Estimating Causal Effects with Ancestral Graph Markov Models

Daniel Malinsky, Peter Spirtes

July 13, 2016

Technical Report No. CMU-PHIL-194

Philosophy

Methodology

Logic

Carnegie Mellon

Pittsburgh, Pennsylvania 15213

Estimating Causal Effects with Ancestral Graph Markov Models

Daniel Malinsky MALINSKY@CMU.EDU
Peter Spirtes PS7Z@ANDREW.CMU.EDU
Carnegie Mellon University
Pittsburgh, USA

Abstract

We present an algorithm for estimating bounds on causal effects from observational data which combines graphical model search with simple linear regression. We assume that the underlying system can be represented by a linear structural equation model with no feedback, and we allow for the possibility of latent variables. Under assumptions standard in the causal search literature, we use conditional independence constraints to search for an equivalence class of ancestral graphs. Then, for each model in the equivalence class, we perform the appropriate regression (using causal structure information to determine which covariates to include in the regression) to estimate a set of possible causal effects. Our approach is based on the "IDA" procedure of Maathuis et al. (2009), which assumes that all relevant variables have been measured (i.e., no unmeasured confounders). We generalize their work by relaxing this assumption, which is often violated in applied contexts. We validate the performance of our algorithm on simulated data and demonstrate improved precision over IDA when latent variables are present. This is an extended version of a conference paper (Malinsky and Spirtes, 2016).

Keywords: Causal inference, ancestral graphs, latent variables, Markov equivalence

1. Introduction

It is well known that regression estimates for causal effects will be biased unless a variety of conditions on the data are satisfied; methods which correct for confounding by covariate adjustment depend on facts about the causal structure of the system under study (e.g., whether all the relevant variables have been measured and how the measured covariates are causally linked to the variables of interest). Maathuis et al. (2009) provide a good overview and explanation of this idea; see also Entner et al. (2013) for related analysis. Roughly speaking, regressing Y on X while controlling for additional covariates does not produce an unbiased estimate of the effect of intervening on X unless the additional covariates account for any possible confounding of X and Y. In the language of causal graphs, the covariates must block all causal pathways from variables (measured or not) which are causes of both X and Y and the covariates should not include effects of X. The conditions under which regression can produce an unbiased estimate of a causal effect can be readily translated into conditions on an appropriate causal graphical model (Pearl 2009).

The method proposed here combines techniques from automated causal search and regression to estimate causal effects (also called intervention effects) from observational data. In particular, the algorithms described in section 4 estimate causal effects even when there are relevant unmeasured variables (i.e., "latent confounding" or "causal insufficiency"). The method is based on the one developed by Maathuis et al. (2009), which has been fruitfully applied in the context of genetics research (Maathuis et al., 2010; Stekhoven et al., 2012). The IDA ("Intervention when the DAG is Absent") algorithm of Maathuis et al. is consistent under a set of assumptions which includes causal sufficiency: the assumption that no variables which are common direct causes of at least two measured variables are unmeasured. Importantly, IDA is feasible in high-dimensional settings, where sample sizes are small but the number of covariates is very large. In their genetics applications there are more than 4000 variables, and the goal is to find variables which are likely strong regulators (causes) of some chosen variable of interest in order to prioritize gene knock-out experiments. In the data which is typical in the social sciences and many areas of biomedical research, the assumption of causal sufficiency is often unwarranted. Even genome-wide expression data may be causally insufficient if there are unmeasured factors like proteins which act as common causes of multiple gene expressions. Our procedure is consistent in the presence of latent common causes and is feasible for large numbers of variables.

The work of Pearl and his collaborators (e.g., Tian and Pearl, 2002; Shpitser and Pearl, 2006) provides techniques for calculating the outcomes of interventions when the true causal structure (i.e., true causal graph) is known. These results relate to the general conditions for "back-door adjustment" and "front-door adjustment" described in Pearl (2009). The back-door criterion is a graphical criterion that is sufficient for adjustment in the following sense: if a set of variables satisfies the back-door criterion for a given graph, then conditioning on that set is sufficient for estimating intervention effects from observed distributions alone. Maathuis and Colombo (2015) generalize the back-door criterion to different types of graphical objects, and their result will play an instrumental role in the algorithms we propose. In order to estimate the intervention effects from data, the researcher must be able to identify the set of covariates which satisfy the back-door criterion. To determine which variables satisfy this condition without substantial background causal knowledge, we use an automated causal search algorithm called FCI (Spirtes et al., 1995; Zhang, 2008b).

One alternative approach to estimating causal effects is worth mentioning here. Algorithms which learn latent variable LiNGAM models (Hoyer et al., 2008; Kawahara et al., 2010; Entner and Hoyer, 2010; Tashiro et al., 2014) allow for the possibility of unmeasured variables. These algorithms exploit assumptions about the causal structure (assumed to be structural equation models which are acyclic, linear, and which have non-Gaussian error terms) to estimate graphical structure and some estimate causal strength parameters simultaneously. See also Henao and Winther (2011) and Shimizu and Bollen (2014) for related Bayesian procedures. One substantial benefit to these algorithms is that they can often identify a unique model or a smaller equivalence class of models than the FCI algorithm can. Unfortunately, computational complexity makes these algorithms mostly infeasible in applied contexts when there are more than a few variables and the sample sizes required are unrealistic for many applications. Furthermore, these algorithms generally require that the researcher stipulates the number of (possible) latent variables explicitly; the approach proposed here is more general in that it does not make any assumptions about the number of (possible) unmeasured variables.

Though our procedure cannot always pin down a unique causal graphical model, from an equivalence class of graphs we can estimate bounds on causal effects. That is, for a given variable pair (X, Y) we can calculate a set of estimates for the causal effect of X on Y. Each estimate corresponds to some model in the equivalence class. The minimum and maximum estimates in such a set are bounds on the true causal effect, and these bounds can be used to prioritize follow-up experiments by, for example, concentrating on experimental manipulations of variables with effects bounded away from zero.

2. Definitions and Background

It is assumed here that the causal structure of the system under study can be represented by a Directed Acyclic Graph (a DAG). A graph $\mathcal{G}$ is a pair $(\mathbf{V}, \mathbf{E})$ where $\mathbf{V}$ is a set of vertices corresponding to random variables $\mathbf{V} = \{X_1, ..., X_p\}$ and $\mathbf{E}$ is a set of edges. A DAG contains only directed edges ($\rightarrow$) and has no cycles (no sequence of directed edges from any variable to itself). If $X_i \rightarrow X_j$ then X_i is called a parent of X_j, and X_j is a child of X_i. Two variables are adjacent if there is some edge between them, and a path is a sequence of distinct adjacent vertices (e.g., $X_i \leftarrow X_j \leftarrow X_k \rightarrow X_l$). A directed path from X_i to X_j is a path which contains only directed edges away from X_i and toward X_j. When there is a directed path from X_i to X_j we call X_i an ancestor of X_j, and X_j is a descendent of X_i. Denote the set of parents of a vertex X in $\mathcal{G}$ by $pa(X, \mathcal{G})$, and the sets of ancestors of X and descendents of X by $An(X, \mathcal{G})$ and $De(X, \mathcal{G})$ respectively. The adjacency set of X is $adj(X, \mathcal{G})$. A v-structure is a triple $\langle X_i, X_j, X_k \rangle$ such that $X_i \rightarrow X_j$, $X_j \leftarrow X_k$ and X_i and X_k are not adjacent. X_j is called a collider because X_i and X_k "collide" at X_j. A collider which is part of a v-structure (i.e., a collider with non-adjacent parents) is also called an unshielded collider.

In a causal DAG, $X_i \rightarrow X_j$ if and only if X_i is a direct cause of X_j relative to $\mathbf{V}$. We assume that our candidate causal models satisify the Causal Markov Condition (CMC) and the Causal Faithfulness Condition (CFC). See Spirtes et al. (2000) for discussion of these assumptions. The CMC requires that every variable in $\mathbf{V}$ is independent of its non-descendents conditional on its parents in the causal graph, i.e., that the joint probability distribution $f(\mathbf{V}) = \prod_{X_i \in \mathbf{V}} f(X_i | pa(X_i, \mathcal{G}))$. The CFC stipulates that the only independencies that are true in the population are the ones implied by the CMC, or equivalently, that the only independence relationships are the ones reflected in Pearl's graphical criterion of *d-separation* (Pearl, 2009). This is a way of stipulating that there is no accidental "cancelling out" of causal pathways, or independencies which are the result of special (measure-zero) parameterizations. Two DAGs are called Markov equivalent if they encode all the same independence relationships among the observed variables. DAGs which share all the same adjacencies and all the same v-structures form a Markov equivalence class (Verma and Pearl, 1991).

A Markov equivalence class can be represented by a single graph, called a Pattern or CPDAG. A Pattern or CPDAG has all the same adjacencies as each DAG in the equivalence class but can contain undirected edges ($-$) in addition to directed edges. An undirected edge $X_i - X_j$ indicates that some DAG in the equivalence class contains $X_i \leftarrow X_j$ and some DAG contains $X_i \rightarrow X_j$. If $X_i - X_j$ in a CPDAG, X_i is called a sibling of X_j and we denote the set of siblings of X by $sib(X, \mathcal{G})$. The PC algorithm of Spirtes et al. (2000) assumes the CMC and CFC to search for a CPDAG. If some of the variables in the set $\mathbf{V}$ are unmeasured, we represent the system with a causal MAG (Maximal Ancestral Graph) over the measured variables. A MAG is a kind of mixed graph so it may have the following kinds of edges: $\rightarrow$ and $\leftrightarrow$. More generally, if we include the possibility of selection variables, a MAG can also have undirected edges, but we will not consider selection variables here.[1] A MAG represents a DAG after all latent variables have been marginalized out, and it preserves all entailed conditional independence relations among the measured variables which are true in the underlying DAG. In a MAG $\mathcal{M}$, a tail mark at X_i (e.g., $X_i \rightarrow X_j$) means that X_i is an ancestor of X_j in all DAGs represented by $\mathcal{M}$. An arrowhead at X_i (e.g., $X_i \leftarrow X_j$ or $X_i \leftrightarrow X_j$) means that X_i is not an ancestor of X_j in all DAGs represented by $\mathcal{M}$. A $\leftrightarrow$ edge between two variables indicates that neither variable is an ancestor of the other (though they are probabalistically

[1] So technically speaking what we call a MAG is a DMAG (a Directed MAG) in the parlance of Zhang and Spirtes (2005).

dependent). See Richardson and Spirtes (2002) for details on MAGs. A Markov equivalence class of MAGs is represented by a PAG (Partial Ancestral Graph), which (possibly) has edges with the additional "circle" edge mark $\circ$ (e.g., $X_i \circ\!\!\rightarrow X_j$). This indicates that in some MAG in the equivalence class there is an arrowhead at X_i and in some other MAG there is a tail at X_i. So, the PAGs we will consider (again, excluding the possibility of selection variables) can have the following edges: $\rightarrow$, $\circ\!\!\rightarrow$, $\circ\!\!-\!\!\circ$, and $\leftrightarrow$. The FCI algorithm assumes the CMC and CFC to search for a PAG.

The total causal effect on Y of an intervention on X_i, written do($X_i = x_i'$) in Pearl's (2009) notation, is $\frac{\partial}{\partial x}\mathbf{E}(Y|\text{do}(X_i = x))|_{x=x_i'}$. That is, we are interested in the change in the expected value of Y when we intervene to change the value of X_i by one unit. For a DAG which represents a linear structural equation model, the total causal effect of X_i on Y with $Y \notin pa(X_i, \mathcal{G})$ is the regression coefficient of X_i in the regression of Y on X_i and $pa(X_i, \mathcal{G})$. Call this regression coefficient $\beta_{i|pa(X_i,\mathcal{G})}$. See Maathuis et al. (2009: 3138) for details on this. If $Y \in pa(X_i, \mathcal{G})$ the causal effect is 0. More generally, for any set $S \subseteq \{X_1, ..., X_p, Y\} \setminus \{X_i\}$, we write $\beta_{i|S}$ to denote the coefficient of X_i in the linear regression of Y on X_i and S, and let $\beta_{i|S} = 0$ if $Y \in S$. The reason we include the parents of X_i in the regression of Y on X_i in calculating the total effect is because $pa(X_i, \mathcal{G})$ is sufficient to block all causal pathways from variables which are causes of both X_i and Y. Another way of putting this is that the set $pa(X_i, \mathcal{G})$ satisfies Pearl's "back-door criterion" for DAGs (Pearl, 2009: ch. 3). Maathuis and Colombo (2015) extend Pearl's back-door criterion for DAGs to the graphical structures above: CPDAGs, MAGs, and PAGs. The sufficient back-door set is more complicated but the principle is the same. We will summarize their result in section 4 and use it to propose a general algorithm for estimating causal effects from PAGs.

3. The IDA Approach

Maathuis et al. (2009) provide algorithms to estimate causal effects under the following assumptions: they assume that the data is generated from an unknown DAG; they assume the Causal Markov Condition and Causal Faithfulness Condition hold; they assume a set of jointly Gaussian variables $\{X_1, ..., X_p, Y\}$; and they assume causal sufficiency, i.e., that there are no unmeasured common causes. The Gaussianity assumption can be weakened to only linearity; joint Gaussianity implies linearity but only linearity is needed so that the total causal effects can be identified with coefficients in linear regressions.[2] Effectively, Maathuis et al. are assuming that the system under study can be represented by a linear structural equation model with no feedback. We will discard the assumption of causal sufficiency in the next section.

In their "global" algorithm, Maathuis et al. begin by searching for a CPDAG from their data with PC. Then, they list all the DAGs in the equivalence class represented by this CPDAG. For each DAG $\mathcal{G}_j$ ($j = 1, ..., m$) in the equivalence class, they regress Y on each non-descendent X_i along with $pa(X_i, \mathcal{G}_j)$ in order to estimate the causal effect θ_{ij}. They collect the θ_{ij}'s in a $p \times m$ matrix Θ, where the columns correspond to covariates and the rows correspond to DAGs in the equivalence class. The "global" IDA algorithm is very slow if the number of covariates is large, because of the step that lists all the DAGs in the equivalence class. For the intended application (genetics data with $p > 4000$) this is infeasible. So, Maathuis et al. propose a second algorithm which is much faster because it only requires "local" information. The key is that for each DAG $\mathcal{G}_j$, one only needs to

[2] The current implementation of their algorithm uses independence tests based on Fisher's z-score, which is only a test of independence when the data is jointly Gaussian. Future implementations can incorporate more general tests of independence instead, e.g., Zhang et al. (2011) or Ramsey (2014).

know the back-door set $pa(X_i, \mathcal{G}_j)$ in order to carry out the regression. Knowledge of the rest of the graph is not necessary. Maathuis et al. exploit this fact in their "local" algorithm. Starting with a CPDAG, the algorithm needs only to examine possible parent sets by orientating undirected edges with vertices in $sib(X_i, \mathcal{G})$. The orientations considered must preserve Markov equivalence; see Maathuis et al. (2009: 3141-3143).

The substantial increase in speed comes at a price, however; the local IDA algorithm sacrifices information about which causal effect estimate comes from which DAG in the equivalence class. Instead of producing the complete matrix Θ, IDA outputs multisets (which are collections in which members are allowed to appear more than once) Θ_i^L of causal effects for each covariate X_i. Each element of the Θ_i^L is the causal effect of X_i on Y in *some* DAG represented by the CPDAG, but we do not know which one. Maathuis et al. prove that Θ_i and Θ_i^L are equal ($i = 1, ..., p$) when they are interpreted as sets (2009: Theorem 3.2). They also provide a sample version of this algorithm, prove its consistency under a variety of assumptions (concerning sparsity of the graph, etc.), and validate it on the genetics dataset by using it to pick out the variables with the largest minimum causal effect. See their paper for a full discussion.

4. Intervention Effects in Causally Insufficient Systems

In this section we sketch two algorithms analogous to the ones presented by Maathuis et al. without the assumption of causal sufficiency. Our algorithm takes the output of FCI (a PAG) as input, and so we must work with the set of MAGs represented by that PAG. In following the procedure of global IDA, we would like to list all the MAGs $\mathcal{M}_1, ..., \mathcal{M}_n$ represented by a PAG $\mathcal{P}$, and estimate the matrix of causal effects. But what set do we regress Y on? We need a back-door set for (X_i, Y) in each MAG. In order to construct a sufficient adjustment set we need several definitions. First, let a collider path from X_i to X_j be a path on which every vertex (except the endpoints) is a collider.

Definition 4.1 (*Visible and invisible edges*) *All directed edges in DAGs and CPDAGs are said to be visible. Given a MAG $\mathcal{M}$ / PAG $\mathcal{P}$, a directed edge $X \to Y$ in $\mathcal{M}$ / $\mathcal{P}$ is visible if there is a vertex Z not adjacent to Y, such that there is an edge between Z and X that is into X, or there is a collider path between Z and X that is into X and every non-endpoint vertex on the path is a parent of Y. Otherwise $X \to Y$ is said to be invisible.*

Definition 4.2 (*D-SEP*$(X, Y, \mathcal{G})$) *Let X and Y be two distinct vertices in mixed graph $\mathcal{G}$. We say that $V \in$ D-SEP$(X, Y, \mathcal{G})$ if $V \neq X$ and there is a collider path between X and V in $\mathcal{G}$, such that every vertex on this path is an ancestor of X or Y in $\mathcal{G}$.*

Definition 4.3 ($\mathcal{R}$ *and* $\mathcal{R}_{\underline{X}}$) *Let X be a vertex in $\mathcal{G}$, where $\mathcal{G}$ represents a causal DAG, CPDAG, MAG, or PAG. Let $\mathcal{R}$ be a DAG or MAG represented by $\mathcal{G}$, in the following sense. If $\mathcal{G}$ is a DAG or MAG, we simply let $\mathcal{R} = \mathcal{G}$. If $\mathcal{G}$ is a CPDAG/PAG, we let $\mathcal{R}$ be a DAG/MAG in the Markov equivalence class described by $\mathcal{G}$ with the same number of edges into X as $\mathcal{G}$. Let $\mathcal{R}_{\underline{X}}$ be the graph obtained from $\mathcal{R}$ by removing all directed edges out of X that are visible in $\mathcal{P}$.*

All of these definitions can be found in Maathuis and Colombo (2015); the definition of visible/invisible edges is a generalization of the standard one introduced in Zhang (2008a). A visible edge between X and Y in a MAG or PAG picks out an ancestral relationship that is incompatible with any latent common cause between X and Y in the underlying DAG. $possibleDe(X, \mathcal{G})$ is defined as the set of possible descendents of X in $\mathcal{G}$, where X_i is a possible descendent of X_j if there is a path from X_j to X_i with no arrowhead pointing towards X_j. $possibleDe(X, \mathcal{G})$ and $De(X, \mathcal{G})$ are equal if $\mathcal{G}$ is a MAG. Maathuis and Colombo (2015) prove the following theorem:

Theorem 4.1 (*Back-door Set*) *Let X and Y be two distinct vertices in a causal DAG, CPDAG, MAG, or PAG $\mathcal{G}$. Let $\mathcal{R}$ and $\mathcal{R}_{\underline{X}}$ be defined as above. If $Y \in adj(X, \mathcal{R}_{\underline{X}})$ or D-SEP$(X, Y, \mathcal{R}_{\underline{X}}) \cap possibleDe(X, \mathcal{G}) \neq \emptyset$, then $f(y|do(x))$ is not identifiable via the generalized back-door criterion. Otherwise D-SEP$(X, Y, \mathcal{R}_{\underline{X}})$ satisfies the generalized back-door criterion relative to (X, Y) and $\mathcal{G}$.*

The set D-SEP$(X_i, Y, \mathcal{M}_{\underline{X_i}})$, when the antecedent condition is not met, is a back-door set for (X_i, Y) in MAG $\mathcal{M}$ so we can take the coefficient of X_i in the regression of Y on X_i and D-SEP$(X_i, Y, \mathcal{M}_{\underline{X_i}})$ to be the causal effect of X_i on Y in $\mathcal{M}$.

Algorithm 4.1: LV-IDA("*global*")

```
Input: PAG $\mathcal{P}$, conditional dependencies of $X_1, ..., X_p, Y$
Output: Matrix $\Theta$ of possible causal effects
List the MAGs $\mathcal{M}_1, ..., \mathcal{M}_n$ in the equivalence class of $\mathcal{P}$.
for $j = 1$ to $n$
    for $i = 1$ to $p$
        if $Y \notin De(X_i, \mathcal{M}_j)$ then $\theta_{ij} = 0$
        if $Y \in adj(X_i, \mathcal{M}_{j,\underline{X_i}})$ or D-SEP$(X_i, Y, \mathcal{M}_{j,\underline{X_i}}) \cap De(X_i, \mathcal{M}_j) \neq \emptyset$
            then $\theta_{ij} =$ "NA"
        else $\begin{cases} S = \text{D-SEP}(X_i, Y, \mathcal{M}_{j,\underline{X_i}}) \\ \theta_{ij} = \beta_{i|S} \end{cases}$
    end
end
```

Algorithm 4.1 is the "global" algorithm. Listing all the MAGs represented by a PAG is more complicated than listing all the DAGs represented by a CPDAG. In the latter case, there are well-known and efficient algorithms which orient undirected edges and exhaustively apply orientation rules (to orient remaining undirected edges) which preserve Markov equivalence; see Meek (1995). No such procedures are currently known for PAGs. One would need a way of transforming circle marks on $\circ\!\!\rightarrow$ and $\circ\!\!-\!\!\circ$ edges into tails and arrowheads, and deciding which further orientations in the graph are implied by these new tails and arrowheads, while preserving Markov equivalence. This is because some combinations of transformations could introduce new independence relationships among the variables, e.g., if transforming two circles into arrowheads simultaneously creates a new v-structure.

The naive approach would be a brute force method that exhaustively tries every combination of circle mark transformations, and then checks if the resulting graph is Markov equivalent to the starting graph using the procedure introduced by Ali et al. (2009). This approach would be exceedingly slow. For large graphs with many circle marks, there are just too many possible combinations of transformed marks and checking Markov equivalence for every resultant graph would require a lot of computation time. We pursued an alternative approach to enumerate the list of MAGs more quickly. The procedure is based on a suggestion by Jiji Zhang, and it exploits a transformational characterization of equivalence between MAGs introduced in Zhang and Spirtes (2005). We call it the ZML (Zhang MAG Listing) algorithm, and it is described in the appendix.

Even with the ZML algorithm for enumerating MAGs, the "global" LV-IDA is too slow for even moderately-sized graphs (e.g., more than 15 or 20 variables). The "local" IDA algorithm operates on the principle that one only needs to know enough information about the DAGs in the equivalence class to determine what the possible back-door sets are. Similarly, for a "local" version of the above algorithm one only needs to know enough about the MAGs to calculate the back-door set.

For the local algorithm, we need to define the set Possible-D-SEP($X_i, Y, \mathcal{G}$), abbreviated as $pds(X_i, Y, \mathcal{G})$:

Definition 4.4 ($pds(X_i, Y, \mathcal{G})$) *Let $V \in pds(X_i, X_j, \mathcal{G})$ if and only if there is a path π between X_i and V in $\mathcal{G}$ such that for every subpath $< X_m, X_l, X_h >$ on π either X_l is a collider on the subpath in $\mathcal{G}$ or $< X_m, X_l, X_h >$ is a triangle in $\mathcal{G}$.*

A triangle is a triple $\langle X_m, X_l, X_h \rangle$ where each pair of vertices is adjacent. There are alternative definitions of $pds(X_i, Y, \mathcal{G})$ which make the set smaller (but potentially more computationally intensive to search for), see Colombo et al. (2012).[3] In order to compute D-SEP($X_i, Y, \mathcal{M}_{\underline{X_i}}$) and check if $Y \in adj(X_i, \mathcal{M}_{\underline{X_i}})$ or D-SEP($X_i, Y, \mathcal{M}_{\underline{X_i}}$)$\cap De(X_i, \mathcal{M}) \neq \emptyset$, we only need the variables in $possibleDe(X_i, \mathcal{P}) \cup pds(X_i, Y, \mathcal{P})$. The set $pds(X_i, Y, \mathcal{P})$ (which includes all the adjacencies of X_i the way it is defined here) is sufficient for determining which edges out of X_i are visible (for constructing $\mathcal{M}_{\underline{X_i}}$). $pds(X_i, Y, \mathcal{P})$ is also needed for checking if $Y \in adj(X_i, \mathcal{M}_{\underline{X}})$ and for constructing D-SEP($X_i, Y, \mathcal{M}_{\underline{X_i}}$). The set of possible descendents of X_i is needed to check whether D-SEP($X_i, Y, \mathcal{M}_{\underline{X_i}}$)$\cap De(X_i, \mathcal{M}) \neq \emptyset$. Knowing the induced subgraph over these variables is at least sufficient for calculating the back-door set for (X_i, Y) in $\mathcal{P}$. We propose Algorithm 4.2.

Essentially we just run the "global" algorithm on the subgraph over the set which is sufficient to calculate all the local back-door sets. This algorithm is really only "semi-local" in the sense that one might have to list a large number of MAGs if the number of vertices in $\mathbf{Z_i}$ is large. However, if the number of vertices in $\mathbf{Z_i}$ is manageably small, this algorithm could be substantially faster than the "global" algorithm. Indeed, the set $\mathbf{Z_i}$ seems to be small enough to run the ZML algorithm in all the simulated trials we ran, which included graphs of over 100 variables.[4]

As with the local IDA algorithm, we sacrifice some information: we no longer know which estimated causal effects correspond to which graphs in the equivalence class. We also cannot determine how many graphs in the equivalence class imply a particular causal effect estimate. Fortunately, we do not sacrifice anything else, as evinced by Theorem 4.2:

Theorem 4.2 *The local and global versions of LV-IDA produce the same output, when the output is interpreted as a set. That is, $\Theta_i \overset{set}{=} \Theta_i^L$ for all $i = 1, ..., p$.*

The proof is in the appendix. This is directly analagous to Theorem 3.2 in Maathuis et al. (2009). Note that the output of LV-IDA may contain elements which are labeled "NA". The causal effects of some variables may not be identifiable by Maathuis and Colombo's generalized back-door criterion, as is clear from the definition. They may sometimes be identifiable by other means (Maathuis and

[3] In our implementation we use both the definition above as well as a variant which requires that V is an ancestor of either X_i or Y.

[4] For large graphs, we used RFCI due to Colombo et al. (2012) instead of FCI to perform the initial PAG search.

Colombo, 2015; Perković et al., 2015; Hyttinen et al., 2015). When an LV-IDA estimate is "NA" this indicates that the measured set of covariates is not sufficient to rule out (using the back-door criterion) confounding in some MAG consistent with the data. Unless one can rule out confounding by background knowledge, one may attribute an arbitrary proportion of the observed correlation between two variables to a latent variable. The number of identifiable, non-zero effects is largely determined by the presence of visible edges in the graph, which of course depends on the causal structure and which covariates are measured. IDA assumes that all causal effects are identifiable by ruling out latent common causes. As a consequence, there may be variable pairs for which IDA will estimate non-trivial effect bounds, but which are not identifiable under the less restrictive assumptions of LV-IDA.

Algorithm 4.2: LV-IDA("$local$")

```
Input: PAG $\mathcal{P}$, conditional dependencies of $X_1, ..., X_p, Y$
Output: Multisets $\Theta_i^L, i = 1, ..., p$
for $i = 1$ to $p$
    Form the set $\mathbf{Z_i} = possibleDe(X_i, \mathcal{P}) \cup pds(X_i, Y, \mathcal{P})$.
    Form $\mathcal{P}^*$, the subgraph of $\mathcal{P}$ over vertices $\mathbf{Z_i}$.
    List the MAGs $\mathcal{M}_1, ..., \mathcal{M}_m$ represented by $\mathcal{P}^*$.
        for $k = 1$ to $m$
            if $Y \notin De(X_i, \mathcal{M}_k)$ then add $\theta_{ik} = 0$ to $\Theta_i^L$
            if $Y \in adj(X_i, \mathcal{M}_{k,\underline{X_i}})$ or D-SEP$(X_i, Y, \mathcal{M}_{k,\underline{X_i}}) \cap De(X_i, \mathcal{M}_k) \neq \emptyset$
                then add $\theta_{ik} =$ "NA" to $\Theta_i^L$
            else $\begin{cases} S = \text{D-SEP}(X_i, Y, \mathcal{M}_{k,\underline{X_i}}) \\ \text{add } \theta_{ik} = \beta_{i|S} \text{ to } \Theta_i^L \end{cases}$
        end
end
```

Hyttinen et al. (2015) introduce a procedure which combines an ASP constraint solver with a version of the do-calculus to calculate causal effects in graphs with latent variables. For small graphs, they find that their approach is faster than a procedure which naively enumerates all the Markov equivalent graphs. Their enumeration procedure differs from the one proposed here – rather than "naive enumeration" we use the ZML algorithm. Further, we exploit the locality of back-door adjustment, and use regression instead of estimation via the do-calculus (which would be much slower). All of these differences contribute to the feasibility of our algorithm on large graphs. The procedure in Hyttinen et al., however, may identify some causal effects which are unidentifiable by LV-IDA, since the do-calculus algorithm they use is complete and the generalized back-door criterion is not. More recently, Perković et al. (2016) have proposed a complete adjustment criterion (and constructive adjustment set). In future work these results can be combined with LV-IDA to perhaps increase the number of identifiable effects.

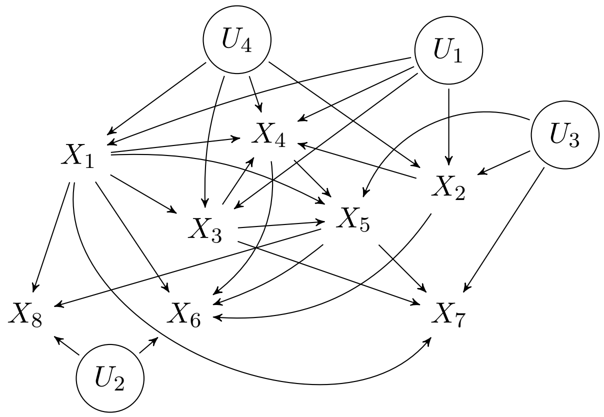

Figure 1: A simulated DAG with several unmeasured confounders $U_1, ..., U_4$. The true causal effects of X_5 on X_6 and X_5 on X_7 are 0.894 and 1.143, respectively. LV-IDA produces the estimates $\{\text{NA}, 0.894, 1.345, 1.707\}$ and $\{\text{NA}, 0, 1.143, 1.662\}$, respectively. IDA produces the estimates $\{1.345, 1.481\}$ and $\{1.603, 1.662\}$, respectively.

5. Simulations

First, we show an example of how LV-IDA and IDA compare in the infinite-sample limit. We simulate a DAG with 8 measured variables and 4 latents. The DAG is parameterized as a linear Gaussian structural equation model. See Figure 1. We run PC and FCI on the true covariance matrix, and then apply IDA and LV-IDA to estimate intervention effects on the output of PC and FCI respectively. LV-IDA is successful in the sense that the true causal effect is contained within the estimated set of possible effects, but IDA gets it wrong. When we estimate the causal effect of X_5 on X_6 using LV-IDA we get $\{\text{NA}, 0.894, 1.345, 1.707\}$, and using IDA we get $\{1.345, 1.481\}$. The true effect size is 0.894 so the output of LV-IDA contains the true value while the output of IDA does not. For the effect of X_5 on X_7, LV-IDA yields $\{\text{NA}, 0, 1.143, 1.662\}$ and IDA yields $\{1.603, 1.662\}$. The true effect is 1.143 so again the output of LV-IDA contains the true value while the output of IDA does not. Note that LV-IDA can produce a set of estimates which includes both "NA" and the true value, and it can also produce estimates which contain the true value and no "NA" while IDA gets it wrong. In general, IDA will yield estimates which do not include the true value in the causally insufficient setting because PC may return graphs with spurious edges or incorrect orientations even in the infinite sample limit. FCI will not make such mistakes in the infinite sample limit.

Next, we ran a number of finite sample simulations. We generated 100 random sparse DAGs with 15 variables, 4 or 5 of which are latent. We parameterized these with linear Gaussian structural equations (coefficients distributed ±Uniform[0.5,1.5]) and generated data vectors with $n = 1000$ samples. We searched for a CPDAG using PC, for a PAG using a variant of FCI, and then used these as inputs to IDA and LV-IDA. The PAG search was done with GFCI, a procedure which mixes greedy score-based search with conditional independence tests (Ogarrio et al., 2016). GFCI achieves better performance in finite samples as compared with FCI. In both PC and GFCI the α

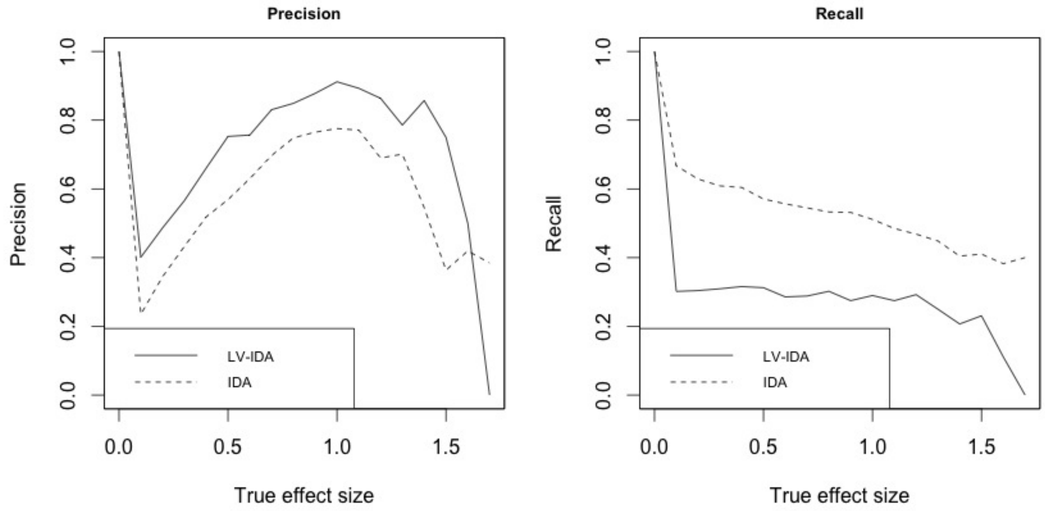

Figure 2: Precision and recall plots for simulation study, described in the text.

tuning parameter was set to 0.01.[5] For every pair of variables in each graph, we estimated the total causal effect and compared our estimates with the true value. In the case of LV-IDA, we confine our results to causal effects which are identifiable, i.e., which have no “NA” among the set of estimates. (About 12.7 percent of estimated effects had an “NA” for some graph in the equivalence class.) Both LV-IDA and IDA can produce multiple estimates for a particular causal effect, so we choose the best estimate to compare with the true value from among the multiset. LV-IDA is more accurate than IDA in terms of mean squared error: the MSE for LV-IDA was 0.022 and the MSE for IDA was 0.056. We plot precision and recall in Figure 2. For both IDA and LV-IDA we use the minimum absolute value estimate in the multiset of causal effects, following Maathuis et al. (2009). While LV-IDA does worse than IDA with respect to recall, it does better with respect to precision. That is, if LV-IDA identifies a large effect estimate (in absolute value), then the true effect is likely to be large (in absolute value). For the intended application of IDA – finding a manageable number of strong regulators in a genetic regulatory network to prioritize knock-out experiments – precision is more important than recall. Our simulation results suggest that in many cases, true large effects are possibly confounded and thus not identifiable. Fortunately we need only correctly identify a small number of true large effects to plan follow-up experiments, and for this task LV-IDA does well.

The performance of LV-IDA is contingent on the accuracy of the underlying PAG search. IDA has been improved by variations on PC like PC-stable (Colombo and Maathuis, 2014) and with stability selection techniques (Stekhoven et al., 2012). Similar steps may likewise improve the performance of LV-IDA.

[5] IDA and PC are implemented in the R package pcalg (Kalisch et al., 2012) and our LV-IDA is also implemented in R. For GFCI and the data generation we used the TETRAD software: https://github.com/cmu-phil/tetrad.

6. Conclusion

The LV-IDA algorithm is a straightfoward extension of the IDA algorithm to the domain of causally insufficient systems, i.e., systems with possible unmeasured confounding. Thus, LV-IDA makes estimating (sets of) intervention effects possible when an unknown number of possibly relevant variables have been left out of the model. Although it may not be feasible to run LV-IDA on very high-dimensional data sets with thousands of variables, it can be applied to local regions of a large graph (e.g., the Markov blanket of some variable of interest). The result of this kind of localized application of LV-IDA should be correct, since ancestral Markov models are closed under marginalization (Richardson and Spirtes, 2002). Then, identified causal effect estimates which are bounded away from zero can be used to prioritize follow-up experiments. In any case, LV-IDA improves on IDA when the research goal requires accurate estimation of intervention effects that account for possible bias from latent variables. Sometimes the causal effect of interest is not identifiable from the current set of measured covariates. In such cases, bounds on causal effects may be misleading so the researcher would be advised to expand their set of measured variables or try to identify the effect by other means.

Acknowledgments

The authors would like to thank Clark Glymour and Joseph Ramsey. This research was supported by grant U54HG008540 awarded by the National Institutes of Health (NIH).

Appendix A.

The subgraph consisting of vertices on $\circ\!-\!\circ$ in a PAG $\mathcal{P}$ is called the circle component of the graph, written $C(\mathcal{P})$.

Algorithm A.1: ZML()

```
Input: PAG $\mathcal{P}$
Output: A list of the MAGs represented by $\mathcal{P}$, called $[\mathcal{P}]$
1. Let $\mathcal{M} = \mathcal{P}$.
2. Transform all $\circ\!\!\rightarrow$ in $\mathcal{M}$, into $\rightarrow$.
3. The remaining circle marks in $\mathcal{M}$ are on $\circ\!-\!\circ$ edges. For each possible orientation
     of $C(\mathcal{M})$ as a DAG with no new v-structures, add the resulting graph to $[\mathcal{P}]$.
4. Let $L$ be a list of circle mark locations in $\mathcal{P}$.
5. for each $\mathcal{M}_k \in [\mathcal{P}]$
6.     for $l = 1$ to the length of $L$
7.         for each sequence of circle marks in $L$ of length $l$
8.             for each circle mark location in the sequence which is a tail in $\mathcal{M}_k$
               (i.e., $X_i \rightarrow X_j$ in $\mathcal{M}_k$ but $X_i \circ\!\!\rightarrow X_j$ or $X_i \circ\!-\!\circ X_j$ in $\mathcal{P}$)
9.                 Transform $X_i \rightarrow X_j$ in $\mathcal{M}_k$ to $X_i \leftrightarrow X_j$ if the conditions in Zhang
                   and Spirtes (2005: Lemma 1) are satisfied.
10.            end
11.        Add the resulting graph to $[\mathcal{P}]$. (Unless it is a duplicate.)
12.        end
13.    end
14. end
```

The graphical object after step 2 in the algorithm is what Zhang (2006) calls the Arrowhead Augmented Graph (AAG). Constructing an AAG from $\mathcal{P}$ and then orienting the circle component as any DAG (with no new v-structures) yields a MAG in the equivalence class of $\mathcal{P}$; see Zhang (2006: Lemma 4.3.6).[6] So, if we enumerate all possible DAG orientations over the circle component of the graph we produce several MAGs in the equivalence class. The last step generates graphs with arrowheads in place of tail marks where there were circle marks in the original PAG. It invokes a rule for transforming $X_i \rightarrow X_j$ into $X_i \leftrightarrow X_j$ while preserving Markov equivalence. The rule is reproduced in Lemma A.1. Note that a path π between D and C, $\pi = \langle D, ..., A, B, C\rangle$, is a discriminating path if and only if: 1) π includes at least three edges; 2) B is a non-endpoint vertex on π, and is adjacent to C on π; and 3) D is not adjacent to C, and every vertex between D and B is a collider on π and is a parent of C.

Lemma A.1 *Let $\mathcal{M}$ be an arbitrary DMAG, and $A \rightarrow B$ an arbitrary directed edge in $\mathcal{M}$. Let $\mathcal{M}'$ be the graph identical to $\mathcal{M}$ except that the edge between A and B is $A \leftrightarrow B$. (In other words, $\mathcal{M}'$*

[6] That the circle component can be oriented into a DAG with no v-structures follows from the fact that the circle component is chordal. See Zhang (2006) for a proof and related references. Also note that we have assumed no selection variables, so contra the general definition of an AAG, there are no $\circ-$ edges to orient.

is the result of simply changing $A \to B$ into $A \leftrightarrow B$ in $\mathcal{M}$.) $\mathcal{M}'$ is a DMAG and Markov equivalent to $\mathcal{M}$ if and only if
(i) there is no directed path from A to B other than $A \to B$ in $\mathcal{M}$;
(ii) for any $C \to A$ in $\mathcal{M}$, $C \to B$ is also in $\mathcal{M}$; and for any $D \leftrightarrow A$ in $\mathcal{M}$, either $D \to B$ or $D \leftrightarrow B$ is in $\mathcal{M}$;
(iii) there is no discriminating path for A on which B is the endpoint adjacent to A in $\mathcal{M}$.

Proof. See Zhang and Spirtes (2005: Lemma 1).

In order to prove Theorem 4.2 we need several more lemmas. Let $\mathcal{P}$ be a PAG produced by the FCI algorithm, and $[\mathcal{P}]$ is the set of MAGs represented by $\mathcal{P}$. $\mathcal{P}^*$ is the subgraph of $\mathcal{P}$ over the vertices in $possibleDe(X_i, \mathcal{P}) \cup pds(X_i, Y, \mathcal{P})$ for some (X_i, Y). Let $[\mathcal{P}^*]$ be the set of graphs generated from $\mathcal{P}^*$ by the ZML algorithm. $C(\mathcal{P}^*)$ is the circle component of $\mathcal{P}^*$. $C(\mathcal{P})$ is chordal, meaning that any cycle of length 4 or more in $\mathcal{P}$ has an edge (chord) connecting two non-adjacent vertices on the cycle. A subgraph of a chordal graph is also chordal so $C(\mathcal{P}^*)$ is also chordal.

Lemma A.2 *The set $possibleDe(X_i, \mathcal{P}) \cup pds(X_i, Y, \mathcal{P})$ is sufficient for determining the generalized backdoor set for (X_i, Y) in every $\mathcal{M} \in [\mathcal{P}]$.*

Proof. First, we note that the subgraph over $pds(X_i, Y, \mathcal{P})$ is sufficient to construct $\mathcal{M}_{\underline{X_i}}$. To construct this graph we need to know which directed edges (if any) out of X_i are visible. A directed edge is from X_i to Y is visible if (i) there exists a vertex X_j such that $X_j \to X_i$ but X_j is not adjacent to Y or (ii) there exists a vertex X_j such that there is a collider path between X_j and X where every non-endpoint vertex is a parent of Y. The set $adj(X_i, \mathcal{P})$ is a subset of $pds(X_i, Y, \mathcal{P})$ so $pds(X_i, Y, \mathcal{P})$ suffices to determine condition (i). $pds(X_i, Y, \mathcal{P})$ also suffices to determine condition (ii) because it includes every vertex on a possible collider path from X_i. $pds(X_i, Y, \mathcal{P})$ is sufficient for checking whether $Y \in adj(X_i, \mathcal{M}_{\underline{X_i}})$, since it is sufficient for constructing $\mathcal{M}_{\underline{X_i}}$ and includes all the adjacencies of X_i. $pds(X_i, Y, \mathcal{P})$ is also sufficient for determining D-SEP$(X_i, Y, \mathcal{M}_{\underline{X_i}})$ by construction. Finally, $possibleDe(X_i, \mathcal{P})$ is sufficient for determining $De(X_i, \mathcal{M}_{\underline{X_i}})$, since any descendent of X_i in one of the MAGs represented by $\mathcal{P}$ is a possible descendent of X_i in $\mathcal{P}$. □

Lemma A.3 *Any DAG orientation of $C(\mathcal{P}^*)$ with no unshielded colliders is a subgraph of some DAG orientation of $C(\mathcal{P})$ with no unshielded colliders, as long as $C(\mathcal{P}^*)$ is connected.*

Proof. Let $C(\mathcal{P}^*)_{DAG}$ denote a DAG orientation (with no unsheilded collider) of $C(\mathcal{P}^*)$, and $C(\mathcal{P})_{DAG}$ is a DAG orientation of $C(\mathcal{P})$ which is includes $C(\mathcal{P}^*)_{DAG}$ as a subgraph. If the Lemma is false, then in $C(\mathcal{P})_{DAG}$ there must be a forced unshielded collider in order to preserve consistency with $C(\mathcal{P}^*)_{DAG}$. We will show that this implies a contradiction.

Let B be a vertex in $C(\mathcal{P})$ which is forced to be an unshielded collider in $C(\mathcal{P})_{DAG}$. Let A and C be the two non-adjacent vertices which collide at B. Note that least one of A, B, or C must not be in $C(\mathcal{P}^*)$ or else the triple would have been oriented in $C(\mathcal{P}^*)_{DAG}$. There must be a vertex D in $C(\mathcal{P})$ which is not adjacent to B and which is oriented as a parent of A by $C(\mathcal{P}^*)_{DAG}$ in order to force the orientation $A \to B$ in $C(\mathcal{P})_{DAG}$. Similarly, there must be a vertex E in $C(\mathcal{P})$ which is not adjecent to B and which is oriented as a parent of C by $C(\mathcal{P}^*)_{DAG}$ in order to force the orientation of $C \to B$ in $C(\mathcal{P})_{DAG}$. Without loss of generality, assume D and E are in $C(\mathcal{P}^*)$. (If they are not, we can find vertices F and G in $C(\mathcal{P}^*)$ which are connected to D and E by a sequence of $\circ\!-\!\circ$ edges, and so force the orientations $A \to B$ and $C \to B$. In this case we just repeat the argument that follows but for F and G.) There are two cases: either $D = E$ or not.

Case 1. $D = E$. This implies $D\circ\!-\!\circ A\circ\!-\!\circ B\circ\!-\!\circ C\circ\!-\!\circ D$ is in $C(\mathcal{P})$. This is a cycle of length 4, and so there must be a chord connecting two non-adjacent vertices since $C(\mathcal{P})$ is chordal. Either the chord is $A\circ\!-\!\circ C$ or $D\circ\!-\!\circ B$. The first contradicts our assumption A and C are not adjacent (and thus form part of an unshielded collider); the second contradicts our assumption that $A \to B$ is a forced orientation, since now it could have been oriented $D \to A \leftarrow B$.

Case 2. $D \neq E$. Then there is a path between D and E in $C(\mathcal{P}^*)$ by the connectedness of $C(\mathcal{P}^*)$. The path could be a single edge between D and E or it could be a longer path which includes other vertices in $C(\mathcal{P}^*)$. Either way $D\circ\!-\!\circ A\circ\!-\!\circ B\circ\!-\!\circ C\circ\!-\!\circ E\circ...\circ D$ is a cycle of length greater than 4. So it must have a chord. The chord cannot be between A and C because they form part of an unshielded collider. No matter how long the cycle is, there will be a chord between D and B or between E and B (to see this, do an induction on path lengths). But then either the orientation $A \to B$ or $C \to B$ is not forced, in contradiction to our assumption. □

Note that Lemma A.3 assumes that $C(\mathcal{P}^*)$ is connected. This is not generally the case. When $C(\mathcal{P}^*)$ is not connected, the graphical structure could be arranged such that some DAG orientation of $C(\mathcal{P}^*)$ is not a subgraph of any DAG orientation of $C(\mathcal{P})$. This can actually only happen under somewhat contrived circumstances; although one can construct a theoretical example, it has never come up in any of our simulations of "random" graphs. In any case, we can protect against this failure by adding two lines to the ZML algorithm (only when LV-IDA is run in "local" mode). After step 3, check whether $C(\mathcal{P}^*)$ is connected. If it is, proceed as usual. If it is not, check whether each DAG orientation of $C(\mathcal{P}^*)$ is extendable to a full DAG orientation of $C(\mathcal{P})$ using the algorithm of Dor and Tarsi (1992). This is a basically a check whether a partially oriented graph – $C(\mathcal{P})$ with induced subgraph $C(\mathcal{P}^*)_{DAG}$ – is consistent with any DAG orientation. Throw out any orientations of $C(\mathcal{P}^*)$ which are not extendable and keep those which are extendable. With this adjustment, the "local" ZML is guaranteed to produce only those orientations of $C(\mathcal{P}^*)$ which are consistent with orientations of $C(\mathcal{P})$.

Lemma A.4 *Every $\mathcal{M}^* \in [\mathcal{P}^*]$ is a subgraph of some $\mathcal{M} \in [\mathcal{P}]$, that is, listing the graphs represented by $\mathcal{P}^*$ does not produce any graphs which are not subgraphs of some MAG in the equivalence class of $\mathcal{P}$.*

Proof. We proceed by showing that every step in the ZML algorithm preserves the truth of the proposition, i.e., that no step of the procedure results in a graph in $[\mathcal{P}^*]$ which is not a subgraph of some graph in $[\mathcal{P}]$. Step 2 clearly preserves the truth of the proposition because the $\circ\!\!\to$ edges in $\mathcal{P}^*$ are just a subset of the $\circ\!\!\to$ edges in $\mathcal{P}$. $C(\mathcal{P}^*)$ is a subgraph of $C(\mathcal{P})$ which is chordal. Any orientation of $C(\mathcal{P}^*)$ as a DAG with no unshielded colliders is a subgraph of some DAG orientation of $C(\mathcal{P})$ with no unshielded colliders (by Lemma A.3 and the text which immediately follows the proof) so step 3 of the algorithm preserves the truth of the proposition.

Step 9 could produce a graph which is not a subgraph of some member in $[\mathcal{P}]$ if some mark change was legal according to rules (i), (ii), and (iii) of Lemma A.1 in $\mathcal{M}^*$ but not legal for all $\mathcal{M} \in [\mathcal{P}]$. In other words, there must be some transformation from $A \to B$ to $A \leftrightarrow B$ which is legal in some $\mathcal{M}^*$ but not legal in any $\mathcal{M} \in [\mathcal{P}]$. There are three ways this could happen, corresponding to the three rules (i), (ii), and (iii). We derive a contradiction in each case.

Case 1. Suppose $A \to B$ is legally transformed into $A \leftrightarrow B$ in $\mathcal{M}^*$ but there is a directed path from A to B (aside from $A \to B$) in every $\mathcal{M} \in [\mathcal{P}]$. $A \circ\!\!-\!\!* B$ must be in $\mathcal{P}^*$ for the transformation to be considered. ($*$ is a "wildcard" edge mark which can represent a circle, tail, or arrowhead.) Then $A \circ\!-\!* B$ is also in $\mathcal{P}$. But if there is a directed path from A to B in every $\mathcal{M} \in [\mathcal{P}]$, then A is an ancestor of B in $\mathcal{P}$ (by the completeness of FCI) and there cannot be a circle at A from B in $\mathcal{P}$. Contradiction.

Case 2. Suppose $A \rightarrow B$ is transformed into $A \leftrightarrow B$ in $\mathcal{M}^*$ but rule (ii) is not satisfied by any $\mathcal{M} \in [\mathcal{P}]$. There are two possibilities: (a) for all $\mathcal{M} \in [\mathcal{P}]$ with $C \rightarrow A$, C is adjacent to B but not $C \rightarrow B$; or (b) for all $\mathcal{M} \in [\mathcal{P}]$ with $D \leftrightarrow A$, D is adjacent to B but neither $D \rightarrow B$ nor $D \leftrightarrow B$. (Note that $\mathcal{M}^*$ and $\mathcal{M}$ have all the same adjacencies.) Suppose (a). Then $C \leftarrow B$ or $C \leftrightarrow B$ for all $\mathcal{M} \in [\mathcal{P}]$. Either way, all $\mathcal{M}$ are not ancestral (a directed cycle in the first case and an almost directed cycle in the second case). Suppose (b). Then $C \leftarrow B$ and all $\mathcal{M}$ are not ancestral (an almost directed cycle). Contradiction.

Case 3. Suppose $A \rightarrow B$ is legally transformed into $A \leftrightarrow B$ in $\mathcal{M}^*$ but there is a discriminating path for A on which B is the endpoint adjacent to A in every $\mathcal{M} \in [\mathcal{P}]$. Again, $A \circ\!-\!* B$ must be in $\mathcal{P}^*$ for the transformation to be considered and then $A\circ\!-\!*B$ is also in $\mathcal{P}$. If the discriminating path exists in every $\mathcal{M} \in [\mathcal{P}]$, then it exists in $\mathcal{P}$. But then the rule $\mathcal{R}4$ in FCI would have oriented $A \circ\!-\!* B$ as either $A \rightarrow B$ or $A \leftrightarrow B$ (see Zhang, 2008b). Contradiction.

So, no mark change would have occured in step 9 that would result in a graph which is not a subgraph of any graph in $[\mathcal{P}]$. □

Lemma A.5 *Every $\mathcal{M} \in [\mathcal{P}]$ is a supergraph of some $\mathcal{M}^* \in [\mathcal{P}^*]$, that is, listing the graphs represented by $\mathcal{P}^*$ produces all possible orientations of circle marks in $\mathcal{P}$, when the set of circle marks is restricted to the ones at vertices in $\mathcal{P}^*$.*

Proof. This follows from inspection of the ZML algorithm. ZML exhaustively orients all circle marks in $\mathcal{P}^*$ as tails and arrowheads, only excluding those arrowhead orientations which are not consistent with the conditions (i), (ii), and (iii) in Lemma A.1. But if an arrowhead orientation over the vertices in $\mathcal{P}$ is illegal by one of these rules, then the same orientation would be illegal in the vertices over $\mathcal{P}^*$. □

Theorem 4.2 follows from Lemmas A.2, A.4, and A.5. Lemma A.2 says that the set we've picked out, $\mathbf{Z_i}$, is sufficient for calculating the back-door set in each MAG. Lemma A.4 says we do not introduce any new orientations among the variables in $\mathbf{Z_i}$ which are not constituent of some MAG represented by $\mathcal{P}$, and Lemma A.5 says that we get leave out any possible orientations among the variables in $\mathbf{Z_i}$ which are constituent of some MAG represented by $\mathcal{P}$.

References

R. A. Ali, T. S. Richardson, and P. Spirtes. Markov equivalence for ancestral graphs. *The Annals of Statistics*, 37(5B):2808–2837, 2009.

D. Colombo and M. H. Maathuis. Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research*, 15(1):3741–3782, 2014.

D. Colombo, M. H. Maathuis, M. Kalisch, and T. S. Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, 40(1):294–321, 2012.

D. Dor and M. Tarsi. A simple algorithm to construct a consistent extension of a partially oriented graph. Technical Report R-185, Cognitive Systems Laboratory, UCLA, 1992.

D. Entner and P. O. Hoyer. Discovering unconfounded causal relationships using linear non-Gaussian models. In *New Frontiers in Artificial Intelligence*, pages 181–195. Springer, 2010.

D. Entner, P. Hoyer, and P. Spirtes. Data-driven covariate selection for nonparametric estimation of causal effects. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, pages 256–264, 2013.

R. Henao and O. Winther. Sparse linear identifiable multivariate modeling. *Journal of Machine Learning Research*, 12:863–905, 2011.

P. O. Hoyer, S. Shimizu, A. J. Kerminen, and M. Palviainen. Estimation of causal effects using linear non-Gaussian causal models with hidden variables. *International Journal of Approximate Reasoning*, 49(2):362–378, 2008.

A. Hyttinen, F. Eberhardt, and M. Järvisalo. Do-calculus when the true graph is unknown. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pages 395–404. AUAI Press, 2015.

M. Kalisch, M. Mächler, D. Colombo, M. H. Maathuis, and P. Bühlmann. Causal inference using graphical models with the `R` package `pcalg`. *Journal of Statistical Software*, 47(11):1–26, 2012.

Y. Kawahara, K. Bollen, S. Shimizu, and T. Washio. GroupLiNGAM: linear non-Gaussian acyclic models for sets of variables. *arXiv preprint arXiv:1006.5041*, 2010.

M. H. Maathuis and D. Colombo. A generalized back-door criterion. *The Annals of Statistics*, 43 (3):1060–1088, 2015.

M. H. Maathuis, M. Kalisch, and P. Bühlmann. Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics*, 37(6A):3133–3164, 2009.

M. H. Maathuis, D. Colombo, M. Kalisch, and P. Bühlmann. Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7:247–248, 2010.

D. Malinsky and P. Spirtes. Estimating causal effects with ancestral graph Markov models. *Journal of Machine Learning Research: Workshop and Conference Proceedings (PGM 16)*, 52:299–309, 2016.

C. Meek. Causal inference and causal explanation with background knowledge. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 403–410. Morgan Kaufmann Publishers Inc., 1995.

J. M. Ogarrio, P. Spirtes, and J. D. Ramsey. A hybrid causal search algorithm for latent variable models. *Journal of Machine Learning Research: Workshop and Conference Proceedings (PGM 16)*, 52:368–379, 2016.

J. Pearl. *Causality*. Cambridge University Press, 2009.

E. Perković, J. Textor, M. Kalisch, and M. H. Maathuis. A complete adjustment criterion. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pages 682–691. AUAI Press, 2015.

E. Perković, J. Textor, M. Kalisch, and M. H. Maathuis. Complete graphical characterization and construction of adjustment sets in Markov equivalence classes of ancestral graphs. *arXiv preprint arXiv:1606.06903*, 2016.

J. D. Ramsey. A scalable conditional independence test for nonlinear, non-Gaussian data. *arXiv preprint arXiv:1401.5031*, 2014.

T. Richardson and P. Spirtes. Ancestral graph Markov models. *The Annals of Statistics*, 30(4): 962–1030, 2002.

S. Shimizu and K. Bollen. Bayesian estimation of causal direction in acyclic structural equation models with individual-specific confounder variables and non-Gaussian distributions. *Journal of Machine Learning Research*, 15(1):2629–2652, 2014.

I. Shpitser and J. Pearl. Identification of joint interventional distributions in recursive semi-Markovian causal models. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, pages 1219–1226, 2006.

P. Spirtes, C. Meek, and T. Richardson. Causal inference in the presence of latent variables and selection bias. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 499–506. Morgan Kaufmann Publishers Inc., 1995.

P. Spirtes, C. N. Glymour, and R. Scheines. *Causation, prediction, and search.* MIT Press, 2000.

D. J. Stekhoven, I. Moraes, G. Sveinbjörnsson, L. Hennig, M. H. Maathuis, and P. Bühlmann. Causal stability ranking. *Bioinformatics*, 28(21):2819–2823, 2012.

T. Tashiro, S. Shimizu, A. Hyvärinen, and T. Washio. ParceLiNGAM: a causal ordering method robust against latent confounders. *Neural Computation*, 26(1):57–83, 2014.

J. Tian and J. Pearl. On the testable implications of causal models with hidden variables. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 519–527. Morgan Kaufmann Publishers Inc., 2002.

T. S. Verma and J. Pearl. Equivalence and synthesis of causal models. In *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*, pages 220–227. Elsevier, 1991.

J. Zhang. *Causal inference and reasoning in causally insufficient systems*. PhD thesis, Carnegie Mellon University, 2006.

J. Zhang. Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, 9:1437–1474, 2008a.

J. Zhang. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172(16):1873–1896, 2008b.

J. Zhang and P. Spirtes. A transformational characterization of Markov equivalence classes for directed acyclic graphs with latent variables. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, pages 667–674. AUAI Press, 2005.

K. Zhang, J. Peters, and B. Schölkopf. Kernel-based conditional independence test and application in causal discovery. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 804–813. AUAI Press, 2011.