A Discovery Algorithm for Directed Cyclic Graphs

Thomas Richardson[1]
Logic and Computation Programme
CMU, Pittsburgh PA 15213

1. Introduction

Directed acyclic graphs have been used fruitfully to represent causal structures (see Pearl (1988)). However, in the social sciences and elsewhere models are often used which correspond both causally and statistically to *cyclic* graphs (Spirtes (1995)). Pearl (1993) discussed predicting the effects of intervention in models of this kind, so-called linear non-recursive structural equation models. This raises the question of whether it is possible to make inferences about cyclic causal structure, from sample data. In particular do there exist general, informative, feasible and reliable procedures for inferring causal structure from conditional independence relations among variables in a sample generated by an unknown causal structure? In this paper I present a discovery algorithm that is correct in the large sample limit, given commonly (but often implicitly) made plausible assumptions, and which provides information about the existence or non-existence of causal pathways from one variable to another. The algorithm is polynomial on sparse graphs.

2. Directed Graph Models

A Directed Graph $\mathcal{G}$ consists of an ordered pair <**V**,**E**> where **V** is a set of vertices, and **E** is a set of directed edges between vertices.[2] If there are no directed cycles[3] in **E** then <**V**,**E**> is called a Directed Acyclic Graph or (DAG). A Directed Cyclic Graph (DCG) *model* (Spirtes (1995)) is an ordered pair <$\mathcal{G}$,$\mathcal{P}$> consisting of a directed graph $\mathcal{G}$ (cyclic or acyclic) and a joint probability distribution $\mathcal{P}$ over the set **V** in which certain conditional

[1] I thank P. Spirtes, C. Glymour, R. Scheines & C. Meek for helpful conversations. Research supported by NSF grant 9102169.

[2] If <A,B>∈ **E** then there is said to be an edge *from* A *to* B, represented by A→B. If <A,B>∈ **E** *or* <B,A>∈ **E**, then in either case there is said to be an edge *between* A and B.

[3] By a ‘directed cycle’ I mean a directed path $X_0 \to X_1 \ldots \to X_{n-1} \to X_0$ of n distinct vertices.

independence relations, encoded by the graph, are true.[4] Directed Acyclic Graph (DAG) models correspond to the special case in which $\mathcal{G}$ is acyclic. The independencies encoded by a given graph are determined by a graphical criterion called d-separation, as explained for the acyclic case in Pearl (1988), and extended to the cyclic case in Spirtes (1995) (See also Koster(1994)). The following definition can be applied to cyclic and acyclic cases and is equivalent to Pearl's in the latter:

Definition: d-connection / d-separation for directed graphs
For disjoint sets of vertices, **X**, **Y** and **Z**, **X** is *d-connected to* **Y** *given* **Z** if and only if for some X∈**X**, and Y∈**Y**,[5] there is an (acyclic) undirected path **U** from X to Y, such that:

(i) If there is an edge between A and B on **U**, and an edge between B and C on **U**, and B∉**Z**, then B is a collider between A and C relative to **U**, i.e. A→B←C is a subpath of **U**.

(ii) If B is a collider between A and C relative to **U**, then there is a descendant D,[6] of C, and D∈**Z**.

For disjoint sets of vertices, **X**, **Y** and **Z**, if **X** and **Y** are not d-connected given **Z** then **X** and **Y** are said to be *d-separated* given **Z**.

The constraint relating $\mathcal{G}$ and $\mathcal{P}$ in a DCG model <$\mathcal{G}$,$\mathcal{P}$> is:

The Global Directed Markov Condition
A DCG model <$\mathcal{G}$,$\mathcal{P}$>, is said to satisfy the Global Directed Markov Property if for all disjoint sets of variables **A**, **B** and **C**, if **A** is d-separated from **B** given **C** in $\mathcal{G}$ then $\mathbf{A} \perp\!\!\!\perp \mathbf{B} \mid \mathbf{C}$ in $\mathcal{P}$.[7]

This condition is important since a wide range of statistical models can be represented as DAG models satisfying the Global Directed Markov Condition, including recursive linear structural equation models with

---

[4]Since the elements of **V**, are both vertices in a graph, and random variables in a joint probability distribution the terms 'variable' and 'vertex' can be used interchangeably.
[5]Upper case Roman letters (**V**) are used to denote sets of variables, and plain face Roman letters (V) to denote single variables. |**V**| denotes the cardinality of the set **V**.
[6]'Descendant' is defined as the reflexive, transitive closure of the 'child' relation, hence every vertex is its own descendant. Similarly every vertex is its own ancestor.
[7]'$\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z}$' means that '**X** is independent of **Y** given **Z**'.

independent errors, regression models, factor analytic models, and discrete latent variable models (via extensions of the formalism). An alternative, but equivalent, definition of this condition is given by Lauritzen *et al.* (1990).

However, not all models can be represented thus as DAG models. Spirtes (1995) has shown that the conditional independencies which hold in non-recursive linear structural equation models[8] are precisely those entailed by the Global Directed Markov condition, applied to the cyclic graph naturally associated with a non-recursive structural equation model[9] with independent errors. It can be shown that in general there is no DAG encoding the conditional independencies which hold in such a model. Non-recursive structural equation models are used to model systems with feedback, and are applied in sociology, economics, biology, and psychology.

We make two assumptions connecting the probability distribution $\mathcal{P}$ and the true causal graph $\mathcal{G}$:

The Causal Markov Assumption:

A distribution generated by a causal structure represented by a directed graph $\mathcal{G}$ satisfies the Global Directed Markov condition.

For linear structural equation models this is true by definition if the error terms are independent.

The Causal Faithfulness Assumption

All conditional independence relations present in $\mathcal{P}$ are consequences of the Global Directed Markov condition applied to the true causal structure $\mathcal{G}$.

This is an assumption that any conditional independence relation true in $\mathcal{P}$ is true in virtue of causal structure rather than a particular parameterization of the model. (Further justification and discussion see Spirtes *et al*. 1993)

3 Discovery

(Cyclic or Acyclic) graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are *Markov equivalent* if any distribution which satisfies the Global Directed Markov condition with

[8]A non-recursive structural equation model is one in which the matrix of coefficients not fixed at zero is not lower triangular, for any ordering of the equations. (Bollen 1989)

[9]i.e. the directed graph in which X is a parent of Y, if and only if the coefficient of X in the structural equation for Y is not fixed at zero by the model.

respect to one graph satisfies it with respect to the other and vice versa. The class of graphs which are Markov Equivalent to $\mathcal{G}$ is denoted Equiv($\mathcal{G}$).
It can be shown to follow from the fact that the Global Directed Markov condition only places conditional independence constraints on distributions that under this definition two graphs are Markov equivalent if and only if the same d-separation relations hold in both graphs.

The Discovery Problem

Given an oracle for conditional independencies in a distribution $\mathcal{P}$, satisfying the Global Markov and Faithfulness conditions w.r.t. some directed (cyclic or acyclic) graph $\mathcal{G}$ without hidden variables, is there an efficient, reliable algorithm for making inferences about the structure of $\mathcal{G}$?

Since if $\mathcal{P}$ satisfies the Global Markov and Faithfulness conditions w.r.t. to $\mathcal{G}$, then it also satisfies them w.r.t. every graph $\mathcal{G}^*$ in Equiv($\mathcal{G}$) the conditional independencies cannot distinguish between graphs in Equiv($\mathcal{G}$). Thus a procedure solving the Discovery Problem will determine causal features common to all graphs in a given Markov equivalence class Equiv($\mathcal{G}$), given an oracle for conditional independencies in $\mathcal{P}$.
I present an feasible (on sparse graphs) algorithm which outputs a list of features common to all graphs in Equiv($\mathcal{G}$), given an oracle for conditional independence relations in a distribution $\mathcal{P}$, satisfying the Global Markov and Faithfulness conditions w.r.t. some directed (cyclic or acyclic) graph $\mathcal{G}$. The strategy adopted is to construct a graphical object, called a Partial Ancestral Graph (PAG) which represents features common to all graphs in the Markov Equivalence class (See Figure 1).

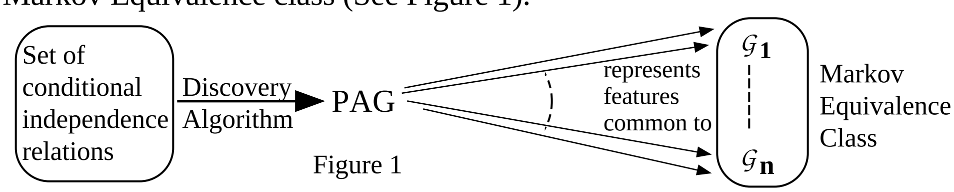


Figure 1

A PAG consists 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. In the following definition '*' is used as a meta-symbol indicating the presence of any one of {o, –, >}.

Partial Ancestral Graphs (PAGs)

$\boldsymbol{\Psi}$ is a PAG for Directed Cyclic Graph $\mathcal{G}$ with vertex set **V**, if and only if

(i) There is an edge between A and B in $\boldsymbol{\Psi}$ if and only if A and B are d-connected in $\mathcal{G}$ given all subsets $\mathbf{W} \subseteq \mathbf{V}\backslash\{A,B\}$.

(ii) If there is an edge in $\boldsymbol{\Psi}$ out of A (not necessarily into B), A–∗ B, then A is an ancestor of B in every graph in Equiv($\mathcal{G}$).

(iii) If there is an edge in $\boldsymbol{\Psi}$ into B, A∗–>B, then in every graph in Equiv($\mathcal{G}$), B is **not** an ancestor of A.

(iv) If there is an underlining A*—*B*—*C in $\boldsymbol{\Psi}$ then B is an ancestor of (at least one of) A or C in every graph in Equiv($\mathcal{G}$).

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

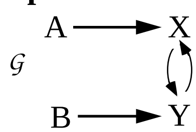
(vi) Any edge endpoint not marked in one of the above ways is left with a small circle thus: o—*.

Condition (i) differs from the other five conditions in stating necessary *and* sufficient conditions for a symbol, an edge, to appear in a PAG. The other five conditions merely give necessary conditions. For this reason there are in fact many different PAGs for a graph $\mathcal{G}$, though they all have the same edges, though not necessarily 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] Some PAGs (providing less information) represent graphs from different Markov equivalence classes. However, the PAGs output by the discovery algorithm I present, provide sufficient information so as to ensure that graphs with the features described by a particular PAG all lie in one Markov equivalence class. By the definition of a PAG, if $\boldsymbol{\Psi}$ is a PAG for $\mathcal{G}$, then $\boldsymbol{\Psi}$ is also a PAG for every $\mathcal{G}^* \in$ Equiv($\mathcal{G}$). Hence a PAG $\boldsymbol{\Psi}$ produced by the algorithm represents a unique Markov equivalence class.

---

[10]If one PAG 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 has a '–' at the end of an edge then every other PAG either has a '–' or an 'o' in that location.

Example:



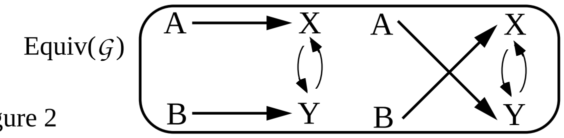
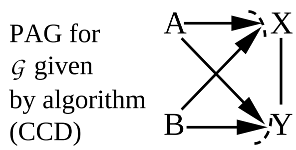


Figure 2

Consider the graph $\mathcal{G}$ in Figure 2. This graph entails that $A \perp\!\!\!\perp B$, and $A \perp\!\!\!\perp B \mid \{X,Y\}$ in any distribution $\mathcal{P}$ with respect to which it satisfies the Global Directed Markov. In this case it can be proved that Equiv($\mathcal{G}$) includes (only) the two graphs shown. Figure 3 shows the PAG given by the algorithm I give, given a conditional independence oracle for a distribution $\mathcal{P}$ satisfying the Global Directed Markov and Faithfulness w.r.t. $\mathcal{G}$.



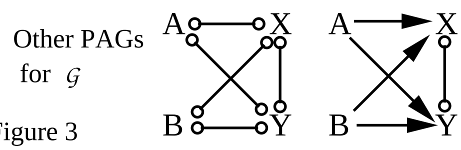


Figure 3

The PAG given by the algorithm allows us to make the following inferences (among others) about every graph in Equiv($\mathcal{G}$), and hence about $\mathcal{G}$:

(a) X is an ancestor of Y, and vice versa, hence there is a cycle.

(b) Neither X nor Y is an ancestor of A or B.

(c) Both A and B are ancestors of X and Y.

Note that not every edge in the PAG appears in every graph in Equiv($\mathcal{G}$). 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 that if there is an edge between two vertices in a PAG, then there is a graph represented by the PAG in which that edge is present. The algorithm I present does not always give the most informative PAG for a given graph $\mathcal{G}$ in that there may be features common to all graphs in the Markov equivalence class which are not captured by the PAG the algorithm outputs. In this sense the algorithm is not complete, though the algorithm is 'd-separation complete' in the sense that each PAG it outputs represents a unique Markov equivalence class.

Two vertices, X, Y in a PAG are *adjacent* if there is an edge between them, i.e. A*–*B. **Adjacent**($\mathcal{D}$,X) is the set of vertices adjacent to X in a PAG[11]

[11]Here as elsewhere '*' is a meta-symbol indicating any of the three ends -, o, >.

3.1 The Cyclic Causal Discovery (CCD) Algorithm

Input: A conditional independence oracle for a distribution $\mathcal{P}$, satisfying the Global Directed Markov and Faithfulness conditions w.r.t. a (cyclic or acyclic) graph $\mathcal{G}$ with vertex set **V.**
(In practice of course statistical tests of conditional independence in sample data take the place of the conditional independence oracle.)

Output: A PAG for the Markov equivalence class Equiv($\mathcal{G}$).

¶A Form a PAG $\mathcal{E}$ with an edge Xo–oY between every pair of vertices.
n =0
repeat
    repeat
        Select an ordered pair of variables X and Y that are adjacent in $\mathcal{E}$ s.t. |**Adjacent**($\mathcal{E}$,X)\{Y}| ≥ *n*, and a set **S**⊆ **Adjacent**($\mathcal{E}$,X)\{Y} s.t. |**S**| =*n*.. If $X \perp\!\!\!\perp Y \mid \mathbf{S}$, delete edge Xo-oY from $\mathcal{E}$ and record **S** in **Sepset**(X,Y) and **Sepset**(Y,X).[12]
    until all pairs of adjacent variables X,Y s.t. |**Adjacent**($\mathcal{E}$,X)\{Y}| ≥ *n* and all sets **S**⊆ |**Adjacent**($\mathcal{E}$,X)\{Y}| s.t. |**S**| =*n* have been tested.
    n =*n* +1;
until for all ordered pairs of adjacent vertices X,Y, |**Adjacent**($\mathcal{E}$,X)\{Y}|<*n*

¶B. For each triple of vertices A,B,C s.t. the pair A,B and the pair B,C are each adjacent in $\mathcal{E}$ but the pair A, C are not adjacent in $\mathcal{E}$, orient A∗—∗B∗—∗C as A—>B<—C if and only if B ∉ **Sepset**<A,C>; orient A∗—∗B∗—∗C as $A{*}\!-\!\underline{{*}B{*}}\!-\!{*}C$ if and only if B ∈ **Sepset**<A,C>.

¶C. For each triple of vertices <A,X,Y> in $\mathcal{E}$ such that (a) A is not adjacent to X or Y, (b) X and Y are adjacent, (c) X ∉ **Sepset**<A,Y> then orient X ∗–∗Y as X<—Y if $A \not\perp\!\!\!\perp X \mid$ **Sepset**<A,Y>.

¶D. For each vertex V in $\mathcal{E}$ form the following set: X∈ **Local**($\mathcal{E}$,V) ⇔ X is adjacent to V in $\mathcal{E}$, or there is a vertex Y s.t. X—>Y<—V in $\mathcal{E}$.[13]

---

[12] **Adjacent**($\mathcal{E}$,X) is updated when the graph $\mathcal{E}$ changes during ¶A. So Y∉ **Adjacent**($\mathcal{E}$,X), X∉ **Adjacent**($\mathcal{E}$,Y), after the edge Xo–oY is removed.
[13] Local($\mathcal{E}$,A) is not recalculated as the algorithm progresses.

$m = 0$
repeat
    repeat
        select an ordered triple <A,B,C> such that A—>B<—C, A and C are not adjacent, and $|\mathbf{Local}(\mathcal{E},A)\setminus\{B,C\}| \geq m$, and a set $\mathbf{T} \subseteq \mathbf{Local}(\mathcal{E},A)\setminus\{B,C\}$, $|\mathbf{T}|=m$, and if $A \perp\!\!\!\perp C \mid \mathbf{T}\cup\{B\}$ then orient A—>B<—C as A—$\underline{>B<}$—C, and record $\mathbf{T} \cup \{B\}$ in **Supset**<A,B,C>.
    until for all triples such that A—>B<—C, (not A—$\underline{>B<}$—C), A and C are not adjacent, $|\mathbf{Local}(\mathcal{E},A)\setminus\{B\}| \geq m$, every subset $\mathbf{T} \subseteq \mathbf{Local}(\mathcal{E},A)$, $|\mathbf{T}| = m$ has been considered.
    $m = m + 1$;
until for all ordered triples <A,B,C> s.t. A—>B<—C, A and C not adjacent, are such that $|\mathbf{Local}(\mathcal{E},A)\setminus\{B\}| < m$.

¶E. If there is a quadruple <A,B,C,D> of distinct vertices in $\mathcal{E}$ such that (i) A—$\underline{>B<}$—C, (ii) A—>D<—C or A—$\underline{>D<}$—C, (iii) B and D are adjacent, then orient B∗–∗D as B—>D in $\mathcal{E}$ if $D \notin$ **Supset**<A,B,C>
else orient B∗–∗D as B∗—D.

¶F. For each quadruple <A,B,C,D> in $\mathcal{E}$ of distinct vertices s.t. D is not adjacent to both A and C, and A—$\underline{>B<}$—C, if $A \not\!\perp\!\!\!\perp D \mid$ **Supset**<A,B,C> $\cup \{D\}$, then orient B∗—∗D as B—>D in $\mathcal{E}$

3.2 Soundness and Completeness

Theorem 1 (Soundness)
Given as input a conditional independence oracle for a distribution $\mathcal{P}$, satisfying the Global Directed Markov and Faithfulness assumptions w.r.t. a (cyclic or acyclic) graph $\mathcal{G}$, the CCD algorithm outputs a PAG Ψ for $\mathcal{G}$.
The proof of Theorem 1 is given in §4.

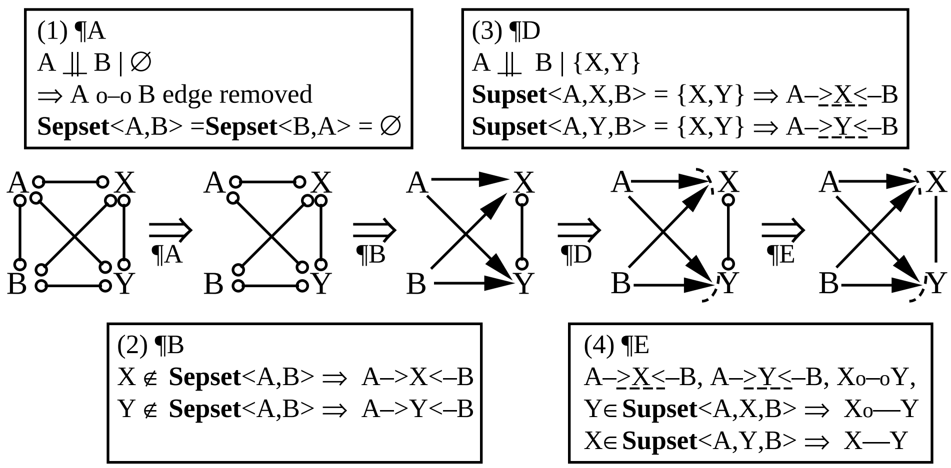
Theorem 2 (d-separation Completeness)
If the CCD algorithm, when given as input conditional independence oracles for distributions $\mathcal{P}_1$, $\mathcal{P}_2$ satisfying the Global Directed Markov and Faithfulness w.r.t. graphs $\mathcal{G}_1$, $\mathcal{G}_2$, respectively produces as output PAGs

Ψ_1, Ψ_2 respectively, then $\Psi_1 \equiv \Psi_2$ if and only if $\mathcal{G}_1$ and $\mathcal{G}_2$ are Markov equivalent.

The proof, (in Richardson(1996)) exploits the characterization of Markov equivalence in Richardson (1994) to establish that if $\mathcal{G}_1$ and $\mathcal{G}_2$ are not Markov equivalent then the algorithm produces different PAGs. (It follows directly from Theorem 1 that if $\mathcal{G}_1$ and $\mathcal{G}_2$ are equivalent then $\Psi_1 \equiv \Psi_2$.)

3.3 Trace of CCD Algorithm

If given a conditional independence oracle for $\mathcal{G}$ in figure 2 the algorithm runs as follows: (Steps ¶C and ¶F do not perform any orientations here.)



3.4 Complexity of the CCD Algorithm

Let $r = \mathrm{MaxDegree}(\mathcal{G}) = \mathop{\mathrm{Max}}_{Y\in\mathbf{V}} \left|\{X \mid Y \leftarrow X, \text{ or } X \leftarrow Y \text{ in } \mathcal{G}\}\right|$,

$k = \mathrm{MaxAdj}(\mathcal{G}) = \mathop{\mathrm{Max}}_{Y\in\mathbf{V}} \left|\{X \mid X \text{ is adjacent to } Y \text{ in any PAG for } \mathcal{G}\}\right|$,[14]

and n = no. of vertices in $\mathcal{G}$. It then follows that in searching (possibly unsuccessfully) for **Sepset**<X,Y> for every pair of distinct variables X,Y,

$$\text{Total no. of tests of conditional independence in ¶A} \leq 2 \cdot \binom{n}{2} \sum_{i=0}^{k} \binom{n-2}{i} \leq \frac{(k+1)n^2(n-2)^{k+1}}{k!}.$$

Since $\mathrm{MaxAdj}(\mathcal{G}) \leq (\mathrm{MaxDegree}(\mathcal{G}))^2$, this step is $O(n^{r^2+3})$. (Even as a

[14]Note k≠r since there may be an edge between two variables X∗–∗Y in a PAG for $\mathcal{G}$, even if there is no edge between X and Y in $\mathcal{G}$

worst case complexity bound this is loose.) ¶C performs at most one conditional independence test for each triple satisfying the conditions given, so this step is O(n^3). In searching (possibly unsuccessfully) for sets **Supset**<X,Y,Z> for triples of distinct variables <X,Y,Z>

$$\text{Total no. of tests of conditional independence in ¶D} \leq 3 \cdot \binom{n}{3} \sum_{i=0}^{m} \binom{n-3}{i} \leq \frac{(m+1)n^3(n-3)^{m+1}}{m!}$$

where m = $\max_{Y \in V}$ |{X | Local($\mathcal{E}$,X)} in ¶D. Since m ≤ (MaxDegree($\mathcal{G}$))^2, it follows that ¶D is O(n^{r^2+4}). ¶F performs at most one test for each quadruple satisfying the conditions, so this step is O(n^4). (¶B and ¶E do not perform any tests). Hence the complexity of the algorithm is polynomial in the number of vertices for graphs of fixed degree (r); it is of course exponential in r. Although there are exponentially many conditional independence facts to check, the algorithm exploits entailment relations between to obviate checking most of them when the graph is sparse.

4 Proof of Theorem 1 (Soundness)

The proof proceeds by showing that each section of the algorithm makes correct inferences from conditional independencies in $\mathcal{P}$, to the structure of any graph satisfying the Global Directed Markov and Faithfulness conditions w.r.t. to $\mathcal{P}$. If $\mathcal{P}$ satisfies the Global Directed Markov and Faithfulness conditions w.r.t. a graph $\mathcal{G}$, then **X** ⫫ **Y** | **Z**, if and only if **X** is d-separated from **Y** by **Z** in $\mathcal{G}$. Hence the oracle for conditional independencies can be thought of as an oracle for testing d-separation relations in $\mathcal{G}$.

<u>Sections ¶A</u>

Lemma 1: Given a PAG Ψ for graph $\mathcal{G}$, if in $\mathcal{G}$ either (i) X→Y or (ii) Y←X or (iii) there is some vertex Z, s.t. X→Z←Y, and Z is an ancestor of X or Y (or both) *then* X and Y are adjacent in **Ψ**, i.e. X and Y are d-connected given any subset **S**⊆**V**\{X,Y} of the other vertices in $\mathcal{G}$.

Proof: If (i) holds then the path X→Y d-connects X and Y given any subset **S**⊆**V**\{X,Y}, hence X and Y are adjacent in any PAG **Ψ** for graph $\mathcal{G}$. The case in which (ii) holds is equally trivial: X←Y is a d-connecting path given any set **S**⊆**V**\{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 **S** be an arbitrary subset of the other variables ($\mathbf{S} \subseteq \mathbf{V} \backslash \{X,Y\}$). If $\mathbf{S} \cap \{Z, A_1 \ldots A_n\} \neq \emptyset$ then $X \rightarrow Z \leftarrow Y$ is a d-connecting path given **S**. If $\mathbf{S} \cap \{Z, A_1 \ldots A_n\} = \emptyset$ then $X \rightarrow Z \rightarrow A_1 \rightarrow \ldots A_n \rightarrow Y$ is d-connecting given **S**. ∴

Lemma 2 In a graph $\mathcal{G}$, with vertices **V**, if all of the following hold:[15]

(i) X is not a parent of Y in $\mathcal{G}$

(ii) Y is not a parent of X in $\mathcal{G}$

(iii) there is no vertex Z s.t. Z is a common child of X and Y, and Z is an ancestor of X or Y

then for any set **Q**, X and Y are d-separated given **T** defined as follows:

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

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

Proof: Every vertex in **S** is an ancestor of X or Y or **Q**. Every vertex in **T** is either a parent of X, a vertex in **S**, or a parent of a vertex in **S**, hence every vertex in **T** is an ancestor of X or Y or **Q**.

Claim: If (i),(ii) and (iii) hold then X and Y are d-separated by **T**.

Suppose there is an undirected path **P** d-connecting X and Y. Let W be the first vertex on **P**. ((i) and (ii) imply $W \neq Y$.) There are two cases:

Case 1 The path **P** goes $X \leftarrow W \ldots Y$.

Subcase A: W is not a descendant of a common child of X and Y.

In this case $W \in \mathbf{T}$ (Since W is a parent of X). Thus since W is a non-collider on **P**, **P** is not d-connecting given **T**. Contradiction.

Subcase B: W is a descendant of a common child of X and Y.

In this case since X is a child of W, then X is a descendant of some common child Z of X and Y. But then Z is an ancestor of X, contradicting (iii).

Case 2 The path **P** goes $X \rightarrow W \ldots Y$.

[15] i.e. None of the conditions in the antecedent of Lemma 1 hold.

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 **P** goes $X \rightarrow W \leftarrow V \ldots Y$.

If **P** 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**. So W is an ancestor of X, Y or **Q**, hence $W \in$ **S**. Moreover, since W is (by hypothesis) not a descendant of a common child, $V \neq Y$, and V is not a descendant of a common child of X and Y. Now V is a parent of W, $W \in$ **S**, $X \neq V \neq Y$ so $V \in$ **T**. Hence **P** fails to d-connect given **T**.

Sub-subcase b: The path **P** goes $X \rightarrow W \rightarrow V \ldots Y$.

If **P** 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**, so $W \in$ **S**. Since W is (by hypothesis) not a descendant of a common child of X and Y, and $X \neq W \neq Y$, $W \in$ **T**. Since in this case W occurs as a non-collider on **P**, **P** 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) $\cap$ **T** = $\emptyset$, 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 is in **T**, if W occurs on d-connecting path **P** then W is a non-collider on **P**. Suppose that there is a collider on **P**, take the first collider on the path after W, let us say <A,B,C>, so that **P** now takes the form: $X \rightarrow W \rightarrow \ldots \rightarrow \ldots \rightarrow A \rightarrow B \leftarrow C \ldots Y$. Since <A,B,C> is the first collider after W, it follows that B is a descendant of W. But if **P** is d-connecting then there is some descendant D of B, s.t. $D \in$ **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** = $\emptyset$. Hence every vertex on **P** is a non-collider.

As there are no colliders on **P** 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 2.$\therefore$

Corollary A

Given a graph $\mathcal{G}$, and PAG Ψ for $\mathcal{G}$, X and Y are adjacent in Ψ if and only if one of the following holds in $\mathcal{G}$: (a) X is a parent of Y, (b) Y is a parent of X (c) 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 1, 'Only if' by Lemma 2 with $\mathbf{Q}=\emptyset$ ∴

X and Y are said to be adjacent *in* $\mathcal{G}$ if at least one of (a), (b), (c) holds for X,Y in $\mathcal{G}$. By Corollary A X and Y are adjacent in $\mathcal{G}$ if and only if X and Y are adjacent in every PAG for $\mathcal{G}$. Therefore I refer to a pair of variables as adjacent without specifying whether in a graph $\mathcal{G}$ or a PAG for $\mathcal{G}$.

Corollary B

In a graph $\mathcal{G}$, 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. Further, either **T** is a subset of the vertices adjacent to X or X is an ancestor of Y.

Proof: Let **S**, **T** be the sets defined in Lemma 2 with $\mathbf{Q}=\emptyset$. By Lemma 2 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**, hence $\mathbf{T} \subseteq$ Ancestors{X,Y}. Moreover, every vertex in **T** is either a parent of X, a child of X, or a parent V of some vertex C in **S**, s.t. $X \rightarrow C$. Any vertex in the first two categories is clearly adjacent to X. Any vertex in the last category is adjacent to X if C is an ancestor of X. Since C is in **S**, C is an ancestor of X or Y.

If X is not an ancestor of Y then no child C of X is an ancestor of Y, so C is an ancestor of X; hence any parent V of C is also adjacent to X.∴

Lemma 3

If A and B are not adjacent, then either A and B are d-separated given a set $\mathbf{T_A}$ of vertices adjacent to A or by a set $\mathbf{T_B}$ of vertices adjacent to B.

Proof: By Corollary B to Lemma 2, if A and B are not adjacent then A and B are d-separated given $\mathbf{T_A}$ where: $\mathbf{S_A}$ = Children(A) $\cap$ Ancestors({A,B})

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

Case 1: A is not an ancestor of B

From the Corollary B to Lemma 2, since A is not an ancestor of B, $\mathbf{T_A} \subseteq \{X \mid X \text{ adjacent to } A\}$.

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 $\mathbf{S_A}$, s.t. A→C. Clearly vertices in the first two categories are adjacent to A; as before, vertices in the last category are 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 adjacent to A. ∴

Let $\mathcal{G}$ be any graph satisfying the Global Markov and Faithfulness conditions w.r.t. the distribution $\mathcal{P}$ given as input. To find a set which d-separates some pair of variables A and B in $\mathcal{G}$ the algorithm tests subsets of the vertices which are adjacent to A in $\mathcal{E}$, and subsets of vertices which are adjacent to B in $\mathcal{E}$ to see if they d-separate A and B. Since the vertices which are adjacent to A and B in $\mathcal{G}$ are at all times a subset of the vertices adjacent to A and B in $\mathcal{E}$ [16] Lemma 3 implies that step **¶A** is guaranteed to find a set which d-separates A and B, if any set d-separates A and B in $\mathcal{G}$.

Section ¶B

Lemma 4 Suppose that Y is not an ancestor of X or Z or a set **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 then **S** d-connects X and Z.

Proof Let $\mathbf{T}^* = \text{Ancestors}(\{X,Z\} \cup \mathbf{R}) \cap \mathbf{S}$. 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

[16] This is because if a pair of vertices X,Y are adjacent in $\mathcal{G}$ then no set is found which d-separates them hence the edge between X and Y in $\mathcal{E}$ is never deleted.

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 (every collider active given $\mathbf{T}^*$ will remain active given **S**). Hence **P** still d-connects X and Z given **S**.∴

S is said to be a *minimal d-separating* set for X and Y if X and Y are d-separated given **S**, and are d-connected given any proper subset of **S**.

Corollary: If **S** is a minimal d-separating set for X and Y, then any vertex in **S** is an ancestor of X or Y.
Proof: Follows by contraposition from Lemma 4 with $\mathbf{R} = \emptyset$ ∴
This shows that the unshielded non-collider orientation rule in **¶B** is correct: If A and B, and B and C are adjacent, but **Sepset**(A,C) contains B, then by the nature of the search procedure A and C are not d-separated given any subset of **Sepset**(A,C) hence it follows that B is an ancestor of A or C, hence $A{*}\!-\!{*}B{*}\!-\!{*}C$ should be oriented as $A{*}\!-\!\underline{{*}B{*}}\!-\!{*}C$.

I will make frequent use of the following Lemma, which I state here without proof (It is a simple extension to the cyclic case of Lemma 3.3.1 in Spirtes *et al.*, 1993, p.376) The Lemma gives conditions under which a set of d-connecting paths may be joined to form a single d-connecting path.

Lemma 3.3.1+ (Richardson 1994, p.82)
In a directed (cyclic or acyclic) graph $\mathcal{G}$ over a set of vertices **V**, *IF* **R** is a sequence of distinct vertices in **V** from A to B, $\mathbf{R} \equiv \langle A\equiv X_0,\ldots X_{n+1}\equiv B\rangle$, $\mathbf{S} \subseteq \mathbf{V}\backslash\{A,B\}$ and $\mathcal{T}$ 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 $\mathbf{S}\backslash\{X_i,X_{i+1}\}$.

(ii) if some vertex X_k in **R**, is in **S**, then the paths in $\mathcal{T}$, that contain X_k as an endpoint collide at 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 **S**.

THEN there is a path **U** in $\mathcal{G}$ that d-connects $A \equiv X_0$ and $B \equiv X_{n+1}$ given **S**.

Lemma 5: If A and B are adjacent, B and C are adjacent, and B is an ancestor of A or C then A and C are d-connected given any set **S**\{A,C}, s.t. B ∉ **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:

Case 1: Some (proper) descendant of B is in **S**.

It follows from Lemma 1 and the adjacency of A and B, that given any set **S**, conditional on **S**\{A,B}, there is a d-connecting path from A to B, and likewise a d-connecting path from B to C, conditional on **S**\{B,C}. Since some descendant of B is in **S**\{A,C}, but B ∉ **S**\{A,C}, it follows by Lemma 3.3.1+ that A and C are d-connected, since it does not matter whether or not the path from A to B and from B to C collide at B.

Case 2: No descendant of B is in **S.**

Again by Lemma 1 there is a path d-connecting from A to B. Since no descendant of B is in **S** the directed path B→…→C is also d-connecting. Since B∉**S**, Lemma 3.3.1+ implies A and C are d-connected by **S**.∴

It follows by contraposition from Lemma 5 that if A and B are adjacent, B and C are 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 ¶B correctly orients A∗—∗B∗—∗C as A—>B<—C.

<u>Section ¶C</u>

Lemma 6: 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**, and some subset **T**⊆**S** is a minimal d-separating set for A and X.

Proof: Let X be an ancestor of Y. Let **S** be any set s.t. there is a path **Q**

which d-connects X and Y given **S**, X$\notin$**S**, and A and Y are d-separated by **S**. Suppose, for a contradiction, that there is a path **P** d-connecting A and X given **S**. There are now two cases:

Case 1: Some descendant of X is in **S**. Since X$\notin$**S**, and some descendant of X is in **S**, Lemma 3.3.1+ implies that the d-connecting paths **P** and **Q**, can be joined to form a path d-connecting A to Y given **S**, a contradiction.

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 **Q***, X$\rightarrow\ldots\rightarrow$Y, given **S.** By Lemma 3.3.1+ **P** and **Q*** can be joined to form a path d-connecting A and Y given **S**, a contradiction.

Thus under the conditions in the antecedent, **S** is a d-separating set for A and X. Let **T** be the smallest subset of **S** which d-separates A and X, **T** is a minimal d-separating set for A and X.$\therefore$

Lemma 7: Let A, X and Y be three vertices in a graph, s.t. X and Y are adjacent. If there is a set **S** s.t. X$\notin$**S**, A and Y are d-separated given **S**, while A and X are d-connected given **S**, then X is not an ancestor of Y.

Proof: If X and Y are adjacent then X and Y are d-connected by every set **S**, s.t. X,Y$\notin$**S**. If there is a set **S** which d-separates A and Y but does not contain any subset which d-separates A and X, where X is adjacent to Y, and X $\notin$**S**, then **S** does not contain a (minimal) d-separating set for A and X, hence, by Lemma 6 X is not an ancestor of Y.$\therefore$

¶C simply applies Lemma 7: If A and X are d-connected given **Sepset**<A,Y>, and X $\notin$ **Sepset**<A,Y>, then since **Sepset**<A,Y> d-separates A and Y, by Lemma 7, **¶C** correctly orients X*–*Y as X<—Y.

<u>Section ¶D</u>

Lemma 8: If in a graph $\mathcal{G}$, Y is a descendant of a common child of X and Z then X and Z are d-connected by any set **S** s.t. Y$\in$**S**, X,Z$\notin$**S**.

Proof: If Y is a descendant of a common child C of X and Z then the path X$\rightarrow$C$\leftarrow$Z d-connects X and Z given any set **S**, s.t. Y$\in$**S**, X,Z$\notin$**S**.

Corollary: If in a graph $\mathcal{G}$, X and Y are adjacent, Y and Z are adjacent, but X and Z are not 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.

Lemma 9: If in graph $\mathcal{G}$, 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) ∩ Ancestors({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 adjacent then Y∈ **T**.

Proof: Lemma 2, with **Q**={Y} implies that X and Z are d-separated by **T**. If Y is a child of X, then since Y is an ancestor of Y, Y∈ **S**. Since Y is not a descendant of a common child of X and Z, Y∈ **T**. If Y is a parent of X then since Y is not a descendant of a common child of X and Z, Y∈ **T**. If X and Y have a common child C that is an ancestor of X or Y, then C∈ **S**; since Y is a parent of C, and Y is not a descendant of a common child of X and Z then Y∈ **T**. So if X and Y are adjacent then Y∈ **T.** ∴

¶D considers each triple A—>B<—C in $\mathcal{E}$, A and C are not adjacent, in turn, and tries to find a set **R**⊆ **Local**($\mathcal{E}$,A)\{B,C} s.t. A and C are d-separated by **R** ∪ {B}. If A and C are d-separated by a set containing B, then Lemma 8 implies that B is not a descendant of a common child of A and C. It then follows from Lemma 9 that the set **T** in Lemma 9 is s.t. B∈ **T**, A and C are d-separated by **T**, and **T** ⊆ **Local**($\mathcal{E}$,X). So **¶D** will find a set which d-separates A and C, but contains B, if such a set exists.

Section ¶E

Lemma 10: If in a graph $\mathcal{G}$, A and D are adjacent, D and C are adjacent, A and C are not adjacent, D is an ancestor of B then any set **S** such that B∈ **S**, and A and C are d-separated by **S**, also contains D.

Proof Suppose for a contradiction that A and C were d-separated by a set **S**, s.t. B∈ **S**, D∉ **S**. Since A is adjacent to D, (D,A∉ **S**), by Lemma 1 there is an undirected path **P** d-connecting A and D given **S**. Likewise there is a path **Q** d-connecting D and C given **S**. Since D is an ancestor of B, B∈ **S**, but D∉ **S**, Lemma 3.3.1+ implies that **P** and **Q** can be joined to form a new path d-connecting A and C given **S**. This is a contradiction.∴

By contraposition Lemma 10 justifies **¶E** in the case where A—>B<—C, A—>D<—C, D∉ **Supset**<A,B,C>, and so B∗–∗D is oriented as B—>D.

In the case in which A—>B<—C, A—>D<—C, and D∈ **Supset**<A,B,C> Lemma 4, and the nature of the search for **Supset**<A,B,C> [17] imply that D is an ancestor of {A,B,C}. But since there are arrowheads at D on the edges A–>D<–C, D is not an ancestor of A or C, so D is an ancestor of B. So ¶E correctly orients B∗–∗D as B∗—D.

In the case where A—>B<—C, A→D←C in *E*, (A and C not adjacent and no dotted line A—>D<—C), Lemma 8 implies that, since A and C are d-connected by any set **S** s.t. D∈ **S**, (A,C∉ **S**), D is a descendant of a common child of A and C. Since A and C are d-separated by **Supset**<A,B,C>, and B∈ **Supset**<A,B,C>, then B is not a descendant of D. So ¶E correctly orients B*–*D as B<—D.

<u>Section ¶F</u>

Lemma 11: If X and Z are d-separated by some set **R**, then for all sets **Q** ⊆ **Ancestors**(**R** ∪ {X,Z})\{X,Z}, X and Z are d-separated by **R** ∪ **Q**.

Proof: Suppose, for a contradiction that there is a path **P** d-connecting X and Z given **R** ∪ **Q**. It follows that every vertex on **P** is an ancestor of either X, Z, or **R** ∪ **Q**. Since **Q** ⊆ **Ancestors**(**R** ∪ {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 exists). The paths X←…←A, and B→…→Z are d-connecting given **R**, since no vertex on the paths is in **R**. 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** ∪ **Q**. Lemma 3.3.1+ implies that these three paths can be joined to form a path d-connecting X and Z given **R**. This is a contradiction.∴

---

[17]Namely, that ¶D looks for the smallest set containing B, which d-separates A and C.

In ¶F, since A and C are d-separated by **Supset**<A,B,C> $\supseteq$ {B}, by Lemma 11, if A and C are d-connected given **Supset**<A,B,C> $\cup$ {D} then D is not an ancestor of B. Further, since B and D are adjacent, B is an ancestor of D. So ¶F correctly orients B∗—∗D as B—>D in $\mathcal{E}$.
This completes the proof of the correctness of the algorithm. ∴

§4.2 Proof of Theorem 2: d-separation Completeness

All that is required is to show that if two graphs $\mathcal{G}_1$, and $\mathcal{G}_2$ when used as a d-separation oracle for the CCD algorithm, result in the same PAG being produced as output, then $\mathcal{G}_1$ and $\mathcal{G}_2$ are equivalent. We shall do this by proving that if $\mathcal{G}_1$ and $\mathcal{G}_2$ when used as input to the CCD algorithm produce the same PAG, then $\mathcal{G}_1$, and $\mathcal{G}_2$ satisfy five conditions of the Cyclic Equivalence Theorem CET(I)-(V) (given below) with respect to one another. I have already shown in Richardson(1994b) that two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are equivalent to one another if and only if they satisfy these 5 conditions.

Before stating the Equivalence Theorem we require a number of extra definitions:

Definition: Unshielded Conductor and Unshielded Non-Conductor

In a cyclic graph $\mathcal{G}$, we say triple of vertices <A,B,C> forms an *unshielded conductor* if:

(i) A and B are adjacent, B and C are adjacent, A and C are not 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.*

Definition: Unshielded Perfect and Imperfect Non-Conductors

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

(i) A and B are adjacent, B and C are adjacent, but A and C are not adjacent.

(ii) B is not an ancestor of A or C.

(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.*

Definition: Itinerary

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

Definition: Mutually Exclusive Unshielded Conductors with respect to 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.

(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* $<X_0,\ldots X_{n+1}>$.

Definition: Uncovered itinerary

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 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 adjacent to other vertices on the itinerary, are those that occur consecutively on the itinerary.

Cyclic Equivalence Theorem: (Richardson 1994) Graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are d-separation equivalent if and only the following five conditions hold:

CET(I) $\mathcal{G}_1$ and $\mathcal{G}_2$ have the same p-adjacencies,

CET(II) $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 ≡<A,B,C,…X,Y,Z> in $\mathcal{G}_1$ if and only if <A,B,C> and <X,Y,Z> are m.e. conductors on some uncovered itinerary Q ≡<A,B,C,…X,Y,Z> in $\mathcal{G}_2$,

CET(IV) If <A,X,B> and <A,Y,B> are unshielded imperfect non-conductors (in $\mathcal{G}_1$ and $\mathcal{G}_2$), then X is an ancestor of Y in $\mathcal{G}_1$ iff X is an ancestor of Y in $\mathcal{G}_2$,

CET(V) If <A,B,C> and <X,Y,Z> are mutually exclusive conductors on some uncovered itinerary **P** ≡<A,B,C,…X,Y,Z> and <A,M,Z> is an unshielded imperfect non-conductor (in $\mathcal{G}_1$ and $\mathcal{G}_2$), then M is a descendant of B in $\mathcal{G}_1$ iff M is a descendant of B in $\mathcal{G}_2$.

Lemma 12: Given a sequence of vertices $<X_0,\ldots X_{n+1}>$ in a directed graph $\mathcal{G}$ having the property that $\forall k,\ 0 \le k \le n$, X_k is an ancestor of X_{k+1}, and X_k is 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 adjacent in the graph if and only if $k = j+1$. i.e. the only Y_k's which are 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 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 adjacent. Moreover, if $Y_j \equiv X_\alpha$[18] and $Y_k \equiv X_\beta$ are 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 13: Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be two graphs satisfying CET(I)–(III) Suppose there is a directed path $D_1 \to \ldots D_n$, in $\mathcal{G}_1$. Let D_0 be a vertex distinct from $D_1,\ldots,D_n$, s.t. D_0 is adjacent to D_1 in $\mathcal{G}_1$ and $\mathcal{G}_2$, D_0 is not adjacent to $D_2,\ldots D_n$ in $\mathcal{G}_1$ or $\mathcal{G}_2$ and D_0 is not a descendant of D_1 in $\mathcal{G}_1$ or $\mathcal{G}_2$. It then follows that D_1 is an ancestor of D_n in $\mathcal{G}_2$.

Proof. By induction on n.

Base Case: n = 2. Since, by hypothesis, D_0 is not adjacent to D_2, it follows that $\langle D_0, D_1, D_2\rangle$ forms an unshielded conductor in $\mathcal{G}_1$ (since D_1 is an ancestor of D_2). Hence this triple of vertices also forms an unshielded conductor in $\mathcal{G}_2$, by CET(IIa). Hence D_1 is an ancestor of D_0 or D_2 in $\mathcal{G}_2$. Since, by hypothesis D_1 is not an ancestor of D_0 in $\mathcal{G}_2$, it follows that D_1 is an ancestor of D_2 in $\mathcal{G}_2$.

Induction Case: Suppose that the hypothesis is true for paths of length n. It follows from Lemma 12 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 adjacent vertices are those that occur consecutively, and in $\mathcal{G}_1$ each vertex is an ancestor of the next vertex in the sequence. Moreover, since, by hypothesis, D_0 is not adjacent to $D_2,\ldots D_n$, it follows that $D_{\alpha(1)} \equiv D_1$. Since $\mathcal{G}_1$ and $\mathcal{G}_2$ satisfy CET(I), they have the same adjacencies, hence in $\mathcal{G}_2$ the only vertices that are adjacent are those that

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

occur consecutively in the sequence. Suppose, for a contradiction that $D_{\alpha(r-1)}$ is not an ancestor of $D_{\alpha(r)}$ in $\mathcal{G}_2$. Let s be the smallest j such that $D_{\alpha(j)}$ is not an ancestor of $D_{\alpha(j-1)}$ in $\mathcal{G}_2$. (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 $\mathcal{G}_2$. But these two triples are not mutually exclusive in $\mathcal{G}_1$ since $D_{\alpha(r-1)}$ is an ancestor of $D_{\alpha(r)}$ in $\mathcal{G}_1$; hence $\mathcal{G}_1$ and $\mathcal{G}_2$ fail to satisfy CET(III), which is a contradiction.

It follows that $D_{\alpha(r-1)}$ is an ancestor of $D_{\alpha(r)}$ in $\mathcal{G}_2$. It then follows from the induction hypothesis that D_1 is an ancestor of $D_{\alpha(r)} \equiv D_n$. $\therefore$

Theorem 2: (d-separation Completeness) If the CCD algorithm, when given as input d-separation oracles for the graphs $\mathcal{G}_1$, $\mathcal{G}_2$ produces as output PAGs Ψ_1, Ψ_2 respectively, then Ψ_1 is identical to Ψ_2 if and only if $\mathcal{G}_1$ and $\mathcal{G}_2$ are d-separation equivalent, i.e. $\mathcal{G}_2 \in$ **Equiv**($\mathcal{G}_1$) and vice versa.

Proof. We will show that if two graphs, $\mathcal{G}_1$ and $\mathcal{G}_2$ are *not* d-separation equivalent, then the PAGs output by the CCD algorithm, given d-separation oracles for $\mathcal{G}_1$ and $\mathcal{G}_2$ as input, would differ in some respect.

It follows from the Cyclic Equivalence Theorem that if $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 $\mathcal{G}_1$ and $\mathcal{G}_2$ as input.

Case 1: $\mathcal{G}_1$ and $\mathcal{G}_2$ fail to satisfy CET(I). In this case the two graphs have different adjacencies. Let us suppose without loss of generality that there is some pair of variables, X and Y which are adjacent in $\mathcal{G}_1$ and not adjacent in $\mathcal{G}_2$. Since X and Y are adjacent in $\mathcal{G}_1$, X and Y are d-connected conditional upon any subset of the other vertices. Hence there is an edge between X and

Y in $\mathbf{\Psi}_1$.

Since X and Y are not adjacent in $\mathcal{G}_2$, there is some subset **S**, (X,Y$\notin$ **S**) such that X and Y are d-separated in $\mathcal{G}_2$ 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 adjacent to X, or **T** is a subset of the vertices adjacent to Y. It follows that in step ¶A of the CCD algorithm the edge between X and Y in $\mathbf{\Psi}_2$ would be removed. Since edges are not added back in at any later stage of the algorithm, there is no edge in $\mathbf{\Psi}_2$ between X and Y. Hence $\mathbf{\Psi}_1$ and $\mathbf{\Psi}_2$ are different.

Case 2: $\mathcal{G}_1$ and $\mathcal{G}_2$ fail to satisfy CET(IIa). We assume that $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 $\mathcal{G}_1$, Y is an ancestor of X or Z, while Y is not an ancestor of either X or Z in $\mathcal{G}_2$.

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$\in$ **Sepset**(X,Z) in $\mathcal{G}_1$. It then follows from the correctness of the algorithm that in $\mathbf{\Psi}_1$, either X—>Y—*Z, X*—Y<—Z, or X*–$\underline{*Y*}$–*Z.

If Y is not an ancestor of X or Z in $\mathcal{G}_2$, then Y is not in any minimal d-separating set for X and Z. In particular Y$\notin$ **Sepset**(X,Z) for $\mathcal{G}_2$. Again it follows from the correctness of the algorithm that <X,Y,Z> is oriented as X*–>Y<–*Z or X*–$\underset{\cdots}{>Y<}$–*Z in $\mathbf{\Psi}_2$. Thus $\mathbf{\Psi}_1$ and $\mathbf{\Psi}_2$ are different.

Case 3: $\mathcal{G}_1$ and $\mathcal{G}_2$ fail to satisfy CET(IIb). We assume that $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 $\mathcal{G}_1$ and $\mathcal{G}_2$, 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 $\mathcal{G}_1$, while in $\mathcal{G}_2$ it is not.

It follows from Lemma 5 that in $\mathcal{G}_1$, 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 **Supset**<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 9 that there is some subset **T** of **Local**(Ψ,X), such that X and Z are d-separated given **T** $\cup$ {Y}. 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: $\mathcal{G}_1$ and $\mathcal{G}_2$ fail to satisfy CET(III). We assume that $\mathcal{G}_1$ and $\mathcal{G}_2$ satisfy CET(I), CET(IIa), CET(IIb). In this case the two graphs have the same 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 $\mathcal{G}_1$ and $\mathcal{G}_2$ there is an uncovered itinerary, $<X_0,\ldots X_{n+1}>$ 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 $\mathcal{G}_1$, while in $\mathcal{G}_2$ they are not.

It follows from the definition of m.e. conductors that the vertices $X_1,\ldots X_n$, inclusive are *not* ancestors of X_0 or X_{n+1} in $\mathcal{G}_1$. Hence $\{X_1,\ldots X_n\} \cap$ **Sepset**$(X_0,X_{n+1}) = \emptyset$, 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 $\mathcal{G}_1$.) For the same reason Descendants$(\{X_1,\ldots X_n\}) \cap$ **Sepset**$(X_0,X_{n+1}) = \emptyset$. 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 $\mathbf{P}_k \equiv X_k \to \ldots \to X_{k+1}$. Since no descendant of $X_1,\ldots,X_n$ is in **Sepset**(X_0,X_{n+1}), each of the directed paths $\mathbf{P}_k$ 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 adjacent there is some path **Q** d-connecting X_0 and X_1 given **Sepset**(X_0,X_{n+1}). Since each $\mathbf{P}_i$ is out of X_i (i.e. the path goes $X_i \rightarrow \ldots \rightarrow X_2$), by applying Lemma 3.3.1+, with $\mathcal{T} = \{\mathbf{Q},\mathbf{P}_1,\ldots\mathbf{P}_n\}$, and **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, <X_0,X_1,X_2> and <X_{n-1},X_n,X_{n+1}> are not mutually exclusive in $\mathcal{G}_2$, 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: $\mathcal{G}_1$ and $\mathcal{G}_2$ fail to satisfy either CET(IV) or CET(V). We assume that $\mathcal{G}_1$ and $\mathcal{G}_2$ satisfy CET(I)–(III).[19] If $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 $\mathcal{G}_1$ and $\mathcal{G}_2$ <X_0,X_1,…X_n, X_{n+1}>, [20] satisfying the following:

[19]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.

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

(a) if $i > j$ then X_i and X_j are 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, adjacent to X_0 and X_{n+1} in $\mathcal{G}_1$ and $\mathcal{G}_2$, not an ancestor of X_0 or X_{n+1} in $\mathcal{G}_1$ or $\mathcal{G}_2$ and not a descendant of a common child of X_0 and X_{n+1} in $\mathcal{G}_1$ or $\mathcal{G}_2$. As explained in case 3, this implies that in both of the PAGs Ψ_1 and Ψ_2, $X_0 \rightarrow \underline{V} \leftarrow X_{n+1}$.

Since $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 $\mathcal{G}_1$, and V is not a descendant of X_1 in $\mathcal{G}_2$. As in previous cases it is sufficient to show that if Ψ_1 and Ψ_2 are the CCD PAGs corresponding to $\mathcal{G}_1$ and $\mathcal{G}_2$ 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 $\mathcal{G}_1$, in the sense that the shortest directed path $\mathbf{P} \equiv X_k \rightarrow \ldots \rightarrow V$ in $\mathcal{G}_1$ contains at most the same number of vertices as the shortest directed path in $\mathcal{G}_1$ 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 adjacent to V, (both in $\mathcal{G}_1$ and $\mathcal{G}_2$ since by CET(I) $\mathcal{G}_1$ and $\mathcal{G}_2$ have the same adjacencies). We will show that the assumption that V is the closest such vertex to any X_k (in $\mathcal{G}_1$) together with the assumption that $\mathcal{G}_1$ and $\mathcal{G}_2$ satisfy CET(I)-(III) imply that W is a descendant of X_1 in $\mathcal{G}_2$. We prove this by showing that every vertex in the directed subpath $P(X_k, W) \equiv X_k \rightarrow \ldots W$ in $\mathcal{G}_1$ is also a descendant of X_1 in $\mathcal{G}_2$.

Proof of Claim: By induction on the vertices of the path $\mathbf{P}(X_k, W)$.

Base Case: X_k.

By hypothesis X_k is a descendant of X_1 in both $\mathcal{G}_1$ and $\mathcal{G}_2$.

Induction Case: Consider Y_r, where $\mathbf{P}(X_k, W) \equiv X_k \rightarrow Y_1 \rightarrow \ldots \rightarrow Y_r \rightarrow$

$\ldots Y_t$, and $Y_t \equiv W$. By the induction hypothesis, for $s < r$, Y_s is a descendant of X_1 in $\mathcal{G}_2$. Now there are two subcases to consider:

Subcase 1: Not both X_0 and X_{n+1} are adjacent to Y_r. Suppose without loss that X_0 is not adjacent to Y_r. Since in $\mathcal{G}_1$ there is a directed path $X_0 \to \ldots X_k \to Y_1 \to \ldots Y_r$, by Lemma 12 it then follows that there is some subsequence of this sequence of vertices, $\mathbf{Q} \equiv \langle X_0, \ldots Y_r \rangle$ such that consecutive vertices in **Q** are adjacent, but only these vertices are adjacent. Moreover, since X_0 is not 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 $\mathcal{G}_1$ and $\mathcal{G}_2$, (either by the induction hypothesis or by the hypothesis of case 5), but X_0 is not a descendant of X_1 in $\mathcal{G}_1$ or $\mathcal{G}_2$ it follows that D is not an ancestor of X_0 in $\mathcal{G}_1$ or $\mathcal{G}_2$. Hence we may apply Lemma 13, 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 adjacent to Y_r. First note that in $\mathcal{G}_1$ 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 $\mathcal{G}_1$. Moreover, since X_0 and X_{n+1} are not adjacent, $\langle X_0, Y_r, X_{n+1} \rangle$ forms an unshielded non-conductor in $\mathcal{G}_1$. Hence $\langle X_0, Y_r, X_{n+1} \rangle$ forms an unshielded non-conductor in $\mathcal{G}_2$, since by hypothesis $\mathcal{G}_1$ and $\mathcal{G}_2$ satisfy CET(IIa). So Y_r is not an ancestor of X_0 or X_{n+1} in $\mathcal{G}_1$ or $\mathcal{G}_2$. Further, since Y_r is an ancestor of V in $\mathcal{G}_1$ and V is not a descendant of a common child of X_0 and X_{n+1} in $\mathcal{G}_1$, it follows that Y_r is not a descendant of a common child of X_0 and X_{n+1} in $\mathcal{G}_1$. Thus $\langle X_0, Y_r, X_{n+1} \rangle$ forms an unshielded imperfect non-conductor in $\mathcal{G}_1$. Since $\mathcal{G}_1$ and $\mathcal{G}_2$ satisfy CET(IIb), $\langle X_0, Y_r, X_{n+1} \rangle$ forms an unshielded imperfect non-conductor in $\mathcal{G}_2$. Now, if Y_r were not a descendant of X_1 in $\mathcal{G}_2$, 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 $\mathcal{G}_1$). This is a contradiction, hence Y_r is a descendant of X_k in $\mathcal{G}_2$.

This completes the proof of the claim. We now show that $\boldsymbol{\Psi}_1$ and $\boldsymbol{\Psi}_2$ are different.

Consider the edge W∗–∗V in $\boldsymbol{\Psi}_1$. In $\mathcal{G}_1$, W is an ancestor of V, hence it follows from the correctness of the algorithm in $\boldsymbol{\Psi}_1$ this edge is oriented as Wo–∗V or W—∗V. In $\mathcal{G}_2$, 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≡X_1. In this case X_0—>X_1<—X_2, in $\boldsymbol{\Psi}_2$ (and $\boldsymbol{\Psi}_1$). **Supset**(X_0,V,X_2) is the smallest set containing {V} which d-separates X_0 and X_2, in the sense that no subset of **Supset**(X_0,V,X_2) which contains V d-separates X_0 and X_2. It follows from Lemma 7 (with **R** = {V}) that every vertex in **Supset**(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 $\mathcal{G}_2$. Hence in step ¶D of the algorithm given a d-separation oracle for $\mathcal{G}_2$ as input $X_1 \notin$ **Supset**(X_0,V,X_2). Thus step ¶E of the CCD algorithm will orient W∗–∗V in $\boldsymbol{\Psi}_2$ as W<–∗V (unless the edge has already been oriented this way in a previous stage of the algorithm). Thus $\boldsymbol{\Psi}_1$ and $\boldsymbol{\Psi}_2$ 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 **Supset**(X_0,V,X_{n+1})∪{W} in $\mathcal{G}_2$.

Proof. We have already shown that W is a descendant of X_1, and so also of X_n in $\mathcal{G}_1$ and $\mathcal{G}_2$. Since in both $\mathcal{G}_1$ and $\mathcal{G}_2$ X_0 is 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 $\mathcal{G}_1$ and $\mathcal{G}_2$ there is a directed path $\mathbf{P}_0$ 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 $\mathbf{P}_1$ from X_1 to W. Concatenating $\mathbf{P}_0$ and $\mathbf{P}_1$ 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 $\mathbf{Q}^*$ 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 **Supset**(X_0,V,X_{n+1}) is an ancestor of X_0, X_{n+1} or V, it follows that no vertex in **Supset**(X_0,V,X_{n+1}) lies on **P*** or $\mathbf{Q}^*$ (X_0, $X_{n+1} \notin$ **Supset**(X_0,V,X_{n+1}) by definition). It now follows by Lemma 3.3.1+ that we can concatenate $\mathbf{P}^*$ and $\mathbf{Q}^*$ to form a path which d-connects X_0 and X_{n+1} given 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 the edge has 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 $\mathcal{G}_1$ and $\mathcal{G}_2$ may fail to satisfy CET(I)-(V), this completes the proof that the CCD algorithm locates the d-separation equivalence class. ∴

References

BOLLEN, K. (1989) Structural Equations with Latent Variables, Wiley, NY.

KOSTER, J.T.A.(1994) Markov Properties of Non-Recursive Causal Models. MS.

LAURITZEN, S, DAWID, A., LARSEN, B., LEIMER, H., (1990). Independence properties of directed Markov fields, *Networks* **20**

PEARL, J. (1988). Probabilistic Reasoning in Intelligent Systems. Morgan

Kaufmann, San Mateo, CA.
PEARL, J. (1993). On the Statistical Interpretation of Structural Equations. Technical Report R-200, Cognitive Systems Lab., UCLA.
RICHARDSON, T. (1994). Properties of Cyclic Graphical Models. MS Thesis, Carnegie Mellon University
RICHARDSON, T. (1996). Discovering Cyclic Causal Structure. Tech Report CMU-PHIL 68.
SPIRTES, P. (1995). Directed Cyclic Graphical Representations of Feedback Models. *Proceedings UAI 95*, ed. P. Besnard and S. Hanks, Morgan Kaufmann San Mateo, CA 491-498.
SPIRTES, P., GLYMOUR , C. & SCHEINES , R. (1993) Causation, Prediction & Search. Lecture Notes in Statistics, Springer-Verlag.
VERMA, T. & PEARL, J., (1990). On Equivalence of Causal Models. Technical Report R-150, Cognitive Systems Lab., UCLA.