $(h \leq k)$.

Suppose that no descendant of $V_i \equiv Y_h$ is in $\mathbf{Z}$. If Y_h occurs on a path P_1 in $\mathcal{T}$ between two vertices X_α and $X_{\alpha+1}$ in $\mathbf{R}$ which was d-connecting given $\mathbf{Z}\backslash\{X_\alpha, X_{\alpha+1}\}$, then either there is a subpath of P_1 that is a directed path from Y_h to X_α, or there is a subpath of P_1 that is a directed path from Y_h to $X_{\alpha+1}$, or it is an ancestor of some element of $\mathbf{Z}\backslash\{X_\alpha, X_{\alpha+1}\}$.[15] Since we supposed that no descendant of V_i is in $\mathbf{Z}$, we must conclude that either there is a subpath of P_1 that is a directed path from V_i to X_α, or there is a subpath of P_1 that is a directed path from V_i to $X_{\alpha+1}$. However since V_i occurs on an edge of the form $Y_{h-1} \rightarrow Y_h \equiv V_i$, it follows that there is a subpath of P_1 that is a directed path from V_i to $X_{\alpha+1}$, and hence V_i is an ancestor of $X_{\alpha+1}$. (Alternatively, if Y_h is in R, then let α be s.t. $X_{\alpha+1} \equiv Y_h$. The next step of the argument then follows.)

But we can now carry out a similar argument with $X_{\alpha+1}$ and $X_{\alpha+2}$. If the path in $\mathcal{T}$ from $X_{\alpha+1}$ to $X_{\alpha+2}$, say P_2, is into $X_{\alpha+1}$, then the two paths P_1 and P_2 in $\mathcal{T}$, collide at

[15]It is easy to show if A and B are d-connected given $\mathbf{Z}$ by some path P, then every vertex V on P is either an ancestor of $\mathbf{Z}$, or on a directed subpath to A, or on a directed subpath to B. For a proof see Lemma 20 of Richardson(1994).

$X_{\alpha+1}$. But in that case, by condition c(iii) of the antecedent, some descendant of $X_{\alpha+1} \in$ **Z**, and hence some descendant of V_i is in **Z**, contrary to hypothesis. Hence the path in $\mathcal{T}$ between $X_{\alpha+1}$ and $X_{\alpha+2}$ is out of $X_{\alpha+1}$. Let W be the first vertex on P_2 after $X_{\alpha+1}$. Since the first edge on P_2 is out of $X_{\alpha+1}$ it follows that either there is a collider on the path that is a descendant of W, or $P_2(W, X_{\alpha+2})$ is a directed path from W to $X_{\alpha+2}$. But W is a descendant of V_i, so it is not an ancestor of any member of **Z**, and hence is not an ancestor of a collider on P_2. It follows that $P_2(W, X_{\alpha+2})$ is a directed path from W to $X_{\alpha+2}$. Hence the path in $\mathcal{T}$ between $X_{\alpha+1}$ and $X_{\alpha+2}$ is of the form: $X_{\alpha+1}\rightarrow\ldots\ldots\rightarrow X_{\alpha+2}$. By repeating this argument we can show that for every r s.t. $\alpha+r \leq n$, the path in $\mathcal{T}$ between $X_{\alpha+r}$ and $X_{\alpha+r+1}$ is of the form: $X_{\alpha+r}\rightarrow\ldots\rightarrow X_{\alpha+r+1}$. But this is a contradiction since the edge $Y_k \leftarrow Y_{k+1}$ occurs on one of these paths. Hence some descendant of V_i is in **Z**.

This completes the proof of Lemma 1. ∴

Lemma 2: In a directed graph *G* with vertices **V**, if **X**, **Y**, and **Z** are disjoint subsets of **V**, and **X** is d-connected to **Y** given **Z** in *G*, then **X** is d-connected to **Y** given **Z** in an acyclic directed subgraph of *G*.

Proof. Suppose that *U* is an undirected path that d-connects X and Y given **Z**, and C is a collider on *U*. Let *length*(C,**Z**) be 0 if C is a member of **Z**; otherwise it is the length of a shortest directed path from C to a member of **Z**. Let *size*(*U*) equal the number of colliders on *U* plus the sum over all colliders C on *U* of *length*(*C*,**Z**). *U* is a **minimal d-connecting path** between X and Y given **Z**, if *U* d-connects X and Y given **Z** and there is no other path *U'* that d-connects X and Y given **Z** such that *size*(*U'*) < *size*(*U*). If there is a path that d-connects X and Y given **Z** there is at least one minimal d-connecting path between X and Y given **Z.**

Suppose **X** is d-connected to **Y** given **Z.** Then for some X in **X** and Y in **Y**, there is a minimal d-connecting path *U* between X and Y given **Z**. *U* in *G*. First we will show that no shortest acyclic directed path D_i from a collider C_i on *U* to a member of **Z** intersects *U* except at C_i by showing that if such a point of intersection exists then *U* is not minimal, contrary to our assumption. See Figure 18.

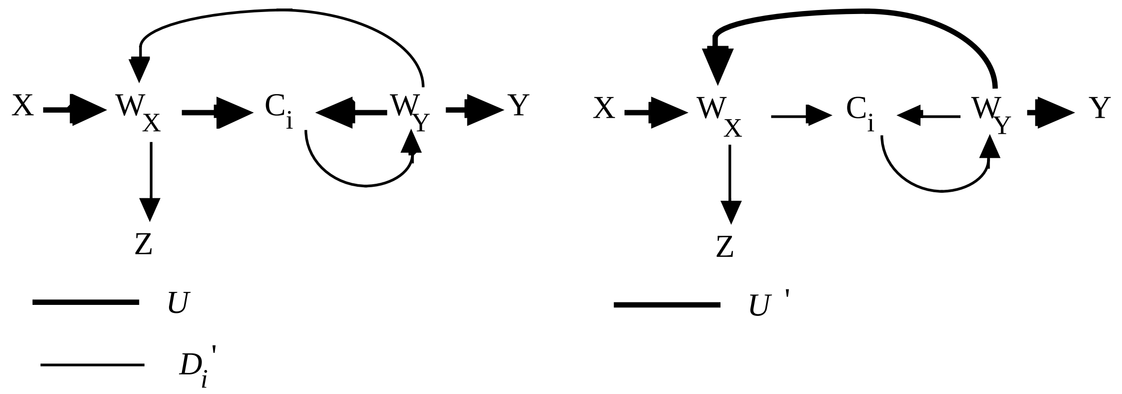

Figure 18

Form the path *U'* in the following way. If D_i intersects *U* at a vertex other than C_i then let W_X be the vertex closest to X on *U* that is on both D_i and *U*, and W_Y be the vertex closest to Y on *U* that is on both D_i and *U*. Suppose without loss of generality that W_X is after W_Y on D_i. Let *U'* be the concatenation of $U(X,W_X)$, $D_i(W_Y,W_X)$, and $U(W_Y,Y)$. It is now easy to show that *U'* d-connects X and Y given **Z**, and *size*(*U'*) < *size*(*U*) because *U'* contains no more colliders than *U* and a shortest directed path from W_X to a member of **Z** is shorter than D_i. Hence *U* is not minimal, contrary to the assumption.

Next, we will show that if *U* is minimal, then it does not contain a pair of colliders C and D such that a shortest directed path from C to a member of **Z** intersects a shortest path from D to a member of **Z**. Suppose this is false. See Figure 19.

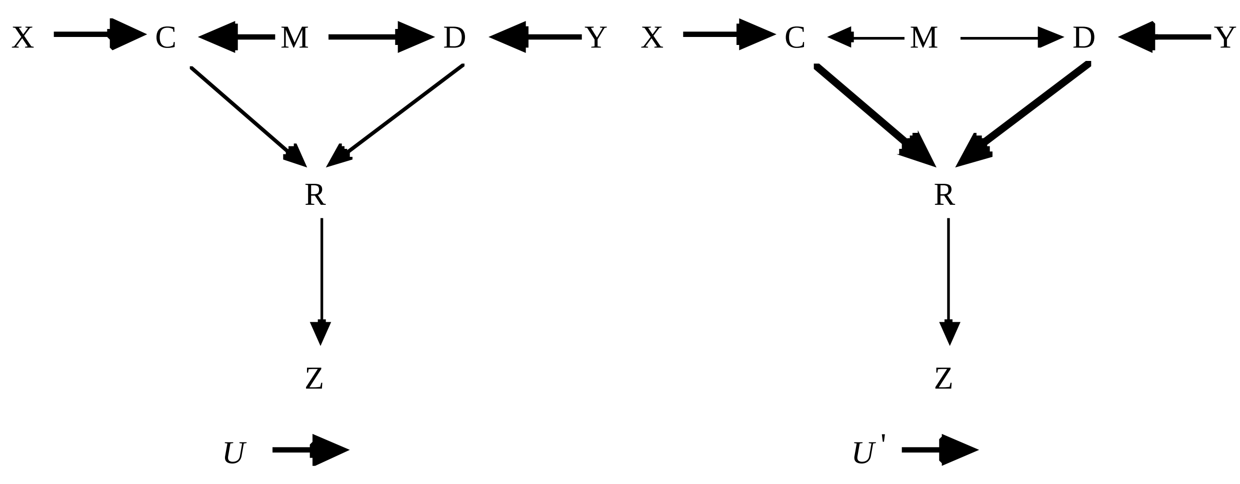

Figure 19

Let D_1 be a shortest directed acyclic path from C to a member of **Z** that intersects D_2, a shortest directed acyclic path from D to a member of **Z**. Let the vertex on D_1 closest to C that is also on D_2 be R. Let U' be the concatenation of U(X,C), D_1(C,R), D_2(D,R), and U(Y,D). It is now easy to show that U' d-connects X and Y given **Z** and $size(U') < size(U)$ because C and D are not colliders on U', the only collider on U' that may not be on U is R, and the length of a shortest path from R to a member of **Z** is less than the length of a shortest path from D to a member of **Z**. Hence U is not minimal, contrary to the assumption.

For each collider C on a minimal path U that d-connects X and Y given **Z**, a shortest directed path from C to a member of **Z** does not intersect U except at C, and does not intersect a shortest directed path from any other collider D to a member of **Z**. It follows that the subgraph consisting of U and a shortest directed acyclic path from each collider on U to a member of **Z** is acyclic. $\therefore$

Lemma 3: In a directed (cyclic or acyclic) graph G, disjoint sets of variables **X**, **Y** and **Z**, **X** and **Y** are Pearl d-connected given **Z** if and only if **X** and **Y** are Lauritzen d-connected given **Z.**

Proof: First we will show that Pearl d-connection entails Lauritzen d-connection. If **X** and **Y** are Pearl d-connected given **Z** then there is some Pearl d-connecting path P from a vertex X$\in$**X** to a vertex Y$\in$**Y**. The adjacencies in the moralized undirected graph Moral(G (An({X}$\cup${Y}$\cup$**Z**))) are a superset of the adjacencies in G, hence there is a path P^* in Moral(G (An({X}$\cup${Y}$\cup$**Z**))) corresponding to P in G. Since P is Pearl d-connecting the only vertices B on P^* that can be in **Z** occur on P in colliders of the form A$\rightarrow$B$\leftarrow$C. A and C are not in **Z** because they are non-colliders on P. Since B$\in$**Z**, A and C are adjacent in Moral(G^*). Thus, for any vertex B$\in$**Z** on P^* it is possible to form a path Q^* from P^* by replacing the A—B—C sub-path in P^* with the edge A—C. Since no vertex in **Z** occurs on Q^*, X and Y are not separated in Moral(G^*(An({X}$\cup${Y}$\cup$**Z**))), hence **X** and **Y** are Lauritzen d-connected given **Z**.

Next we will show that Lauritzen d-connection entails Pearl d-connection. If **X** and **Y** are Lauritzen d-connected given **Z** then there is some path Q in the moralized undirected graph Moral(G(An(**X**$\cup$**Y**$\cup$**Z**))) connecting a vertex X$\in$**X** to a vertex Y$\in$**Y**, on which there is no vertex in **Z**. The edges in Moral(G(An(**X**$\cup$**Y**$\cup$**Z**))) are a superset of the edges in G(An(**X**$\cup$**Y**$\cup$**Z**)), but whenever there is an A—C edge in Moral(G(An(**X**$\cup$**Y**$\cup$**Z**))) which is not present in G(An(**X**$\cup$**Y**$\cup$**Z**)) then there is some vertex B such that A$\rightarrow$B$\leftarrow$C in G, and B is an ancestor in G of some vertex in **X**$\cup$**Y**$\cup$**Z** (since every vertex in G(An(**X**$\cup$**Y**$\cup$**Z**)) is an ancestor of **X**$\cup$**Y**$\cup$**Z**.)

Form an undirected path P in G as follows. Replace each undirected edge on Q by a directed edge in G if such exists (if there is more than one then pick one). If there is an undirected edge A—C on Q which is not present in G then replace A—C by a subpath $A \rightarrow B \leftarrow C$ where $B \in$ Ancestors($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$); by the construction of the moralized graph there is guaranteed to be such a B. Note that since such vertices B are in Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))), it follows that every vertex on P occurs in Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))).

Index the vertices on the path P as follows: $\langle V_0, \ldots V_{n+1} \rangle$ (so $V_0 \equiv X$, $V_{n+1} \equiv Y$). Now construct the following sets:

$\mathbf{T_X} = \{V_i \mid V_i \in \mathrm{An}(\mathbf{X}) \backslash \mathrm{An}(\mathbf{Z})\}$ and $\mathbf{T_Y} = \{V_i \mid V_i \in \mathrm{An}(\mathbf{Y}) \backslash \mathrm{An}(\mathbf{Z})\}$.

Let α be the largest k s.t. $V_k \in \mathbf{T_X}$, (if $\mathbf{T_X} = \emptyset$, let $\alpha = 0$). Let β be the smallest k greater than α s.t. $V_k \in \mathbf{T_Y}$, (if $\mathbf{T_Y} = \emptyset$, let $\beta = n+1$).

We will now show that every vertex V_k on $P(V_\alpha, V_\beta)$, except for V_α and V_β, is an ancestor of $\mathbf{Z}$. Suppose some vertex V_k ($\alpha < k < \beta$) is not an ancestor of $\mathbf{Z}$, then since every vertex V_i on P occurs in Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))), $V_i \in \mathrm{An}(\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z})$. If $V_i \notin \mathrm{An}(\mathbf{Z})$ then $V_i \in \mathrm{An}(\mathbf{X} \cup \mathbf{Y})$. However, if $V_i \in \mathrm{An}(\mathbf{X}) \backslash \mathrm{An}(\mathbf{Z})$ then $V_i \in \mathbf{T_X}$, so α is not the largest k s.t. $V_k \in \mathbf{T_X}$. Similarly if $V_i \in \mathrm{An}(\mathbf{Y}) \backslash \mathrm{An}(\mathbf{Z})$ then $V_m \in \mathbf{T_Y}$, so β is not the smallest k s.t. $V_k \in \mathbf{T_Y}$. Hence $V_i \in \mathrm{An}(\mathbf{Z})$.

Since $V_\alpha \in \mathbf{T_X}$, $V_\alpha \in \mathrm{An}(\mathbf{X})$ hence there is a directed path D_α in G from V_α to some vertex X' in $\mathbf{X}$. (If $\alpha = 0$ then this is trivially true since $V_\alpha \equiv V_0 \equiv X \in \mathbf{X}$.) Further, since $V_\alpha \in \mathbf{T_X}$, $V_\alpha \notin \mathrm{An}(\mathbf{Z})$, so no vertex in $\mathbf{Z}$ occurs on the path D_α. Likewise there is a directed path D_β in$G$ from V_β to some vertex Y' in $\mathbf{Y}$, and no vertex in $\mathbf{Z}$ occurs on the path D_β.

We now construct a set $\mathcal{T}$ of paths as follows: For each edge $V_i \rightarrow V_{i+1}$ or $V_i \leftarrow V_{i+1}$ on $Q(V_\alpha, V_\beta)$ put the (one-edge) path between V_i and V_{i+1} into $\mathcal{T}$. If the directed path D_α is of length greater than 0 then put and D_α into $\mathcal{T}$, and similarly for D_β.

We will now show that $\mathcal{T}$ satisfies the conditions for Lemma 1, with the sequence $R \equiv \langle X', V_\alpha, \ldots V_\beta, Y' \rangle$ (in the cases in which $V_\alpha \equiv X \equiv X'$ omit X′, similarly if $V_\beta \equiv Y \equiv Y'$ omit Y′).

Condition (a): (Pairwise distinctness.) This follows directly from the fact that there is an edge between V_k and V_{k+1}.

Condition (b): $\mathbf{Z} \subseteq \mathbf{V} \setminus \{X', Y'\}$. This follows since **X**, **Y** and **Z** are pairwise disjoint.

Condition (c(i)): A directed path *P* in $\mathcal{T}$ between each pair of consecutive vertices which d-connects given $\mathbf{Z} \setminus \{$endpoints of $P\}$. This holds trivially for V_i, V_{i+1} since there is an edge between V_i and V_{i+1}. The path D_α d-connects X´ and V_α since, as we showed above, it is a directed path no vertex of which is in **Z**. (In the case in which $V_\alpha \equiv X \equiv X'$, this case does not arise.) Similarly the path D_β is d-connecting given **Z** since no vertex on the path is in **Z**.

Condition(c(ii)): If some vertex W in *R* is in **Z** then the paths in $\mathcal{T}$ that contain W as an endpoint collide at W. The vertices on $V_\alpha, \ldots V_\beta$ on *P* are either (a) vertices that were present on the path *Q* in Moral(*G*(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))), or (b) they are vertices that were added when edges (A—C) not present in *G* were replaced by unshielded colliders ($A \rightarrow B \leftarrow C$), (or both, since the path *P* is not guaranteed to be acyclic). Any vertex in category (a) is not in **Z**, since if it were then the original path *Q* would not have been connecting in Moral(*G*(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))) given **Z**. The vertices V_α and V_β are not in **Z**, since V_α and V_β are either not ancestors of **Z**, or in the case in which $V_\alpha \equiv X$ or $V_\beta \equiv Y$ are elements of $\mathbf{X} \cup \mathbf{Y}$; in both cases $V_\alpha, V_\beta \notin \mathbf{Z}$. Hence if $V_i \in \mathbf{Z}$, then V_i occurs as a collider (possibly more than once) on the path **P**, hence all paths in $\mathcal{T}$ that contain V_i as an endpoint collide at V_i.

Condition(c(iii)): If for three consecutive vertices <A,B,C> in the sequence *R* the d-connecting paths between A and B, and between B and C in $\mathcal{T}$ collide at B then B has a descendant in **Z**. We have already shown that every vertex V_k on $P(V_\alpha, V_\beta)$, except for V_α and V_β is an ancestor of **Z**. Thus it is sufficient to show that the paths in $\mathcal{T}$ do not collide at V_α or V_β. However, this follows immediately from the fact that D_α is a directed path from V_α to X´, and D_β is a directed path from V_α to Y´. (In the cases in which $V_\alpha \equiv X$, V_α is the first vertex in the sequence *R* so the case does not arise, similarly if $V_\beta \equiv Y$, then V_β is the last vertex in the sequence.)

We can now apply Lemma 1 to construct an acyclic d-connecting path from X´ to Y´ given **Z**. $\therefore$

Since Lauritzen d-separation and Pearl d-separation are equivalent, henceforth we will simply refer to "d-separation" when the context makes clear which definition is being used.

Theorem 3: If $\mathbf{V}$ is a set of random variables with a probability measure P that has a positive density function $f(\mathbf{V})$, and P satisfies the global directed Markov property for directed (cyclic or acyclic) graph G, then $f(\mathbf{V})$ factors according to G.

Proof. Assume that probability measure over $\mathbf{V}$ satisfies the global directed Markov property for directed (cyclic or acyclic) graph G. Let $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ be three arbitrary disjoint sets of vertices in G. Since by Lemma 3, Pearl d-separation and Lauritzen d-separation are equivalent, we will now show that for any disjoint sets of variables $\mathbf{R}$, $\mathbf{S}$, and $\mathbf{T}$ included in An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$), if $\mathbf{R}$ and $\mathbf{S}$ are separated given $\mathbf{T}$ in Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))), then $\mathbf{R}$ and $\mathbf{S}$ are independent given $\mathbf{T}$. If $\mathbf{R}$, $\mathbf{S}$, and $\mathbf{T}$ are included in An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$), then An($\mathbf{R} \cup \mathbf{S} \cup \mathbf{T}$) is included in An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$). Any pair of vertices A and B adjacent in Moral(G(An($\mathbf{R} \cup \mathbf{S} \cup \mathbf{T}$))) is also adjacent in Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))) because G(An($\mathbf{R} \cup \mathbf{S} \cup \mathbf{T}$)) is a subgraph of G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$)). Hence Moral($G$(An($\mathbf{R} \cup \mathbf{S} \cup \mathbf{T}$))) is a subgraph of Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))). It follows that if $\mathbf{R}$ and $\mathbf{S}$ are separated given $\mathbf{T}$ in Moral(G(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$))) they are also separated in Moral(G(An($\mathbf{R} \cup \mathbf{S} \cup \mathbf{T}$))). But by the global directed Markov property, if $\mathbf{R}$ and $\mathbf{S}$ are separated given $\mathbf{T}$ in Moral(G(An($\mathbf{R} \cup \mathbf{S} \cup \mathbf{T}$))) then $\mathbf{R}$ and $\mathbf{S}$ are independent given $\mathbf{T}$. It follows from the Hammersly-Clifford Theorem (see Lauritzen *et al.* 1990) that the density function f(An($\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$)) can be factored as

$$f(\mathrm{An}(\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z})) = \prod_{\mathrm{V} \in \mathrm{An}(\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z})} g_{\mathrm{V}}(\mathrm{V}, \mathrm{Parents}(\mathrm{V}))$$

where each g_V is a positive function, i.e., the density function factors according to G. $\therefore$

7.2. Proof of Theorem 4

Theorem 4: The probability measure P over the substantive variables of a linear SEM L (recursive or non-recursive) with jointly independent error varaibles satisfies the global directed Markov property for the directed (cyclic or acyclic) graph G of L, i.e. if $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ are disjoint sets of variables in G and $\mathbf{X}$ is d-separated from $\mathbf{Y}$ given $\mathbf{Z}$ in G, then $\mathbf{X}$ and $\mathbf{Y}$ are independent given $\mathbf{Z}$ in P.

Proof. Let **Err**($\mathbf{X}$) be the set of error variables corresponding to a set of substantive variables $\mathbf{X}$. In order to distinguish the density function for $\mathbf{V}$ from the density function for the error variables we will use $f_{\mathbf{V}}$ to represent the density function (including marginal densities) for the latter and $f_{\mathbf{Err}}$ to represent the density function of the former. If $\mathbf{V}$ is the set of variables in G, then by hypothesis,

$$f_{\mathbf{Err}}(\mathbf{Err}(\mathbf{V})) = \prod_{\varepsilon \in \mathbf{Err}(\mathbf{V})} f_{\mathbf{Err}}(\varepsilon)$$

It is possible to integrate out the error variables not in **Err**(An(**X**)) and obtain

$$f_{\mathbf{Err}}(\mathbf{Err}(\mathrm{An}(\mathbf{X}))) = \prod_{\varepsilon \in \mathbf{Err}(\mathrm{An}(\mathbf{X}))} f_{\mathbf{Err}}(\varepsilon)$$

Because for each variable X in **V**, X is a linear function of its parents in *G* plus a unique error variable ε_X, it follows that ε_X is a linear function g_X of X and the parents of X in *G*. Hence **Err**(An(**X**)) is a function of An(**X**). Following Haavelmo (1943) it is possible to derive the density function for the set of variables An(**X**) by replacing each ε_X in $f_{\mathbf{Err}}(\varepsilon_X)$ by g_X(X,Parents(X)) and multiplying by the absolute value of the Jacobian:

$$f_{\mathrm{V}}(\mathrm{An}(\mathbf{X})) = \prod_{\mathrm{X} \in \mathrm{An}(\mathbf{X})} f_{\mathbf{Err}}(g_X(\mathrm{X}, \mathrm{Parents}(\mathrm{X}))) \times |J|$$

where *J* is the Jacobian of the transformation. Because the transformation is linear, the Jacobian is a constant. All of the terms in the multiplication are non-negative because they are either a density function or a positive constant. It follows from Theorem 2 that if **X** and **Y** are d-separated given **Z** then **X** and **Y** are independent given **Z**. $\therefore$

7.3. Proof of Theorem 5

Theorem 5: In a linear SEM *L* with jointly independent error variables and directed (cyclic or acyclic) graph *G* containing disjoint sets of variables **X**, **Y** and **Z**, if **X** is not d-separated from **Y** given **Z** in *G* then *L* does not linearly entail that **X** is independent of **Y** given **Z**.

Proof. Suppose that **X** is not d-separated from **Y** given **Z**. By Lemma 2, if **X** is not d-separated from **Y** given **Z** in a cyclic graph *G*, then there is some acyclic subgraph *G'* of *G* in which **X** is not d-separated from **Y** given **Z**. Geiger and Pearl (1988) have shown that if **X** is not d-separated from **Y** given **Z** in a directed acyclic graph, then there is some distribution represented by the directed acyclic graph in which **X** is not independent of **Y** given **Z**, and it has been shown (Spirtes, Glymour and Scheines, 1993) that there is in particular a linear normal distribution P in which **X** is not independent of **Y** given **Z**. If P satisfies the global directed Markov property for *G'* it also satisfies it for *G* because every d-connecting path in *G'* is a d-connecting path in *G*. Hence there is some linear normal distribution represented by *G* in which **X** is not independent of **Y** given **Z**. $\therefore$

7.4. Proof of Theorem 6

Theorem 6: In a linear SEM *L* with jointly independent error variables and (cyclic or acyclic) directed graph *G* containing substantive variables X, Y and **Z**, where X ≠ Y and **Z** does not contain X or Y, X is d-separated from Y given **Z** in *G* if and only if *L* linearly entails that $\rho_{XY.\mathbf{Z}} = 0$.

Proof. (This proof for cyclic or acyclic graphs is based on the proof for acyclic graphs in Verma and Pearl, 1990.) Let *L'* be a linear SEM with the same directed graph *G* and that is the same as *L* except that the exogenous variables are jointly normally distributed with the same variances as the corresponding variables in *L*. By Theorem 4 and Theorem 5, *L'* linearly entails that X is independent of Y given **Z** if and only if X is d-separated from Y given **Z** in *G*. Hence for all values of the linear coefficients and all joint normal distributions over the exogenous variables, $\rho_{XY.\mathbf{Z}} = 0$ if and only if X is d-separated from Y given **Z** in *G*. Because the value of a partial correlation in a linear SEM depends only on the values of the linear coefficients and the variances of the exogenous variables, *L'* linearly entails $\rho_{XY.\mathbf{Z}} = 0$ if and only if X is d-separated from Y given **Z** in *G* and hence *L* also linearly entails that $\rho_{XY.\mathbf{Z}} = 0$ if and only if X is d-separated from Y given **Z** in *G*.
∴

7.5. Proof of Theorem 7

Theorem 7: (Soundness) Given as input an oracle for testing d-separation relations in the directed (cyclic or acyclic) graph *G*, then the output is a PAG **Ψ** for *G*.

Proof. The proof proceeds by showing that each section of the CCD algorithm makes correct inferences from the d-separation oracle for *G*, to the structure of any graph in **Equiv**(*G*).

<u>Section ¶A</u>
Lemma 4: Given a PAG **Ψ** for graph *G* with vertex set **V**, if at least one of the following holds:

(i) X is a parent of Y in *G*, or
(ii) Y is a parent of X in *G*, or
(iii) there is some vertex Z which is a child of both X and Y, such that Z is an ancestor of either X or Y (or both)

then X and Y are p-adjacent in **Ψ**, i.e. X and Y are d-connected given any subset $\mathbf{S}\subseteq\mathbf{V}\backslash\{X,Y\}$ of the other vertices in *G*.

Proof. If (i) holds then the path $X \rightarrow Y$ d-connects X and Y given any subset $\mathbf{S} \subseteq \mathbf{V} \backslash \{X,Y\}$, hence X and Y are p-adjacent in any PAG $\boldsymbol{\Psi}$ for graph *G*. The case in which (ii) holds is equally trivial: $X \leftarrow Y$ is a d-connecting path given any set $\mathbf{S} \subseteq \mathbf{V} \backslash \{X,Y\}$.

If (iii) holds there is a common child (Z) of X and Y which is an ancestor of X or Y; therefore either there is a directed path $X \rightarrow Z \rightarrow A_1 \rightarrow \ldots A_n \rightarrow Y$ $(n \geq 0)$, or there is a directed path $Y \rightarrow Z \rightarrow A_1 \rightarrow \ldots A_n \rightarrow X$. Suppose without much loss of generality that it is the former. Let $\mathbf{S}$ be an arbitrary subset of the other variables ($\mathbf{S} \subseteq \mathbf{V} \backslash \{X,Y\}$). There are two cases to consider:

Case 1: $\mathbf{S} \cap \{Z, A_1 \ldots A_n\} \neq \emptyset$; in this case $X \rightarrow Z \leftarrow Y$ is a d-connecting path.

Case 2: $\mathbf{S} \cap \{Z, A_1 \ldots A_n\} = \emptyset$; then $X \rightarrow Z \rightarrow A_1 \rightarrow \ldots A_n \rightarrow Y$ is a d-connecting path. $\therefore$

Lemma 5: In a graph *G*, with vertices $\mathbf{V}$, if the following hold:[16]

(i) X is not a parent of Y in *G*
(ii) Y is not a parent of X in *G*, and
(iii) there is no vertex Z s.t. Z is a common child of X and Y, and an ancestor of X or Y,

then for any set $\mathbf{Q}$, X and Y are d-separated given $\mathbf{T}$, defined as follows:

$\mathbf{S}$ = Children(X) $\cap$ An($\{X,Y\} \cup \mathbf{Q}$)

$\mathbf{T}$ = (Parents($\mathbf{S} \cup \{X\}$) $\cup$ $\mathbf{S}$)\(Descendants(Children(X)$\cap$Children(Y)) $\cup$ $\{X,Y\}$).

Proof. Every vertex in $\mathbf{S}$ is an ancestor of X or Y or $\mathbf{Q}$. Every vertex in $\mathbf{T}$ is either a parent of X, a vertex in $\mathbf{S}$, or a parent of a vertex in $\mathbf{S}$, hence every vertex in $\mathbf{T}$ is an ancestor of X or Y or $\mathbf{Q}$. We will now show that if (i), (ii), and (iii) hold then X and Y are d-separated given $\mathbf{T}$.

Suppose, on the contrary that there is a path d-connecting X and Y given $\mathbf{T}$. Let W be the first vertex on the path from X to Y. (It follows from (i) and (ii) that $W \neq Y$.) There are two cases to consider:

Case 1 The path contains $X \leftarrow W \ldots Y$.

Subcase A: W is not a descendant of a common child of X and Y. If W is not a descendant of a common child, then $W \in \mathbf{T}$ (Since W is a parent of X). Thus since W is a non-collider on the path, the path is not d-connecting given $\mathbf{T}$.

---

[16]i.e. None of the conditions in the antecedent of Lemma 4 hold.

Subcase B: W is a descendant of a common child of X and Y. In this case since X is a child of W, it follows that X is a descendant of some common child Z of X and Y. But this is contrary to the assumption that (iii) holds.

Case 2 The path contains X→W… Y.

Subcase A: W is not a descendant of a common child of X and Y. Let V be the next vertex on the path.

Sub-subcase a: The path contains X→W←V…Y. If this path is d-connecting then some descendant of W is in **T**, but then some descendant of W is an ancestor of X or Y or **Q**. Hence W is an ancestor of X, Y or **Q**. So if some descendant of W is in **T**, then W is in **S**. Moreover, since W is (by hypothesis) not a descendant of a common child, V ≠ Y.

Now V is a parent of W, and W∈ **S**. Moreover V is not a descendant of a common child since in that instance W would also be a descendant of a common child, contrary to hypothesis. X ≠ V ≠ Y, so V∈ **T**. Thus V occurs as a non-collider, but V ∈ **T**, hence the path fails to d-connect given **T**.

Sub-subcase b: The path contains X→W→V…Y. If some path X→W→V…Y d-connects given **T** then W is either an ancestor of Y or some vertex in **T**. However if W is an ancestor of some vertex in **T**, then W is an ancestor of X, Y or **Q**, since every vertex in **T** is an ancestor of X, Y or **Q**. Hence W∈ **S**, and thus since W is (by hypothesis) not a descendant of a common child of X and Y, and X ≠ W ≠ Y, W ∈ **T**. Since W occurs as a non-collider on this path, it follows that any path X→W→V…Y fails to d-connect given **T**. (This allows for the possibility that V = Y).

Subcase B: W is a descendant of a common child. Thus Descendants(W) ∩ **T** = ∅, since descendants of W are also descendants of common children of X and Y and so cannot occur in **T**.

Since no descendant of W has been conditioned on, if W occurs on a d-connecting path then W is a non-collider. We can show that any other vertex on such a d-connecting path must be a non-collider. Suppose that there is a collider on the path, then take the first collider on the path after W, let us say <A, B, C>, so that the path now takes the form: X→W→V→…→…→A→B←C…Y. Since <A, B, C> is the first collider after V, it follows that B is a descendant of W. But if the path is d-connecting then some descendant of B, say D, has been conditioned on, i.e. D∈ **T**. But then since

D is a descendant of B, and B is a descendant of W, D$\in$ Descendants(W) which is a contradiction since Descendants(W) $\cap$ **T** = $\emptyset$.

As there are no colliders on the path it follows that W is an ancestor of Y. But then W is a descendant of a common child of X and Y, *and* an ancestor of Y. But this contradicts (iii).

This completes the proof of Lemma 5.$\therefore$

Corollary 1: Given a graph *G*, and PAG Ψ for *G*, X and Y are p-adjacent in Ψ if and only if at least one of the following holds in *G*:

(i) X is a parent of Y, or

(ii) Y is a parent of X, or

(iii) there is some vertex Z which is a child of both X and Y, such that Z is an ancestor of either X or Y (or both).

Proof. 'If' is proved by Lemma 4. 'Only if' follows from Lemma 5 with **Q** = $\emptyset$ by contraposition. $\therefore$

We have obtained necessary and sufficient conditions on a graph *G* for a pair of vertices to be p-adjacent in any PAG for *G*. Thus it makes sense to speak of a pair of vertices X, Y being **p-adjacent in graph** *G*, where this means that at least one of (i), (ii) and (iii) holds.

The previous Corollary tells us that a pair of vertices are p-adjacent in *G* if and only if they are p-adjacent in every PAG for *G*. For this reason we will often refer to a pair of variables as p-adjacent without specifying whether we are referring to the graph or the PAG.

Corollary 2: In a graph *G*, if X and Y are d-separated by some set **R**, then X and Y are d-separated by a set **T** in which every vertex is an ancestor of X or Y. Furthermore, either **T** is a subset of the vertices p-adjacent to X or X is an ancestor of Y in *G*.

Proof. Since X and Y are d-separated by some set **R**, X and Y are not p-adjacent in *G*. Apply Lemma 5, with **Q** = $\emptyset$. In that case

S = Children(X) $\cap$ An({X,Y})

T = (Parents(**S** $\cup$ {X}) $\cup$ **S**)\(Descendants(Children(X)$\cap$Children(Y)) $\cup$ {X,Y})

It follows from Lemma 5 that X and Y are d-separated given **T**. Every vertex in **S** is an ancestor of X or Y. Every vertex in **T** is either a parent of X, a vertex in **S**, or a parent of a vertex in **S**, and hence every vertex in **T** is an ancestor of X or Y.

We will now show that either **T** is a subset of the vertices p-adjacent to X or X is an ancestor of Y in *G*. Every vertex in **T** is either a parent of X, a child of X, or a parent V of some vertex C in **S**, where C is also a child of X. Any vertex in the first two categories is clearly p-adjacent to X. Since C is in **S**, C is an ancestor of X or Y. If C is an ancestor of X, then V is p-adjacent to X. If C is an ancestor of Y, then X is an ancestor of Y.∴

Lemma 6: In a graph *G*, if A and B are not p-adjacent, then either A and B are d-separated given a set $\mathbf{T_A}$ of vertices p-adjacent to A or by a set $\mathbf{T_B}$ of vertices p-adjacent to B.

Proof. By Corollary 2, if A and B are not p-adjacent then A and B are d-separated given $\mathbf{T_A}$ where:

$\mathbf{S_A}$ = Children(A) $\cap$ An({A,B})

$\mathbf{T_A}$ **=** (Parents($\mathbf{S} \cup$ {A}) $\cup$ **S**)\(Descendants(Children(A)$\cap$Children(B)) $\cup$ {A,B}),

Case 1: A is not an ancestor of B. From Corollary 2, since A is not an ancestor of B, $\mathbf{T_A} \subseteq$ {X | X p-adjacent to A}.

Case 2: B is not an ancestor of A. It follows again by symmetry that A and B are d-separated given $\mathbf{T_B}$, where $\mathbf{T_B}$ is defined symmetrically to $\mathbf{T_A}$ in Case 1.

Case 3: B is an ancestor of A and A is an ancestor of B. Now any vertex V in $\mathbf{T_A}$ is either a child of A, a parent of A or a parent of some vertex C in $\mathbf{S_A}$, which is itself a child of A. Clearly vertices in the first two categories are p-adjacent to A; as before, vertices in the last category are p-adjacent to A if C is an ancestor of A. Any vertex in $\mathbf{S_A}$ is an ancestor of A or B. Since A is an ancestor of B, and B is an ancestor of A, it follows that every vertex in $\mathbf{S_A}$ is an ancestor of A, hence every vertex in $\mathbf{T_A}$ is p-adjacent to A. (Note that it is also the case that every vertex in $\mathbf{T_B}$ is p-adjacent to B.) ∴

Suppose that the input to the algorithm is a d-separation oracle for a directed graph *G*. To find a set which d-separates some pair of variables A and B in *G* the algorithm tests subsets of the vertices which are p-adjacent to A in Ψ, and subsets of vertices which are p-adjacent to B in Ψ to see if they d-separate A and B. Since the vertices which are p-adjacent to A and B in *G* are at all times a subset of the vertices p-adjacent to A and B in Ψ[17] it follows from Lemma 6 that step ¶A is guaranteed to find a set which d-separates A and B, if any set d-separates A and B in *G*.

[17]This is because if a pair of vertices X,Y are p-adjacent in *G* then no set is found which d-separates them, and hence the edge between X and Y in Ψ is never deleted.

Section ¶B

The next lemma gives an important property of d-separating sets that are found through a search which never tests a set unless it has already tested every proper subset of that set (as in the CCD algorithm.)

Lemma 7: Suppose that in a graph G, Y is not an ancestor of X or Z or **R**. If there is a set **S**, $\mathbf{R}\subset\mathbf{S}$, such that $Y\in\mathbf{S}$ and every proper subset **T** s.t. $\mathbf{R}\subseteq\mathbf{T}\subset\mathbf{S}$, not containing Y, d-connects X and Z, in G , then **S** d-connects X and Z in G.

Proof. Let $\mathbf{T}^* = \mathrm{An}(\{X,Z\}\cup\mathbf{R})\cap\mathbf{S}$. Now, $\mathbf{R}\subseteq\mathbf{T}^*$, and $\mathbf{T}^*$ is a proper subset of **S**, so by hypothesis there is a d-connecting path, P, conditional on $\mathbf{T}^*$. By the definition of a d-connecting path, every element on P is either an ancestor of one of the endpoints, or $\mathbf{T}^*$. Moreover, by definition, every element in $\mathbf{T}^*$ is an ancestor of X or Z or **R**. Thus every element on the path P is an ancestor of X or Z or **R**. Since neither Y nor any element in $\mathbf{S}\backslash\mathbf{T}^*$ is an ancestor of X or Z or **R**, it follows that no vertex in $\mathbf{S}\backslash\mathbf{T}^*$ lies on P. Since $\mathbf{T}^*\subset\mathbf{S}$ the only way in which P could fail to d-connect given **S** would be if some element of $\mathbf{S}\backslash\mathbf{T}^*$ lay on the path. Hence P still d-connects X and Z given **S**.$\therefore$

In a graph G , if X and Y are d-separated given **S**, and are d-connected given any proper subset of **S**, then **S** is a **minimal d-separating** set for X and Y in G.

The following corollary is useful here:

Corollary 3: In a graph G, if **S** is a minimal d-separating set for X and Y, then any vertex in **S** is an ancestor of X or Y in G.

Proof. The corollary follows immediately from Lemma 7, with $\mathbf{R} = \emptyset$ via contraposition.$\therefore$

This shows that the non-collider orientation rule in ¶B is correct. If A and B, and B and C are p-adjacent, but **Sepset**(A,C) contains B, then we know from the search procedure that A and C are not d-separated given any subset of **Sepset**(A,C). It follows that B is an ancestor of A or C. Hence $A{*}\!\!-\!\!{*}B{*}\!\!-\!\!{*}C$ should be oriented as $A{*}\!-\!\underline{{*}B{*}}\!-\!{*}C$ in the PAG.

The following Lemma shows the correctness of the orientation rule in ¶B:

Lemma 8: In a graph G, if A and B are p-adjacent, B and C are p-adjacent, and B is an ancestor of A or C then A and C are d-connected given any set **S**, s.t. $A,B,C \notin \mathbf{S}$.

Proof. Without loss of generality, let us suppose that B is an ancestor of C. It is sufficient to prove that A and C are d-connected conditional on **S**. There are two cases to consider, depending upon whether or not some (proper) descendant of B is in **S**.

Case 1: Some (proper) descendant of B is in **S**. It follows from Lemma 4 and the p-adjacency of A and B that given any set **S** such that A, B, C∉**S**, there is a d-connecting path from A to B and likewise a d-connecting path from B to C, conditional on **S**. Since some descendant of B is in **S**, but B itself is not in **S**, it follows again by a simple application Lemma 1 that A and C are d-connected, since it does not matter whether or not the path from A to B and the path from B to C collide at B.

Case 2: No descendant of B is in **S**. It follows from Lemma 4 that there is a path d-connecting A and B. Since no descendant of B has been conditioned on the directed path B→…→C is d-connecting. Since B∉**S**, it follows from Lemma 1 that A and C are d-connected given **S**. ∴

It follows by contraposition that if A and B are p-adjacent, B and C are p-adjacent, A and C are d-separated given **Sepset**<A,C>, and B∉**Sepset**<A,C>, then B is not an ancestor of A or C, hence A∗—∗B∗—∗C should be oriented as A—>B<—C in the PAG.

Section ¶C

Lemma 9: In a graph *G*, suppose X is an ancestor of Y. If there is a set **S** such that A and Y are d-separated given **S**, X and Y are d-connected given **S**, and X∉**S**, then A and X are d-separated given **S**.

Proof. Let X be an ancestor of Y. Let **S** be any set such that X and Y are d-connected given **S**, X∉**S**, and A and Y are d-separated by **S**. Suppose, for a contradiction, that A and X are d-connected given **S**. It then follows that there is a d-connecting path *P* from A to X. There are now two cases:

Case 1: Some descendant of X is in **S**. Since X∉**S**, and some descendant of X is in **S**, it follows from Lemma 1 that we can put together the d-connecting path from A to X and the d-connecting path from X to Y given **S**, to form a d-connecting path from A to Y given **S**. This is a contradiction since we assumed that A and Y were d-separated given **S**.

Case 2: No descendant of X is in **S**. In this case since X is an ancestor of Y, there is a d-connecting directed path X→…→Y. Again, by Lemma 1 we can put together the d-connecting path from A to X and the d-connecting directed path from X to Y. This is again a contradiction since we assumed that A and Y were d-separated given **S**.

We have now shown that under the conditions in the antecedent, **S** is a d-separating set for A and X. ∴

Lemma 10: Let A, X and Y be three vertices in a graph *G*, such that X and Y are p-adjacent. If there is a set **S** such that

(i) X∉**S**,

(ii) A and Y are d-separated given **S**, and

(iii) A and X are d-connected given **S**,

then X is not an ancestor of Y.

Proof. Suppose that there is such a set **S**. If X and Y are p-adjacent then X and Y are d-connected by every subset of the other variables. In particular X and Y are d-connected given **S**. Since **S** d-separates A and Y but d-connects A and X, it follows from Lemma 9 that X is not an ancestor of Y.∴

Step ¶C simply applies Lemma 10. Suppose that <A,X,Y> is a triple such that:

(i) A is not p-adjacent to X or Y,

(ii) X and Y are p-adjacent in Ψ,

(iii) X ∉ **Sepset**<A,Y>.

¶C(i) is justified in the following way. Suppose that **Sepset**<A,Y> ⊂ **Sepset**<A,X>. Recall that the search procedure used in ¶A to find **Sepset**<A,X> tests every subset of **Sepset**<A,X> to see if it d-separates A and X, before testing **Sepset**<A,X>. In particular, if **Sepset**<A,Y> ⊂ **Sepset**<A,X>, then A and X are d-connected given **Sepset**<A,Y>, so taking **S** = **Sepset**<A,Y>, we can apply Lemma 10 to orient X ∗–∗Y as X<—Y.

¶C(ii) is justified in the following way. Suppose that A and X are d-connected given **Sepset**<A,Y>. Since X ∉ **Sepset**<A,Y>, setting **S** = **Sepset**<A,Y>, we can again apply Lemma 10 to orient X ∗–∗Y as X<—Y.

The condition in ¶C(ii) that **Sepset**<A,X> ⊆ **Sepset**<A,Y> is not needed to make ¶C(ii) correct (as evidenced by the fact that it plays no role in the justification of the rule); it is included in order to avoid carrying out a redundant test of d-separation. If **Sepset**<A,X> $\subsetneq$ **Sepset**<A,Y>, then A and X are not d-connected given **Sepset**<A,Y>. (This is because Y∉ **Sepset**<A,X>. Hence X∗–∗Y will eventually by another application of ¶C(i) be oriented as X—>Y in the PAG. It follows that X is an ancestor of Y in *G*. By Lemma 9, since X is an ancestor of Y in *G*, A and X are not d-connected given **Sepset**<A,Y>.) If **Sepset**<A,Y> = **Sepset**<A,X> then there is no need to test whether A and X are d-connected given **Sepset**<A,Y>, because it is already known that they are not d-connected (by definition of **Sepset**<A,X>).

It is a feature of this orientation rule that X and Y may be arbitrarily far from A. Rules of this type are needed by a cyclic discovery algorithm, because, as was shown in Richardson (1994b), two cyclic graphs may agree 'locally' on d-separation relations, but disagree on some d-separation relation between distant variables. (Whether or not such rules will ever be used on real data, in which 'distant' variables are generally found to be independent by statistical tests is another question.)

Section ¶D

This section searches to find 'extra' d-separating sets for unshielded colliders. In the acyclic case a triple of vertices X∗-∗Y∗-∗Z, where X and Y are p-adjacent, Y and Z are p-adjacent, but X and Z are not p-adjacent either has the property that every d-separating set for X and Z contains Y, or that every d-separating set for X and Z does not contain Y.[18] However, in the cyclic case it is possible for X and Z to be d-separated by one set containing Y, and one set not containing Y. We already know from Lemma 8 that if X and Z are d-separated by some set which does not contain Y, then Y is not an ancestor of X or Z. What can we infer if in addition X and Z are also d-separated by a set which contains Y? This is answered by the next Lemma and Corollary.

Lemma 11: If in a graph *G*, Y is a descendant of a common child of X and Z then X and Z are d-connected by any set containing Y.

Proof. Suppose that Y is a descendant of a common child C of X and Z. Then the path X→C←Z d-connects X and Z given any set containing Y. ∴

Corollary 4: If in a graph *G*, X and Y are p-adjacent, Y and Z are p-adjacent, but X and Z are not p-adjacent, Y is not an ancestor of X or Z, and there is some set **S** such that Y∈ **S**, and X and Z are d-separated given **S**, then Y is not a descendant of a common child of X and Z.

It follows from Lemma 12 that if <X,Y,Z> is a triple such that X and Z are d-connected given any set containing Y, and d-separated by some set not containing Y, then Y is a descendant of a common child of X and Z.

Lemma 12: If in graph *G*, Y is not a descendant of a common child of X and Z, then X and Z are d-separated by the set **T**, defined as follows:

S = Children(X) ∩ An({X,Y,Z})
T = (Parents(**S** ∪ {X}) ∪ **S**)\(Descendants(Children(X)∩Children(Z)) ∪ {X,Z})

Further, if X and Y, and Y and Z are p-adjacent then Y∈**T**.

Proof. It follows from Lemma 5, with **Q**={Y} that X and Z are d-separated given **T**.
All that remains is to show that Y∈ **T**. There are three cases to consider here:

---

[18] This is also true even in the acyclic case with latent variables.

Case 1: Y is a child of X. If Y is a child of X, then since Y is an ancestor of Y, Y$\in$ **S**. In both cases since Y is not a descendant of a common child of X and Z, Y$\in$**T**.

Case 2: Y is a parent of X. Since Y is a parent of X and Y is not a descendant of a common child of X and Z, Y$\in$**T**.

Case 3: X and Y have a common child C that is an ancestor of X or Y. Since C is a child of X and an ancestor of X or Y, C$\in$**S**. Since Y is a parent of C, and Y is not a descendant of a common child of X and Z then Y$\in$**T**. $\therefore$

Lemma 13: In directed graph *G*, if X and Z are d-separated by some set **R**, then for all sets **Q** $\subseteq$ An(**R** $\cup$ {X,Z})\{X,Z}, X and Z are d-separated by **R** $\cup$ **Q**.

Proof. Suppose for a contradiction that there is a path *P* d-connecting X and Z given **R** $\cup$ **Q**. It follows that every vertex on *P* is an ancestor of either X, Z, or **R** $\cup$ **Q**. Since **Q** $\subseteq$ An(**R** $\cup$ {X,Z}) it follows that every vertex on *P* is an ancestor of X, Z or **R**. Let A be the collider furthest from X on *P* which is an ancestor of X and not **R** (or X if no such collider exists), let B be the first collider after A on *P* which is an ancestor of Z and not **R** (or Z if no such collider exists). Clearly the paths X$\leftarrow\ldots\leftarrow$A, and B$\rightarrow\ldots\rightarrow$Z are d-connecting given **R**, since by the definition of A and B, no vertex on these paths is in **R**. In addition the subpath of **P** between A and B is also d-connecting given **R** since every collider is an ancestor of **R**, and no non-collider lies in **R**, since, by hypothesis *P* d-connects given **R** $\cup$ **Q**. It follows, by Lemma 1, that there is a path d-connecting X and Z given **R**. This is a contradiction.$\therefore$

The search in section ¶D considers in turn each triple A—>B<—C in Ψ, A and C are not p-adjacent, and attempts to find a set **R** which is a subset of **Local**(Ψ,A)\{C} such that A and C are d-separated given **R** $\cup$ {B} $\cup$ **Sepset**<A,C>. It follows from Lemma 11, that if there is some set which d-separates A and C, and contains B, then B is not a descendant of a common child of A and C. It then follows from Lemma 12 that in this case there is some subset, the set **T** given in the Lemma, which contains B, d-separates A and C and in which every vertex is either a parent of A, a child of A, or a parent of a child of A and so **T** $\subseteq$ **Local**(Ψ,X). Since **Sepset**<A,C> is a minimal d-separating set for A and C, it follows that **Sepset**<A,C> $\subseteq$ An({A,C})\{A,C} ($\subseteq$ An(**T** $\cup${A,C}) . Hence by Lemma 13, **T** $\cup$ **Sepset**<A,C> also d-separates A and C.

The reader may wonder why ¶D tests sets of the form **T** $\cup$ **Sepset**<A,C>, (where **T** $\subseteq$ **Local**(Ψ,A)), instead of just testing sets of the form **T** $\subseteq$ **Local**(Ψ,A)); Lemma 12 shows that a search of the latter kind would succeed in finding a d-separating set for A and C which contained B. The answer is that from Lemma 13 we know that any set

T ⊆ **Local**(Ψ,A) which d-separates A and C is such that **T** ∪ **Sepset**<A,C> also d-separates A and C, but the reverse is not true. In particular the smallest set **T** such that **T** ∪ **Sepset**<A,C> d-separates A and C may be considerably smaller than the smallest set **T** which d-separates A and C alone, hence the search is significantly faster.[19]

We require one more lemma to explain why we initialize m = 1, and do not test **T** = ∅:

Lemma 14: In directed graph *G*, if X and Y are p-adjacent, Y and Z are p-adjacent, X and Z are not p-adjacent, Y is not an ancestor of X or Z, and **S** is a minimal d-separating set for X and Z then X and Z are d-connected given **S** ∪ {Y}.

Proof. According to Corollary 1, if X and Y are p-adjacent then either X→Y, Y→X or X→C←Y, where C is an ancestor of X or Y. Thus under the hypothesis that Y is not an ancestor of X it follows that X is an ancestor of Y. Moreover, it follows that there is a directed path *P* from X to Y, on which every vertex except X is a descendant of Y, and hence on which every vertex except X is not an ancestor of X or Z. (In the case X→Y, the last assertion is trivial. In the other case it merely states a property of the path X→C→…Y, where C is a common child of X and Y.) Likewise there is a path **Q** from Z to Y on which every vertex except Z is not an ancestor of X or Z.

If **S** is a minimal d-separating set every vertex in **S** is an ancestor of X or Z, (and X,Z ∉ **S**). Hence no vertex on *P* or *Q* is in **S**. It follows that *P* d-connects X and Y given **S**, and *Q* d-connects Y and Z given **S**. It then follows from Lemma 1 that these paths can be joined to form a single d-connecting path, hence X and Z are d-connected given **S** ∪ {Y}.∴

This completes the proof that step ¶D of the algorithm will succeed in finding a set which d-separates A and C, and contains B, for each triple A—>B<—C in the PAG, if any such set exists.

Section ¶E

The following Lemma provides the justification of ¶E where A—>B<—C, A—>D<—C, and D is not in **SupSepset**<A,B,C>, in which case B∗–∗D is oriented as B—>D.

---

[19]In some cases the cardinality of the smallest set (**T** ∪ **Sepset**<A,C>) may be greater than the cardinality of the smallest **T**; but this is not true in general, and since we only intend to discover linear models this is insignificant. (With discrete models conditioning on a large set of variables in a conditional independence test may reduce dramatically the power of the test.)

Lemma 15: If in a PAG Ψ for *G*, X—>V<—Z, X—>W<—Z, X and Z are not p-adjacent, and W is an ancestor of V in *G*, then any set **S** such that V∈ **S**, and X and Z are d-separated by **S**, also contains W.

Proof. Suppose there were some d-separating set **S** for X and Z which contained V and did not contain W. Then, since W is an ancestor of V and V∈ **S**, but W∉ **S**, it follows by Lemma 1 that we could put together a d-connecting path from X to W and from W to Z to form a new d-connecting path (irrespective of whether or not these paths collide at W). Such d-connecting paths between X and W, and between W and Z exist (by Corollary 1) since X is p-adjacent to W and W is p-adjacent to Z. This is a contradiction.∴

In the case in which A—>B<—C, A—>D<—C, and D is in **SupSepset**<A,B,C> the algorithm orients B∗–∗D as B∗—D, this inference can be justified as follows. If D is in **SupSepset**<A,B,C> then it follows from Lemma 7 and the nature of the search of **SupSepset**<A,B,C>[20] that D is an ancestor of {B} ∪ **Sepset**<A,C>. Since **Sepset**<A,C> is a minimal d-separating set for A and C, every vertex in **Sepset**<A,C> is an ancestor of A or C. Thus if D is in **SupSepset**<A,B,C>, then D is an ancestor of A,C or B. However, since there are arrowheads at D on the edges from A to D, and C to D, it follows that D is not an ancestor of A or C, and hence D is an ancestor of B. Thus it is correct to orient B∗–∗D as B∗—D.

In the case in which A—>D<—C in Ψ, (A and C are not p-adjacent and there is no dotted line A—>D<—C), it follows from Lemma 12 by contraposition that since A and C are d-connected by any set **S** that contains D, (and does not contain A or C), D is a descendant of a common child of A and C. Moreover since A and C are d-separated by some set containing B, B is not a descendant of a common child of A and C. Hence B is not a descendant of D. Thus in the case where in Ψ, A—>B<—C, A—>D<—C, B and D are p-adjacent, B*–*D should be oriented as B<—D.

<u>Section **¶F**</u>

A and C are d-separated by **SupSepset**<A,B,C>, and B∈ **SupSepset**<A,B,C>. Hence by Lemma 13, if D is an ancestor of B, then A and C are d-separated by **SupSepset**<A,B,C> ∪ {D}. Hence by contraposition, if A and C are d-connected given **SupSepset**<A,B,C> ∪ {D} then D is not an ancestor of B. (In fact, it follows that D is not an ancestor of A,B or C.) Since D is not an ancestor of B, but B and D are p-adjacent it follows that B is an ancestor of D. Thus B∗—∗D should be oriented as B—>D in Ψ.

[20]Namely the fact that section ¶D looks for the smallest superset of {B} ∪ **Sepset**<A,C>, which d-separates A and C.

This completes the proof of the correctness of the CCD algorithm. ∴

7.6. Proof of Theorem 8

In order to prove the d-separation completeness of the CCD algorithm, all that is required is to show that whenever the first input to the CCD algorithm is a d-separation oracle for G_1 that results in output Ψ_1, and the second input to the CCD algorithm is a d-separation oracle for G_2 that results in output Ψ_2, and Ψ_1 and Ψ_2 are identical, then G_1 and G_2 are d-separation equivalent. We shall do this by proving that when d-separation oracles for G_1 and G_2 are used as input to the CCD algorithm and produce the same PAG as output, then G_1, and G_2 satisfy the five conditions of the Cyclic Equivalence Theorem CET(I)-(V) (given below) with respect to one another. It has already been shown in Richardson(1994b) that two graphs G_1 and G_2 are d-separation equivalent to one another if and only if they satisfy these 5 conditions.

Before stating the Cyclic Equivalence Theorem we require a number of extra definitions. In a cyclic graph G, we say triple of vertices <A,B,C> forms an **unshielded conductor** if:

(i) A and B are p-adjacent, B and C are p-adjacent, A and C are not p-adjacent
(ii) B is an ancestor of A or C

If <A,B,C> satisfies (i), but B is not an ancestor of A or C, we say <A,B,C> is an **unshielded non-conductor.**

In a cyclic graph G, we say triple of vertices <A,B,C> is an **unshielded perfect non-conductor** if

(i) A and B are p-adjacent, B and C are p-adjacent, but A and C are not p-adjacent,
(ii) B is not an ancestor of A or C, and
(iii) B is a descendant of a common child of A and C.

If <A,B,C> satisfies (i) and (ii) but B is not a descendant of a common child of A and C, we say <A,B,C> is an **unshielded imperfect non-conductor**.

If $\langle X_0,X_1,\ldots X_{n+1}\rangle$ is a sequence of distinct vertices s.t. $\forall i\ 0 \le i \le n$, X_i and X_{i+1} are p-adjacent then we will refer to $\langle X_0,X_1,\ldots X_{n+1}\rangle$ as an **itinerary**.

If $<X_0,\ldots X_{n+1}>$ is an itinerary such that:

(i) $\forall t\ 1\le t \le n$, $<X_{t-1}, X_t, X_{t+1}>$ is an unshielded conductor,

(ii) $\forall k\ 1\le k \le n$, X_{k-1} is an ancestor of X_k, and X_{k+1} is an ancestor of X_k, and

(iii) X_0 is *not* a descendant of X_1, and X_n is *not* an ancestor of X_{n+1},

then $<X_0,X_1,X_2>$ and $<X_{n-1},X_n,X_{n+1}>$ are **mutually exclusive (m.e.) unshielded conductors on the itinerary** $<X_0,\ldots X_{n+1}>$.

If $<X_0,\ldots X_{n+1}>$ is an itinerary such that $\forall i,j\ 0 \le i < j-1 < j \le n+1$ X_i and X_j are not p-adjacent in the graph then we say that $<X_0,\ldots X_{n+1}>$ is an **uncovered itinerary.**. i.e. an itinerary is uncovered if the only vertices on the itinerary which are p-adjacent to other vertices on the itinerary, are those that occur consecutively on the itinerary.

Theorem 9: (**Cyclic Equivalence Theorem,** Richardson 1994b) Graphs G_1 and G_2 are d-separation equivalent if and only the following five conditions hold:

CET(I) G_1 and G_2 have the same p-adjacencies,

CET(II) G_1 and G_2 have the same unshielded elements i.e.

(IIa) the same unshielded conductors, and

(IIb) the same unshielded perfect non-conductors,

CET(III) For all triples $<A,B,C>$ and $<X,Y,Z>$, $<A,B,C>$ and $<X,Y,Z>$ are m.e. conductors on some uncovered itinerary $P \equiv<A,B,C,\ldots X,Y,Z>$ in G_1 if and only if $<A,B,C>$ and $<X,Y,Z>$ are m.e. conductors on some uncovered itinerary $Q \equiv<A,B,C,\ldots X,Y,Z>$ in G_2 ,

CET(IV) If $<A,X,B>$ and $<A,Y,B>$ are unshielded imperfect non-conductors (in G_1 and G_2), then X is an ancestor of Y in G_1 iff X is an ancestor of Y in G_2,

CET(V) If $<A,B,C>$ and $<X,Y,Z>$ are mutually exclusive conductors on some uncovered itinerary $\mathbf{P} \equiv<A,B,C,\ldots X,Y,Z>$ and $<A,M,Z>$ is an unshielded imperfect non-conductor (in G_1 and G_2), then M is a descendant of B in G_1 iff M is a descendant of B in G_2.

Lemma 16: Given a sequence of vertices $<X_0,\ldots X_{n+1}>$ in a directed graph G having the property that $\forall k$, $0 \le k \le n$, X_k is an ancestor of X_{k+1}, and X_k is p-adjacent to X_{k+1} there is a subsequence of the X_i's, which we label the Y_j's having the following properties:

(a) $X_0\equiv Y_0$

(b) $\forall j$, Y_j is an ancestor of Y_{j+1}

(c) $\forall j,k$ If $j < k$, Y_j and Y_k are p-adjacent in the graph if and only if $k = j+1$. i.e. the only Y_k's which are p-adjacent are those that occur consecutively.

Proof. The Y_k's can be constructed as follows:

Let $Y_0 \equiv X_0$.

Let $Y_{k+1} \equiv X_\eta$ where η is the greatest $h > j$ such that X_h is p-adjacent to X_j where $X_j \equiv Y_k$.

Property (a) is immediate from the construction. Property (b) follows from the transitivity of the ancestor relation, and the fact that the Y_k's are a subsequence of the X_i's. It is also clear, from the construction that if $k = j+1$ then Y_j and Y_k are p-adjacent. Moreover, if $Y_j \equiv X_\alpha$[21] and $Y_k \equiv X_\beta$ are p-adjacent, and $j < k$, then it follows again from the construction that if $Y_{j+1} \equiv X_\gamma$, then $\beta \leq \gamma$, so $k \leq j+1$. (This is because the Y_k's are a subsequence of the X_i's.) Hence $Y_{j+1} \equiv Y_k$.∴

Lemma 17: Let G_1 and G_2 be two graphs satisfying CET(I)–(III) Suppose there is a directed path $D_1 \rightarrow \ldots D_n$, in G_1. Let D_0 be a vertex distinct from $D_1,\ldots,D_n$, s.t. D_0 is p-adjacent to D_1 in G_1 and G_2, D_0 is not p-adjacent to $D_2,\ldots D_n$ in G_1 or G_2 and D_0 is not a descendant of D_1 in G_1 or G_2. It then follows that D_1 is an ancestor of D_n in G_2.

Proof. By induction on n.

Base Case: n = 2. Since, by hypothesis, D_0 is not p-adjacent to D_2, it follows that $\langle D_0, D_1, D_2\rangle$ forms an unshielded conductor in G_1 (since D_1 is an ancestor of D_2). Hence this triple of vertices also forms an unshielded conductor in G_2, by CET(IIa). Hence D_1 is an ancestor of D_0 or D_2 in G_2. Since, by hypothesis D_1 is not an ancestor of D_0 in G_2, it follows that D_1 is an ancestor of D_2 in G_2.

Induction Case: Suppose that the hypothesis is true for paths of length n. It follows from Lemma 16 that there is a subsequence $\langle D_{\alpha(0)} \equiv D_0, D_{\alpha(1)}, D_{\alpha(2)} \ldots D_{\alpha(r)} \equiv D_n\rangle$ such that the only p-adjacent vertices are those that occur consecutively, and in G_1 each vertex is an ancestor of the next vertex in the sequence. Moreover, since, by hypothesis, D_0 is not p-adjacent to $D_2,\ldots D_n$, it follows that $D_{\alpha(1)} \equiv D_1$. Since G_1 and G_2 satisfy CET(I), they have the same p-adjacencies, hence in G_2 the only vertices that are p-adjacent are those that occur consecutively in the sequence. Suppose, for a contradiction that $D_{\alpha(r-1)}$ is not an ancestor of $D_{\alpha(r)}$ in G_2. Let s be the smallest j such that $D_{\alpha(j)}$ is not an ancestor of $D_{\alpha(j-1)}$ in G_2. (Such a j exists since $D_{\alpha(1)} \equiv D_1$ and $D_{\alpha(0)} \equiv D_0$ is not a descendant of D_1.) It then follows that $\langle D_{\alpha(s-1)}, D_{\alpha(s)}, D_{\alpha(s+1)}\rangle$ and $\langle D_{\alpha(r-2)}, D_{\alpha(r-1)}, D_{\alpha(r)}\rangle$ are mutually exclusive conductors on the unshielded itinerary $\langle D_{\alpha(s-1)}, \ldots D_{\alpha(r)}\rangle$ in G_2. But these two triples are not mutually exclusive in G_1 since $D_{\alpha(r-1)}$ is an ancestor of $D_{\alpha(r)}$ in G_1; hence G_1 and G_2 fail to satisfy CET(III), which is a contradiction.

---

[21] That is, the j^{th} vertex in the sequence of Y vertices is the α^{th} vertex in the sequence of X vertices.

It follows that $D_{\alpha(r-1)}$ is an ancestor of $D_{\alpha(r)}$ in G_2. It then follows from the induction hypothesis that D_1 is an ancestor of $D_{\alpha(r)} \equiv D_n$. ∴

Theorem 8: (d-separation Completeness) If the CCD algorithm, when given as input d-separation oracles for the graphs G_1, G_2 produces as output PAGs Ψ_1, Ψ_2 respectively, then Ψ_1 is identical to Ψ_2 if and only if G_1 and G_2 are d-separation equivalent, i.e. $G_2 \in$ **Equiv**(G_1) and vice versa.

Proof. We will show that if two graphs, G_1 and G_2 are *not* d-separation equivalent, then the PAGs output by the CCD algorithm, given d-separation oracles for G_1 and G_2 as input, would differ in some respect.

It follows from the Cyclic Equivalence Theorem that if G_1 and G_2 are not d-separation equivalent, then they fail to satisfy one or more of the five conditions CET(I)-(V). Let Ψ_1 and Ψ_2 denote, respectively, the PAGs output by the CCD algorithm when given d-separation oracles for G_1 and G_2 as input.

Case 1: G_1 and G_2 fail to satisfy CET(I). In this case the two graphs have different p-adjacencies. Let us suppose without loss of generality that there is some pair of variables, X and Y which are p-adjacent in G_1 and not p-adjacent in G_2. Since X and Y are p-adjacent in G_1, X and Y are d-connected conditional upon any subset of the other vertices. Hence there is an edge between X and Y in Ψ_1.

Since X and Y are not p-adjacent in G_2, there is some subset **S**, (X,Y∉**S**) such that X and Y are d-separated in G_2 given **S**. It follows from Lemma 6 that X and Y are d-separated by a set of variables **T**, such that either **T** is a subset of the vertices p-adjacent to X, or **T** is a subset of the vertices p-adjacent to Y. It follows that in step ¶A of the CCD algorithm the edge between X and Y in Ψ_2 would be removed. Since edges are not added back in at any later stage of the algorithm, there is no edge in Ψ_2 between X and Y. Hence Ψ_1 and Ψ_2 are different.

Case 2: G_1 and G_2 fail to satisfy CET(IIa). We assume that G_1 and G_2 satisfy CET(I). In this case the two graphs have different unshielded non-conductors. Thus we may assume, without loss of generality, that there is some triple of vertices <X,Y,Z> such that in G_1, Y is an ancestor of X or Z, while Y is not an ancestor of either X or Z in G_2.

If Y is an ancestor of X or Z then it follows from Lemma 8 that every set which d-separates X and Z includes Y. Hence Y∈**Sepset**(X,Z) in G_1. It then follows from the correctness of the algorithm that in Ψ_1, either X—>Y—∗Z, X∗—Y<—Z, or X∗–$\underline{*Y*}$–∗Z.

If Y is not an ancestor of X or Z in G_2, then Y is not in any minimal d-separating set for X and Z. In particular Y∉ **Sepset**(X,Z) for G_2. Again it follows from the correctness of the algorithm that <X,Y,Z> is oriented as X∗–>Y<–∗Z or X∗–>Y<–∗Z in Ψ_2. Thus Ψ_1 and Ψ_2 are different.

Case 3: G_1 and G_2 fail to satisfy CET(IIb). We assume that G_1 and G_2 satisfy CET(I), CET(IIa). In this case the two graphs have different unshielded imperfect non-conductors, i.e. there is some triple <X,Y,Z> such that it forms an unshielded non-conductor in both G_1 and G_2, but in one graph Y is a descendant of a common child of X and Z, while in the other graph it is not. Let us assume that Y is a descendant of a common child of X and Z in G_1, while in G_2 it is not.

It follows from Lemma 11 that in G_1, X and Z are d-connected given any subset containing Y. In this case the search in CCD section ¶D will fail to find any set **Supsepset**<X,Y,Z>. Hence <X,Y,Z> will be oriented as X –>Y<– Z (i.e. without dotted underlining) in Ψ_1. If Y is not a descendant of a common child of X and Z, then it follows from Lemma 12 and Lemma 13 that there is some subset **T** of **Local**(Ψ,X), such that X and Z are d-separated given **T** ∪ {Y} ∪ **Sepset**<X,Z>. Section ¶D will find such a set **T**, and hence <X,Y,Z> will be oriented as X∗–>Y<–∗Z in Ψ_2. Since no subsequent orientation rule removes or adds dotted underlining, it follows that Ψ_1 and Ψ_2 are different.

Case 4: G_1 and G_2 fail to satisfy CET(III). We assume that G_1 and G_2 satisfy CET(I), CET(IIa), CET(IIb). In this case the two graphs have the same p-adjacencies, and the same unshielded conductors, perfect non-conductors, and imperfect non-conductors. However, the two graphs have different mutually exclusive conductors. Hence in both G_1 and G_2 there is an uncovered itinerary, <X_0,…X_{n+1}> such that every triple <X_{k-1},X_k,X_{k+1}> ($1 \le k \le n$) on this itinerary is a conductor, but in one graph <X_0,X_1,X_2> and <X_{n-1},X_n,X_{n+1}> are mutually exclusive, i.e. X_1 is not an ancestor of X_0, and X_n is not an ancestor of X_{n+1}, while in the other they are not mutually exclusive. Let us suppose without loss of generality that <X_0,X_1,X_2> and <X_{n-1},X_n,X_{n+1}> are mutually exclusive in G_1, while in G_2 they are not.

It follows from the definition of m.e. conductors that the vertices X_1,…X_n, inclusive are *not* ancestors of X_0 or X_{n+1} in G_1. Hence {X_1,…X_n} ∩ **Sepset**(X_0,X_{n+1}) = ∅, since **Sepset**(X_0,X_{n+1}) is minimal, and so is a subset of An(X_0,X_{n+1}). (**Sepset**(X_0,X_{n+1}) is calculated for G_1.) For the same reason Descendants({X_1,…X_n}) ∩ **Sepset**(X_0,X_{n+1}) = ∅. It follows from the definition of a pair of m.e. conductors on an itinerary that X_k is an ancestor of X_{k+1} ($1 \le k < n$), thus there is a directed path $P_k \equiv X_k \rightarrow \ldots \rightarrow X_{k+1}$. Since no descendant of $X_1, \ldots, X_n$ is in **Sepset**(X_0,X_{n+1}), each of the directed paths P_k d-connects each vertex X_k to its successor X_{k+1} ($1 \le k < n$), conditional on **Sepset**(X_0,X_{n+1}). In addition, since X_0 and X_1 are p-adjacent there is some path Q d-connecting X_0 and X_1 given **Sepset**(X_0,X_{n+1}). Since each P_i is out of X_i (i.e. the path goes $X_i \rightarrow \ldots \rightarrow X_2$), by applying Lemma 1, with $\mathcal{T} = \{Q, P_1, \ldots P_n\}$, $R = \langle X_0, \ldots X_n \rangle$, and $\mathbf{S}$ = **Sepset**(X_0,X_{n+1}) that we can form a path d-connecting X_0 and X_n given **Sepset**(X_0,X_{n+1}). A symmetric argument shows that X_1 and X_{n+1} are also d-connected given **Sepset**(X_0,X_{n+1}). It then follows that the edges X_0*—*X_1 and X_n*—*X_{n+1} are oriented as X_0—>X_1 and X_n<—X_{n+1} by stage ¶C of the CCD algorithm (unless they have already been oriented this way in a previous stage of the algorithm). Thus again, by the correctness of the algorithm these arrowheads will be present in Ψ_1. (Subsequent stages of the algorithm only add '–' and '>' endpoints, not 'o' endpoints. If either of the arrowhead at X_1 or X_n were replaced with a '–' the algorithm would be incorrect.)

Since by hypothesis, $\langle X_0, X_1, X_2 \rangle$ and $\langle X_{n-1}, X_n, X_{n+1} \rangle$ are not mutually exclusive in G_2, either X_1 is an ancestor of X_0, or X_n is an ancestor of X_{n+1}. It follows from the correctness of the orientation rules in the CCD algorithm that the edges X_0*—*X_1 and X_n*—*X_{n+1} will not both be oriented as X_0*—>X_1 and X_n<—*X_{n+1} in Ψ_2. Thus Ψ_1 and Ψ_2 will once again be different.

Case 5: G_1 and G_2 fail to satisfy either CET(IV) or CET(V). We assume that G_1 and G_2 satisfy CET(I)–(III).[22] If G_1 and G_2 fail to satisfy either CET(IV) or CET(V), then in either case we have the following situation: There is some sequence of vertices in G_1 and G_2 $\langle X_0, X_1, \ldots X_n, X_{n+1} \rangle$, [23] satisfying the following:

(a) if $i > j$ then X_i and X_j are p-adjacent if and only if $i = j+1$,

(b) X_1 is not an ancestor of X_0, and X_n is not an ancestor of X_{n+1}, and

(c)$\forall k$, $1 \le k \le n$, X_{k-1}, and X_{k+1} are ancestors of X_k.

In addition there is some vertex V, p-adjacent to X_0 and X_{n+1} in G_1 and G_2 , not an ancestor of X_0 or X_{n+1} in G_1 or G_2 and not a descendant of a common child of X_0 and X_{n+1} in G_1 or G_2. As explained in case 3, this implies that in both of the PAGs Ψ_1 and Ψ_2, X_0—>V<—X_{n+1}.

[22]The conditions under which CET(IV) or CET(V) fail are quite intricate precisely because the assumption that CET(I)-(III) are satisfied implies that the graphs agree in many respects.

[23] In the case where CET(IV) fails n=1, while if CET(V) fails, n>1.

Since G_1 and G_2 fail to satisfy CET(IV) or CET(V), in one graph V is a descendant of X_1, while in the other graph V is not a descendant of X_1. Let us suppose without loss of generality that V is a descendant of X_1 in G_1, and V is not a descendant of X_1 in G_2. As in previous cases it is sufficient to show that if Ψ_1 and Ψ_2 are the CCD PAGs corresponding to G_1 and G_2 respectively, then Ψ_1 and Ψ_2 are different. We may suppose, again without loss of generality that V is the closest such vertex to any X_k ($1 \le k \le n$) in G_1, in the sense that the shortest directed path $P \equiv X_k \rightarrow \ldots \rightarrow V$ in G_1 contains at most the same number of vertices as the shortest directed path in G_1 from any X_k ($1 \le k \le n$) to some other vertex V' satisfying the conditions on V.

Claim: Let W be the first vertex on P which is p-adjacent to V, (both in G_1 and G_2 since by CET(I) G_1 and G_2 have the same p-adjacencies). We will show that the assumption that V is the closest such vertex to any X_k (in G_1) together with the assumption that G_1 and G_2 satisfy CET(I)-(III) imply that W is a descendant of X_1 in G_2. We prove this by showing that every vertex in the directed subpath $P(X_k, W) \equiv X_k \rightarrow \ldots W$ in G_1 is also a descendant of X_1 in G_2.

Proof of Claim: By induction on the vertices occurring on the path $P(X_k, W)$.

Base Case: X_k. By hypothesis X_k is a descendant of X_1 in both G_1 and G_2.

Induction Case: Consider Y_r, where $P(X_k, W) \equiv X_k \rightarrow Y_1 \rightarrow \ldots \rightarrow Y_r \rightarrow \ldots Y_t \equiv W$. By the induction hypothesis, for $s < r$, Y_s is a descendant of X_1 in G_2. Now there are two subcases to consider:

Subcase 1: Not both X_0 and X_{n+1} are p-adjacent to Y_r. Suppose without loss that X_0 is not p-adjacent to Y_r. Since in G_1 there is a directed path $X_0 \rightarrow \ldots X_k \rightarrow Y_1 \rightarrow \ldots Y_r$, by Lemma 16 it then follows that there is some subsequence of this sequence of vertices, $Q \equiv \langle X_0, \ldots Y_r \rangle$ such that consecutive vertices in Q are p-adjacent, but only these vertices are p-adjacent. Moreover, since X_0 is not p-adjacent to Y_r, this sequence of vertices is of length greater than 2, i.e. $Q \equiv \langle X_0, D \ldots Y_r \rangle$ where D is the first vertex in the subsequence after X_0, hence either $D \equiv X_\kappa$ ($1 \le \kappa \le k$) or $D \equiv Y_\mu$, ($1 \le \mu < r$). Since in either case D is a descendant of X_1 in both G_1 and G_2, (either by the induction hypothesis or by the hypothesis of case 5), but X_0 is not a descendant of X_1 in G_1 or G_2 it follows that D is not an ancestor of X_0 in G_1 or G_2. Hence we may apply Lemma 17, to deduce that Y_r is a descendant of D. Hence Y_r is a descendant of X_1, since X_1 is an ancestor of D.

Subcase 2: Both X_0 and X_{n+1} are p-adjacent to Y_r. First note that in G_1 the vertex Y_r is a descendant of X_k, and X_k is not an ancestor of X_0 or X_{n+1}. It follows that Y_r is not an ancestor of X_0 or X_{n+1} in G_1 . Moreover, since X_0 and X_{n+1} are not p-adjacent, $\langle X_0, Y_r,$

X_{n+1}> forms an unshielded non-conductor in G_1. Hence <X_0, Y_r, X_{n+1}> forms an unshielded non-conductor in G_2, since by hypothesis G_1 and G_2 satisfy CET(IIa). So Y_r is not an ancestor of X_0 or X_{n+1} in G_1 or G_2. Further, since Y_r is an ancestor of V in G_1 and V is not a descendant of a common child of X_0 and X_{n+1} in G_1, it follows that Y_r is not a descendant of a common child of X_0 and X_{n+1} in G_1. Thus <X_0, Y_r,X_{n+1}> forms an unshielded imperfect non-conductor in G_1. Since G_1 and G_2 satisfy CET(IIb), <X_0, Y_r,X_{n+1}> forms an unshielded imperfect non-conductor in G_2. Now, if Y_r were not a descendant of X_1 in G_2 , then Y_r would satisfy the conditions on V, yet be closer to X_k than V (Y_r occurs before V on the shortest directed path from X_k to V in G_1). This is a contradiction, hence Y_r is a descendant of X_k in G_2.

This completes the proof of the claim. We now show that Ψ_1 and Ψ_2 are different.

Consider the edge $W *\!-\!* V$ in Ψ_1. In G_1, W is an ancestor of V, hence it follows from the correctness of the algorithm in Ψ_1 this edge is oriented as $W \circ\!-\!* V$ or $W —\!* V$. In G_2, however, since X_1 is not an ancestor of V, but, as we have just shown X_1 is an ancestor of W, it follows that W is not an ancestor of V. There are now two cases to consider:

Subcase 1: n = 1 and $W \equiv X_1$. In this case $X_0 \rightarrow X_1 \leftarrow X_2$, in Ψ_2 (and Ψ_1). **Supsepset**(X_0,V,X_2) is the smallest set containing **Sepset**(X_0,X_2)∪{V} which d-separates X_0 and X_2, in the sense that no subset of **Supsepset**(X_0,V,X_2) which contains **Sepset**(X_0,X_2) ∪{V} d-separates X_0 and X_2. It follows from Lemma 7 (with **R** = **Sepset**(X_0,X_2) ∪{V}) that every vertex in **Supsepset**(X_0,V,X_2) is an ancestor of X_0, X_2 or **Sepset**(X_0,X_2) ∪{V}. Since every vertex in **Sepset**(X_0,X_2) is an ancestor of X_0 or X_2, it follows that every vertex in **Supsepset**(X_0,V,X_2) is an ancestor of X_0, X_2, or V. X_1 is not an ancestor of X_0, X_2, or V in G_2. Hence in step ¶D of the algorithm given a d-separation oracle for G_2 as input $X_1 \notin$ **Supsepset**(X_0,V,X_2). Thus step ¶E of the CCD algorithm will orient $W *\!-\!* V$ in Ψ_2 as $W <\!-\!* V$ (unless they have already been oriented this way in a previous stage of the algorithm). Thus Ψ_1 and Ψ_2 are not the same.

Subcase 2: n > 1, or W is not equal to X_1.

Claim: X_0 and X_{n+1} are d-connected given **Supsepset**(X_0,V,X_{n+1})∪{W} in G_2.

Proof. We have already shown that W is a descendant of X_1, and so also of X_n in G_1 and G_2. Since in both G_1 and G_2 X_0 is p-adjacent to X_1, but X_1 is not an ancestor of X_0, it follows that X_0 is an ancestor of X_1. Hence in both G_1 and G_2 there is a directed path P_0 from X_0 to X_1 on which every vertex except for X_0 is a descendant of X_1. (In the case $X_0 \rightarrow X_1$, the last assertion is trivial. In the case where X_0 and X_1 have a common child that is an ancestor of X_0 or X_1, and X_1 is not an ancestor of X_0, it merely states a property of the path $X_0 \rightarrow C \rightarrow \ldots X_1$, where C is a common child of X_0 and X_1.) Since W is a descendant of X_1, it follows that there is a directed path P_1 from X_1 to W. Concatenating P_0 and P_1 we construct a directed path P^* from X_0 to W on which every vertex except X_0 is a descendant of X_1. Since X_1 is not an ancestor of X_0, X_{n+1} or V, it follows that no vertex on P^*, except X_0, is an ancestor of X_0, X_{n+1} or V. Similarly we can construct a path from Q^* from X_{n+1} to W on which no vertex, except X_{n+1}, is an ancestor of X_0, X_{n+1} or V.

Since every vertex in **Supsepset**(X_0,V,X_{n+1}) is an ancestor of X_0, X_{n+1} or **Sepset**(X_0,X_{n+1})$\cup${V}, it follows as before that every vertex in **Supsepset**(X_0,V,X_{n+1}) is an ancestor of X_0, X_{n+1} or V. Thus no vertex in **Supsepset**(X_0,V,X_{n+1}) lies on P^* or Q^* (X_0, $X_{n+1} \notin$ **Supsepset**(X_0,V,X_{n+1}) by definition). It now follows by Lemma 1 that we can concatenate P^* and Q^* to form a path which d-connects X_0 and X_{n+1} given **Supsepset**(X_0,V,X_{n+1})$\cup${W}.

It follows directly from this claim that step ¶F of the CCD algorithm will orient V*–*W as V—>W in Ψ_2 (unless they have already been oriented this way in a previous stage of the algorithm). Hence Ψ_1 and Ψ_2 are different.

Since Cases 1-5 exhaust the possible ways in which G_1 and G_2 may fail to satisfy CET(I)-(V), this completes the proof that the CCD algorithm locates the d-separation equivalence class. $\therefore$

8. References

Aho, A.V., Hopcroft, J.E., & Ullman, J.D., (1974) The Design and Analysis of Computer Algorithms. Addison-Wesley, Reading Mass.

Basmann, R. (1965) A note on the statistical testability of ‘explicit causal chains’ against the class of ‘interdependent’ models.JASA, 60, 1080-1093.

Bekker, P.A., Merckens, A., Wansbeek, T.J., (1994) Identification, equivalent models, and computer algebra. Boston : Academic Press.

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

Cox, D.R., and Wermuth, N. (1993) Linear dependencies represented by chain graphs. In Statistical Science, 1993, 8 No.3 , 204-283.

Dawid, A. P. (1979) Conditional Independence in statistical theory (with discussion) Journal Royal Statistical Society Ser. B 41, 1-31.

Frydenberg, M., (1990) The chain graph Markov property,Scandinvaian Journal of Statistics, 17, 333-353.

Fisher, M.F. and Nagin, D. (1978) On the Feasibility of Identifying The Crime Function in a Simultaneous Model of Crime Rates and Sanction Levels. In Deterrence and Incapacitation: Estimating the Effects of Criminal Sanctions on Crime, Blumstein, Cohen and Nagin eds.

Geiger, D. (1990). Graphoids: a qualitative framework for probabilistic inference. PhD dissertation, Univ. California, Los Angeles.

Geiger, D., and Pearl, J., (1988) Logical and Algorithmic properties of Conditional Independence. Technical Report R-97, Cognitive Systems Laboratory, University of California, Los Angeles.

Goldberger, A., Duncan, O. (eds.), (1973) Structural Equation Models in the Social Sciences (Seminar Press, New York).

Goldberger, A. S. (1964). Econometric Theory. Wiley, New York.

Haavelmo, T., (1943) The statistical implications of a system of simultaneous equations, Econometrica, 11, 1-12.

Heise D.(1975). Causal Analysis. Wiley, New York.

Huggins, W.H. and Entwistle, D.R.(1968) Introductory Systems and Design, Ginn Blaisdell, Waltham Massachussetts.

Kiiveri, H. and Speed, T., (1982) Structural analysis of multivariate data: A review, Sociological Methodology, Leinhardt, S. (ed.). Jossey-Bass, San Francisco.

Kiiveri, H., Speed, T., and Carlin, J., (1984) Recursive causal models, Journal of the Australian Mathematical Society, 36, 30-52.

Koster, J., (1995) Markov Properties of Non-Recursive Causal Models, To appear in the Annals of Statistics, November 1995.

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

Lee, S. (1987) Model Equivalence in Covariance Structure Modelling. Ph.D. Thesis, Department of Psychology, Ohio State University.

Mason, S., (1953) Feedback theory-some properties of signal flow graphs, Proceedings of the IRE, 41.

Mason, S., (1956) Feedback theory-further properties of signal flow graphs, Proceedings of the IRE, 44.

Muthen, B. (1984) A general structural equation model, with dichotomous, ordered categorical and continuous latent variable indicators. Psychometrika 49: 115-132.

Pearl, J., (1986) Fusion, propagation, and structuring in belief networks, Artificial Intelligence 29, 241-88.

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

Pearl, J. and Verma, T. (1991) A theory of inferred causation, in Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference (Morgan Kaufmann, San Mateo, CA).

Richardson, T. (1994a). Equivalence in Non-Recursive Structural Equation Models. Proceedings:Compstat 94 , Physica Verlag.

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

Richardson T.(1995). A polynomial-Time Algorithm for Deciding Markov Equivalence of Directed Cyclic Graphical Models, Technical Report CMU-PHIL-63, Philosophy Department, Carnegie Mellon University.

Sobel, M. E. (1993) Comment in Statistical Science Vol.8 3 p.271.

Spirtes, P. Directed Cyclic graphs, Conditional Independence and Non-Recursive Linear Structural Equation Models. Philosophy, Methodology and Logic Technical Report 35, Carnegie Mellon University.

Spirtes, P. and Glymour, C., (1990) Causal Structure Among Measured Variables Preserved with Unmeasured Variables. Technical Report CMU-LCL-90-5, Laboratory for Computational Linguistics, Carnegie Mellon University.

Spirtes, P., and Glymour, C., (1991) An algorithm for fast recovery of sparse causal graphs, Social Science Computer Review, 9, 62-72.

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.,(1996) Causal Inference in the Presence of Selection Bias, Technical Report CMU-PHIL-64, Philosophy Department, Carnegie Mellon University.

Stetzl, I. (1986) Changing causal relationships without changing the fit: Some rules for generating equivalent LISREL models. Multivariate Behavior Research 21, 309-331.

Strotz, R.H. and Wold, H.O.A. (1960) Recursive versus Nonrecursive systems: An Attempt at Synthesis. Econometrica 28 pp.417-427. (Also in Causal Models in the Social Sciences, H.M. Blalock Jr. ed., Chicago: Aldine Atherton, 1971).

Verma, T. & Pearl, J., (1990). On Equivalence of Causal Models. Technical Report R-150, Cognitive Systems Lab., UCLA.

Wermuth, N. (1992) On Block-Recursive Linear Regression Equations, Revista Brasileira de Probabilidade e Estatistica 6, pp.1-56 (with discussion).

Wermuth, N., (1980) Linear recursive equations, covariance selection and path analysis, Journal of the American Statistical Association, 75, 963-972.

Wermuth, N. and Lauritzen, S., (1983) Graphical and recursive models for contingency tables, Biometrika, 72, 537-552.

Wermuth, N. and Lauritzen, S., (1990) On substantive research hypotheses, conditional independence graphs and graphical chain models, Journal of the Royal Statistical Society, Series B, 52, 21-50.

Whittaker, J., (1990) Graphical Models in Applied Multivariate Statistics (Wiley, New York).

Wold, H.O.A. Ends and Means in Econometric Model Building. Basic Considerations Reviewed. in Probability and Statistics, (The Harald Cramer volume) U. Grenander, ed., Stockholm: Almquist & Wiksell, 1959.

Wold, H.O.A., Stotz, R.H. (1960) Recursive versus Non-Recursive Systems: An attempt at Synthesis, Econometrica 28 pp.417-427.

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