Alternative Markov and Causal Properties for Acyclic Directed Mixed Graphs

Abstract

We extend Andersson-Madigan-Perlman chain graphs by (i) relaxing the semidirected acyclity constraint so that only directed cycles are forbidden, and (ii) allowing up to two edges between any pair of nodes. We introduce global, and ordered local and pairwise Markov properties for the new models. We show the equivalence of these properties for strictly positive probability distributions. We also show that when the random variables are continuous, the new models can be interpreted as systems of structural equations with correlated errors. This enables us to adapt Pearl's *do*-calculus to them. Finally, we describe an exact algorithm for learning the new models from observational and interventional data via answer set programming.

1 INTRODUCTION

Chain graphs (CGs) are graphs with possibly directed and undirected edges but without semidirected cycles. They have been extensively studied as a formalism to represent probabilistic independence models, because they can model symmetric and asymmetric relationships between random variables. Moreover, they are much more expressive than directed acyclic graphs (DAGs) and undirected graphs (UGs) (Sonntag and Peña, 2016). There are three different interpretations of CGs as independence models: The Lauritzen-Wermuth-Frydenberg (LWF) interpretation (Lauritzen, 1996), the multivariate regression (MVR) interpretation (Cox and Wermuth, 1996), and the Andersson-Madigan-Perlman (AMP) interpretation (Andersson et al., 2001). No interpretation subsumes another (Andersson et al., 2001; Sonntag and Peña, 2015). Moreover, AMP and MVR CGs are coherent with data generation by block-recursive normal linear regressions (Andersson et al., 2001).

Richardson (2003) extends MVR CGs by (i) relaxing the semidirected acyclity constraint so that only directed cycles are forbidden, and (ii) allowing up to two edges between any pair of nodes. The resulting models are called acyclic directed mixed graphs (ADMGs). These are the models in which Pearl's *do*-calculus operates to determine if the causal effect of an intervention is identifiable from observed quantities (Pearl, 2009). In this paper, we make the same two extensions to AMP CGs. We call our ADMGs alternative as opposed to the ones proposed by Richardson, which we call original. It is worth mentioning that neither the original ADMGs nor any other family of mixed graphical models that we know of (e.g. summary graphs (Cox and Wermuth, 1996), ancestral graphs (Richardson and Spirtes, 2002), MC graphs (Koster, 2002) or loopless mixed graphs (Sadeghi and Lauritzen, 2014)) subsume AMP CGs and hence our alternative ADMGs. To see it, we refer the reader to the works by Richardson and Spirtes (2002, p. 1025) and Sadeghi and Lauritzen (2014, Section 4.1). Therefore, our work complements the existing works.

The rest of the paper is organized as follows. Section 2 introduces some preliminaries. Sections 3 and 4 introduce global, and ordered local and pairwise Markov properties for our ADMGs, and prove their equivalence. When the random variables are continuous, Section 5 offers an intuitive interpretation of our ADMGs as systems of structural equations with correlated errors, so that Pearl's *do*-calculus can easily be adapted to them. Section 6 describes an exact algorithm for learning our ADMGs from observational and interventional data via answer set programming (Gelfond, 1988; Niemelä, 1999; Simons et al., 2002). We close the paper with some discussion in Section 7. Formal proofs of the claims made in this paper can be found in the supplementary material.

2 PRELIMINARIES

In this section, we introduce some concepts about graphical models. Unless otherwise stated, all the graphs and probability distributions in this paper are defined over a finite set

V. The elements of V are not distinguished from singletons. An ADMG G is a graph with possibly directed and undirected edges but without directed cycles. There may be up to two edges between any pair of nodes, but in that case the edges must be different and one of them must be undirected to avoid directed cycles. Edges between a node and itself are not allowed. See Figure 1 for two examples of ADMGs.

Given an ADMG G, we represent with $A \multimap B$ that $A \to B$ or $A - B$ (or both) is in G. The parents of $X \subseteq V$ in G are $Pa_G(X) = \{A | A \to B$ is in G with $B \in X\}$. The children of X in G are $Ch_G(X) = \{A | A \leftarrow B$ is in G with $B \in X\}$. The neighbours of X in G are $Ne_G(X) = \{A | A - B$ is in G with $B \in X\}$. The ancestors of X in G are $An_G(X) = \{A | A \to \ldots \to B$ is in G with $B \in X$ or $A \in X\}$. The descendants of X in G are $De_G(X) = \{A | A \leftarrow \ldots \leftarrow B$ is in G with $B \in X$ or $A \in X\}$. The semidescendants of X in G are $de_G(X) = \{A | A \circ\!\!- \ldots \circ\!\!- B$ is in G with $B \in X$ or $A \in X\}$. The non-semidescendants of X in G are $Nd_G(X) = V \setminus de_G(X)$. The connectivity components of X in G is $Cc_G(X) = \{A | A - \ldots - B$ is in G with $B \in X$ or $A \in X\}$. The connectivity components in G are denoted as $Cc(G)$. A route between a node V_1 and a node V_n on G is a sequence of (not necessarily distinct) nodes $V_1, \ldots, V_n$ such that V_i and V_{i+1} are adjacent in G for all $1 \leq i < n$. We do not distinguish between the sequences $V_1, \ldots, V_n$ and $V_n, \ldots, V_1$, i.e. they represent the same route. If the nodes in the route are all distinct, then the route is called a path. Finally, the subgraph of G induced by $X \subseteq V$, denoted as G_X, is the graph over X that has all and only the edges in G whose both ends are in X.

Let X, Y, W and Z be disjoint subsets of V. We represent by $X \perp_p Y | Z$ that X and Y are conditionally independent given Z in a probability distribution p. Every probability distribution p satisfies the following four properties: Symmetry $X \perp_p Y | Z \Rightarrow Y \perp_p X | Z$, decomposition $X \perp_p Y \cup W | Z \Rightarrow X \perp_p Y | Z$, weak union $X \perp_p Y \cup W | Z \Rightarrow X \perp_p Y | Z \cup W$, and contraction $X \perp_p Y | Z \cup W \wedge X \perp_p W | Z \Rightarrow X \perp_p Y \cup W | Z$. If p is strictly positive, then it also satisfies the intersection property $X \perp_p Y | Z \cup W \wedge X \perp_p W | Z \cup Y \Rightarrow X \perp_p Y \cup W | Z$. Some (not yet characterized) probability distributions also satisfy the composition property $X \perp_p Y | Z \wedge X \perp_p W | Z \Rightarrow X \perp_p Y \cup W | Z$.

3 GLOBAL MARKOV PROPERTY

In this section, we introduce four separation criteria for ADMGs. Moreover, we show that they are all equivalent. A probability distribution is said to satisfy the global Markov property with respect to an ADMG if every separation in the graph can be interpreted as an independence in the distribution.

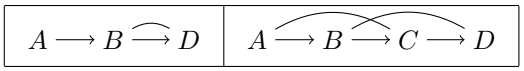

Figure 1: Examples of ADMGs.

Criterion 1. A node C on a path in an ADMG G is said to be a collider on the path if $A \to C \circ\!\!- B$ is a subpath. Moreover, the path is said to be connecting given $Z \subseteq V$ when

- every collider on the path is in $An_G(Z)$, and
- every non-collider C on the path is outside Z unless $A - C - B$ is a subpath and $Pa_G(C) \setminus Z \neq \emptyset$.

Let X, Y and Z denote three disjoint subsets of V. When there is no path in G connecting a node in X and a node in Y given Z, we say that X is separated from Y given Z in G and denote it as $X \perp_G Y | Z$.

Criterion 2. A node C on a route in an ADMG G is said to be a collider on the route if $A \to C \circ\!\!- B$ is a subroute. Note that maybe $A = B$. Moreover, the route is said to be connecting given $Z \subseteq V$ when

- every collider on the route is in Z, and
- every non-collider C on the route is outside Z.

Let X, Y and Z denote three disjoint subsets of V. When there is no route in G connecting a node in X and a node in Y given Z, we say that X is separated from Y given Z in G and denote it as $X \perp_G Y | Z$.

Criterion 3. Let G^u denote the UG over V that contains all and only the undirected edges in G. The extended subgraph $G[X]$ with $X \subseteq V$ is defined as $G[X] = G_{An_G(X)} \cup (G^u)_{Cc_G(An_G(X))}$. Two nodes A and B in G are said to be collider connected if there is a path between them such that every non-endpoint node is a collider, i.e. $A \to C \circ\!\!- B$ or $A \to C - D \leftarrow B$. Such a path is called a collider path. Note that a single edge forms a collider path. The augmented graph G^a is the UG over V such that $A - B$ is in G^a if and only if A and B are collider connected in G. The edge $A - B$ is called augmented if it is in G^a but A and B are not adjacent in G. A path in G^a is said to be connecting given $Z \subseteq V$ if no node on the path is in Z. Let X, Y and Z denote three disjoint subsets of V. When there is no path in $G[X \cup Y \cup Z]^a$ connecting a node in X and a node in Y given Z, we say that X is separated from Y given Z in G and denote it as $X \perp_G Y | Z$.

Criterion 4. Given an UG H over V and $X \subseteq V$, we define the marginal graph H^X as the UG over X such that $A - B$ is in H^X if and only if $A - B$ is in H or $A - V_1 - \ldots - V_n - B$ is H with $V_1, \ldots, V_n \notin X$. We define the marginal extended subgraph $G[X]^m$ as $G[X]^m =$

$G_{An_G(X)} \cup ((G^u)_{Cc_G(An_G(X))})^{An_G(X)}$. Let X, Y and Z denote three disjoint subsets of V. When there is no path in $(G[X \cup Y \cup Z]^m)^a$ connecting a node in X and a node in Y given Z, we say that X is separated from Y given Z in G and denote it as $X \perp_G Y|Z$.

The first three separation criteria introduced above coincide with those introduced by Andersson et al. (2001) and Levitz et al. (2001) for AMP CGs. The equivalence for AMP CGs of these three separation criteria has been proven by Levitz et al. (2001, Theorem 4.1). The following theorems prove the equivalence for ADMGs of the four separation criteria introduced above.

Theorem 1 *There is a path in an ADMG G connecting a node in X and a node in Y given Z if and only if there is a path in $G[X \cup Y \cup Z]^a$ connecting a node in X and a node in Y given Z.*

Theorem 2 *There is a path in an ADMG G connecting A and B given Z if and only if there is a route in G connecting A and B given Z.*

Theorem 3 *Given an ADMG G, there is a path in $G[X \cup Y \cup Z]^a$ connecting a node in X and a node in Y given Z if and only if there is a path in $(G[X \cup Y \cup Z]^m)^a$ connecting a node in X and a node in Y given Z.*

Unlike in AMP CGs, two non-adjacent nodes in an ADMG are not necessarily separated. For example, $A \perp_G D|Z$ does not hold for any Z in the ADMGs in Figure 1. This drawback is shared by the original ADMGs (Evans and Richardson, 2013, p. 752), summary graphs and MC graphs (Richardson and Spirtes, 2002, p. 1023), and ancestral graphs (Richardson and Spirtes, 2002, Section 3.7). For ancestral graphs, the problem can be solved by adding edges to the graph without altering the separations represented until every missing edge corresponds to a separation (Richardson and Spirtes, 2002, Section 5.1). A similar solution does not exist for our ADMGs (we omit the details).

4 ORDERED LOCAL AND PAIRWISE MARKOV PROPERTIES

In this section, we introduce ordered local and pairwise Markov properties for ADMGs. Given an ADMG G, the directed acyclity of G implies that we can specify a total ordering ($\prec$) of the nodes of G such that $A \prec B$ only if $B \notin An_G(A)$. Such an ordering is said to be consistent with G. Let the predecessors of A with respect to $\prec$ be defined as $Pre_G(A, \prec) = \{B|B \prec A \text{ or } B = A\}$. Given $S \subseteq V$, we define the Markov blanket of $B \in S$ with respect to $G[S]$ as $Mb_{G[S]}(B) = Ch_{G[S]}(B) \cup Ne_{G[S]}(B \cup Ch_{G[S]}(B)) \cup Pa_{G[S]}(B \cup Ch_{G[S]}(B) \cup Ne_{G[S]}(B \cup Ch_{G[S]}(B)))$. We say that a probability distribution p satisfies the ordered local Markov property with respect to G and $\prec$ if for any $A \in V$ and $S \subseteq Pre_G(A, \prec)$ such that $A \in S$

$$B \perp_p S \setminus (B \cup Mb_{G[S]}(B))|Mb_{G[S]}(B)$$

for all $B \in S$.

Theorem 4 *Given a probability distribution p satisfying the intersection property, p satisfies the global Markov property with respect to an ADMG if and only if it satisfies the ordered local Markov property with respect to the ADMG and a consistent ordering of its nodes.*

Similarly, we say that a probability distribution p satisfies the ordered pairwise Markov property with respect to G and $\prec$ if for any $A \in V$ and $S \subseteq Pre_G(A, \prec)$ such that $A \in S$

$$B \perp_p C|V(G[S]) \setminus (B \cup C)$$

for all nodes $B, C \in S$ that are not adjacent in $G[S]^a$, and where $V(G[S])$ denotes the nodes in $G[S]$.

Theorem 5 *Given a probability distribution p satisfying the intersection property, p satisfies the global Markov property with respect to an ADMG if and only if it satisfies the ordered pairwise Markov property with respect to the ADMG and a consistent ordering of its nodes.*

For each $A \in V$ and $S \subseteq Pre_G(A, \prec)$ such that $A \in S$, the ordered local Markov property specifies an independence for each $B \in S$. The number of independences to specify can be reduced by noting that $G[S] = G[An_G(S)]$ and, thus, we do not need to consider every set $S \subseteq Pre_G(A, \prec)$ but only those that are ancestral, i.e. those such that $S = An_G(S)$. The number of independences to specify can be further reduced by considering only maximal ancestral sets, i.e. those sets S such that $Mb_{G[S]}(B) \subset Mb_{G[T]}(B)$ for every ancestral set T such that $S \subset T \subseteq Pre_G(A, \prec)$. The independences for the non-maximal ancestral sets follow from the independences for the maximal ancestral sets by decomposition. A characterization of the maximal ancestral sets is possible but notationally cumbersome (we omit the details). All in all, for each node and maximal ancestral set, the ordered local Markov property specifies an independence for each node in the set. This number is greater than for the original ADMGs, where a single independence is specified for each node and maximal ancestral set (Richardson, 2003, Section 3.1). Even fewer independences are needed for the original ADMGs when interpreted as linear causal models (Kang and Tian, 2009, Section 4). All in all, our ordered local Markov property serves its purpose, namely to identify a subset of the independences in the global Markov property that implies the rest.

Note that Andersson et al. (2001, Theorem 3) describe local and pairwise Markov properties for AMP CGs that are equivalent to the global one under the assumption of the intersection and composition properties. Our ordered local

and pairwise Markov properties above only require assuming the intersection property. Note that this assumption is also needed to prove similar results for much simpler models such as UGs (Lauritzen, 1996, Theorem 3.7). For AMP CGs, however, we can do better than just using the ordered local and pairwise Markov properties for ADMGs above. Specifically, we introduce in the next section neater local and pairwise Markov properties for AMP CGs under the intersection property assumption. Later on, we will also use them to prove some results for ADMGs.

4.1 LOCAL AND PAIRWISE MARKOV PROPERTIES FOR AMP CGS

Andersson et al. (2001, Theorem 2) prove that a probability distribution p satisfies the global Markov property with respect to an AMP CG G if and only if it satisfies the block-recursive Markov property which requires that the following three properties hold for all $C \in Cc(G)$:

- C1: $C \perp_p Nd_G(C) \setminus Cc_G(Pa_G(C)) | Cc_G(Pa_G(C))$.
- C2: $p(C|Cc_G(Pa_G(C)))$ satisfies the global Markov property with respect to G_C.
- C3*: $D \perp_p Cc_G(Pa_G(C)) \setminus Pa_G(D) | Pa_G(D)$ for all $D \subseteq C$.

We simplify the block-recursive Markov property as follows.

Theorem 6 *C1, C2 and C3* hold if and only if the following two properties hold:*

- *C1*: $D \perp_p Nd_G(D) \setminus Pa_G(D) | Pa_G(D)$ for all $D \subseteq C$.*
- *C2*: $p(C|Pa_G(C))$ satisfies the global Markov property with respect to G_C.*

Andersson et al. (2001, Theorem 3) also prove that a probability distribution p satisfying the intersection and composition properties satisfies the global Markov property with respect to an AMP CG G if and only if it satisfies the local Markov property which requires that the following two properties hold for all $C \in Cc(G)$:

- L1: $A \perp_p C \setminus (A \cup Ne_G(A)) | Nd_G(C) \cup Ne_G(A)$ for all $A \in C$.
- L2: $A \perp_p Nd_G(C) \setminus Pa_G(A) | Pa_G(A)$ for all $A \in C$.

We introduce below a local Markov property that is equivalent to the global one under the assumption of the intersection property only.

Theorem 7 *A probability distribution p satisfying the intersection property satisfies the global Markov property with respect to an AMP CG G if and only if the following two properties hold for all $C \in Cc(G)$:*

- *L1: $A \perp_p C \setminus (A \cup Ne_G(A)) | Nd_G(C) \cup Ne_G(A)$ for all $A \in C$.*
- *L2*: $A \perp_p Nd_G(C) \setminus Pa_G(A \cup S) | S \cup Pa_G(A \cup S)$ for all $A \in C$ and $S \subseteq C \setminus A$.*

Finally, Andersson et al. (2001, Theorem 3) also prove that a probability distribution p satisfying the intersection and composition properties satisfies the global Markov property with respect to an AMP CG G if and only if it satisfies the pairwise Markov property which requires that the following two properties hold for all $C \in Cc(G)$:

- P1: $A \perp_p B | Nd_G(C) \cup C \setminus (A \cup B)$ for all $A \in C$ and $B \in C \setminus (A \cup Ne_G(A))$.
- P2: $A \perp_p B | Nd_G(C) \setminus B$ for all $A \in C$ and $B \in Nd_G(C) \setminus Pa_G(A)$.

We introduce below a pairwise Markov property that is equivalent to the global one under the assumption of the intersection property only.

Theorem 8 *A probability distribution p satisfying the intersection property satisfies the global Markov property with respect to an AMP CG G if and only if the following two properties hold for all $C \in Cc(G)$:*

- *P1: $A \perp_p B | Nd_G(C) \cup C \setminus (A \cup B)$ for all $A \in C$ and $B \in C \setminus (A \cup Ne_G(A))$.*
- *P2*: $A \perp_p B | S \cup Nd_G(C) \setminus B$ for all $A \in C$, $S \subseteq C \setminus A$ and $B \in Nd_G(C) \setminus Pa_G(A \cup S)$.*

5 CAUSAL INTERPRETATION

Let us assume that V is normally distributed. In this section, we show that an ADMG G can be interpreted as a system of structural equations with correlated errors. Specifically, the system includes an equation for each $A \in V$, which is of the form $A = \beta_A \cdot Pa_G(A) + \epsilon_A$ where ϵ_A denotes the error term. The error terms are represented implicitly in G. They can be represented explicitly by magnifying G into the ADMG G' as follows:

1	Set $G' = G$
2	For each node A in G
3	Add the node ϵ_A and the edge $\epsilon_A \rightarrow A$ to G'
4	For each edge $A - B$ in G
5	Replace $A - B$ with the edge $\epsilon_A - \epsilon_B$ in G'

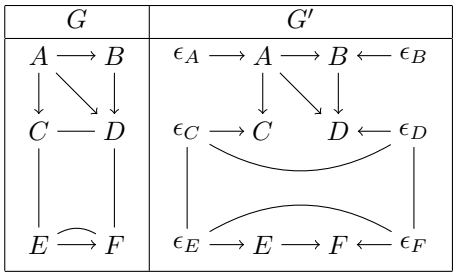

Figure 2: Example of the magnification of an ADMG.

The magnification above basically consists in adding the error nodes ϵ_A to G and connect them appropriately. Figure 2 shows an example. Note that every node $A \in V$ is determined by $Pa_{G'}(A)$ and that ϵ_A is determined by $A \cup Pa_{G'}(A) \setminus \epsilon_A$. Let ϵ denote all the error nodes in G'. Formally, we say that $A \in V \cup \epsilon$ is determined by $Z \subseteq V \cup \epsilon$ when $A \in Z$ or A is a function of Z. We use $Dt(Z)$ to denote all the nodes that are determined by Z. From the point of view of the separations, that a node outside the conditioning set of a separation is determined by the conditioning set has the same effect as if the node were actually in the conditioning set. Bearing this in mind, it is not difficult to see that, as desired, G and G' represent the same separations over V. The following theorem formalizes this result.

Theorem 9 *Let X, Y and Z denote three disjoint subsets of V. Then, $X \perp_G Y|Z$ if and only if $X \perp_{G'} Y|Z$.*

Finally, let $\epsilon \sim \mathcal{N}(0, \Lambda)$ such that $(\Lambda^{-1})_{\epsilon_A,\epsilon_B} = 0$ if $\epsilon_A - \epsilon_B$ is not in G'. Then, G can be interpreted as a system of structural equations with correlated errors as follows. For any $A \in V$

$$A = \sum_{B \in Pa_G(A)} \beta_{AB} B + \epsilon_A \tag{1}$$

and for any other $B \in V$

$$covariance(\epsilon_A, \epsilon_B) = \Lambda_{\epsilon_A,\epsilon_B}. \tag{2}$$

The following two theorems confirm that the interpretation above works as intended. A similar result to the second theorem exists for the original ADMGs (Koster, 1999, Theorem 1).

Theorem 10 *Every probability distribution $p(V)$ specified by Equations (1) and (2) is Gaussian.*

Theorem 11 *Every probability distribution $p(V)$ specified by Equations (1) and (2) satisfies the global Markov property with respect to G.*

The equations above specify each node as a linear function of its parents with additive normal noise. The equations can be generalized to nonlinear or nonparametric functions as long as the noise remains additive normal. That is, $A = f(Pa_G(A)) + \epsilon_A$ for all $A \in V$, with $\epsilon \sim \mathcal{N}(0, \Lambda)$ such that $(\Lambda^{-1})_{\epsilon_A,\epsilon_B} = 0$ if $\epsilon_A - \epsilon_B$ is not in G'. That the noise is additive normal ensures that ϵ_A is determined by $A \cup Pa_{G'}(A) \setminus \epsilon_A$, which is needed for Theorem 9 to remain valid which, in turn, is needed for Theorem 11 to remain valid.

A less formal but more intuitive alternative interpretation of ADMGs is as follows. We can interpret the parents of each node in an ADMG as its observed causes. Its unobserved causes are grouped into an error node that is represented implicitly in the ADMG. We can interpret the undirected edges in the ADMG as the correlation relationships between the different error nodes. The causal structure is constrained to be a DAG, but the correlation structure can be any UG. This causal interpretation of our ADMGs parallels that of the original ADMGs (Pearl, 2009). There are however two main differences. First, the noise in the original ADMGs is not necessarily additive normal. Second, the correlation structure of the error nodes in the original ADMGs is represented by a covariance graph, i.e. a graph with only bidirected edges (Pearl and Wermuth, 1993). Therefore, whereas a missing edge between two error nodes in the original ADMGs represents marginal independence, in our ADMGs it represents conditional independence given the rest of the error nodes. This means that the original and our ADMGs represent complementary causal models. Consequently, there are scenarios where the identification of the causal effect of an intervention is not possible with the original ADMGs but is possible with ours, and vice versa. We elaborate on this in the next section.

5.1 *do*-CALCULUS

We start by adapting Pearl's *do*-calculus, which operates on the original ADMGs, to our ADMGs. The original *do*-calculus consists of the following three rules, whose repeated application permits in some cases to identify (i.e. compute) the causal effect of an intervention from observed quantities:

- Rule 1 (insertion/deletion of observations):
  $p(Y|do(X), Z \cup W) = p(Y|do(X), W)$ if $Y \perp_{G''} Z|X \cup W||X$.
- Rule 2 (action/observation exchange):
  $p(Y|do(X), do(Z), W) = p(Y|do(X), Z \cup W)$ if $Y \perp_{G''} F_Z|X \cup W \cup Z||X$.
- Rule 3 (insertion/deletion of actions):
  $p(Y|do(X), do(Z), W) = p(Y|do(X), W)$ if $Y \perp_{G''} F_Z|X \cup W||X$.

where X, Y, Z and W are disjoint subsets of V, G'' is the original ADMG G augmented with an intervention random

variable F_A and an edge $F_A \rightarrow A$ for every $A \in V$, and "$||X$" denotes an intervention on X in G'', i.e. any edge with an arrowhead into any node in X is removed. See Pearl (1995, p. 686) for further details and the proof that the rules are sound. Fortunately, the rules also apply to our ADMGs by simply redefining "$||X$" appropriately. The proof that the rules are still sound is essentially the same as before. Specifically, "$||X$" should now be implemented as follows:

1	Delete from G'' all the edges $A \rightarrow B$ with $B \in X$
2	For each path $A - V_1 - \ldots - V_n - B$ in G'' with $A, B \notin X$ and $V_1, \ldots, V_n \in X$
3	Add the edge $A - B$ to G''
4	Delete from G'' all the edges $A - B$ with $B \in X$

Line 1 is shared with an intervention in an original ADMG. Lines 2-4 are best understood in terms of the magnified ADMG G': They correspond to marginalizing the error nodes associated to the nodes in X out of G'_ϵ, the UG that represents the correlation structure of the error nodes. In other words, they replace G'_ϵ with $(G')^{\epsilon \setminus \epsilon_X}$, the marginal graph of G'_ϵ over $\epsilon \setminus \epsilon_X$. This makes sense since ϵ_X is no longer associated to X due to the intervention and, thus, we may want to marginalize it out because it is unobserved. This is exactly what lines 2-4 imply. To see it, note that the ADMG after the intervention and the magnified ADMG after the intervention represent the same separations over V, by Theorem 9.

Now, we show that the original and our ADMGs allow for complementary causal reasoning. Specifically, we show an example where our ADMGs allow for the identification of the causal effect of an intervention whereas the original ADMGs do not, and vice versa. Consider the DAG in Figure 3, which represents the causal relationships among all the random variables in the domain at hand.[1] However, only A, B and C are observed. Moreover, U_S represents selection bias. Although other definitions may exist, we say that selection bias is present if two unobserved causes have a common effect that is omitted from the study but influences the selection of the samples in the study (Pearl, 2009, p. 163). Therefore, the corresponding unobserved causes are correlated in every sample selected. Note that this definition excludes the possibility of an intervention affecting the selection because, in a causal model, unobserved causes do not have observed causes. Note also that our goal is not the identification of the causal effect of an intervention in the whole population but in the subpopulation that satisfies the selection bias criterion.[2] For causal effect identification

[1]For instance, the DAG may correspond to the following fictitious domain: A = Smoking, B = Lung cancer, C = Drinking, U_A = Parents' smoking, U_B = Parents' lung cancer, U_C = Parents' drinking, U = Parents' genotype that causes smoking and drinking, U_S = Parents' hospitalization.

[2]For instance, in the fictitious domain in the previous footnote, we are interested in the causal effect that smoking may have on the development of lung cancer for the patients with hospitalized parents.

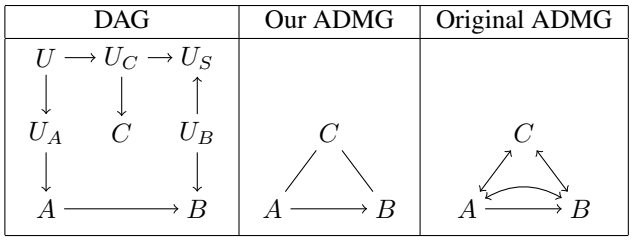

Figure 3: Example of a case where $p(B|do(A))$ is identifiable with our ADMG but not with the original one.

in the whole population, see Bareinboim and Tian (2015).

The ADMGs in Figure 3 represent the causal model represented by the DAG when only the observed random variables are modeled. According to our interpretation of ADMGs above, our ADMG is derived from the DAG by keeping the directed edges between observed random variables, and adding an undirected edge between two observed random variables if and only if their unobserved causes are not separated in the DAG given the unobserved causes of the rest of the observed random variables. In other words, $U_A \perp U_B | U_C$ holds in the DAG but $U_A \perp U_C | U_B$ and $U_B \perp U_C | U_A$ do not and, thus, the edges $A - C$ and $B - C$ are added to the ADMG but $A - B$ is not. Deriving the original ADMG is less straightforward. The bidirected edges in an original ADMG represent potential marginal dependence due to a common unobserved cause, also known as confounding. Thus, the original ADMGs are not meant to model selection bias. The best we can do is then to use bidirected edges to represent potential marginal dependences regardless of their origin. This implies that we can derive the original ADMG from the DAG by keeping the directed edges between observed random variables, and adding a bidirected edge between two observed random variables if and only if their unobserved causes are not separated in the DAG given the empty set. Clearly, $p(B|do(A))$ is not identifiable with the original ADMG but is identifiable with our ADMG (Pearl, 2009, p. 94). Specifically,

$$p(B|do(A)) = \sum_C p(B|do(A), C)p(C|do(A))$$

$$= \sum_C p(B|do(A), C)p(C) = \sum_C p(B|A, C)p(C)$$

where the first equality is due to marginalization, the second due to Rule 3, and the third due to Rule 2.

The original ADMGs assume that confounding is always the source of correlation between unobserved causes. In the example above, we consider selection bias as an additional source. However, this is not the only possibility. For instance, U_B and U_C may be tied by a physical law of the

form $f(U_B, U_C) = constant$ devoid of causal meaning, much like Boyle's law relates the pressure and volume of a gas as $pressure \cdot volume = constant$ if the temperature and amount of gas remain unchanged within a closed system. In such a case, the discussion above still applies and our ADMG allows for causal effect identification but the original does not. For an example where the original ADMGs allow for causal effect identification whereas ours do not, simply replace the subgraph $U_C \rightarrow U_S \leftarrow U_B$ in Figure 3 with $U_C \leftarrow W \rightarrow U_B$ where W is an unobserved random variable. Then, our ADMG will contain the same edges as before plus the edge $A - B$, making the causal effect non-identifiable. The original ADMG will contain the same edges as before with the exception of the edge $A \leftrightarrow B$, making the causal effect identifiable.

In summary, the bidirected edges of the original ADMGs have a clear semantics: They represent potential marginal dependence due to a common unobserved cause. This means that we have to know the causal relationships involving the unobserved random variables to derive the ADMG. Or at least, we have to know that there is no selection bias or tying laws so that marginal dependence can be attributed to a common unobserved cause due to Reichenbach's principle (Pearl, 2009, p. 30). This knowledge may not be available in some cases. Moreover, the original ADMGs are not meant to represent selection bias or tying laws. To solve these two problems, we may be willing to use the bidirected edges to represent potential marginal dependences regardless of their origin. Our ADMGs are somehow dual to the original ADMGs, since the undirected edges represent potential saturated conditional dependence between unobserved causes. This implies that in some cases, such as in the example above, our ADMGs may allow for causal effect identification whereas the original may not.

6 LEARNING VIA ASP

In this section, we introduce an exact algorithm for learning ADMGs via answer set programming (ASP), which is a declarative constraint satisfaction paradigm that is well-suited for solving computationally hard combinatorial problems (Gelfond, 1988; Niemelä, 1999; Simons et al., 2002). ASP represents constraints in terms of first-order logical rules. Therefore, when using ASP, the first task is to model the problem at hand in terms of rules so that the set of solutions implicitly represented by the rules corresponds to the solutions of the original problem. One or multiple solutions to the original problem can then be obtained by invoking an off-the-shelf ASP solver on the constraint declaration. Each rule in the constraint declaration is of the form `head :- body`. The head contains an atom, i.e. a fact. The body may contain several literals, i.e. negated and non-negated atoms. Intuitively, the rule is a justification to derive the head if the body is true. The body is true if its non-negated atoms can be derived, and its negated atoms cannot. A rule with only the head is an atom. A rule without the head is a hard-constraint, meaning that satisfying the body results in a contradiction. Soft-constraints are encoded as rules of the form `:~ body. [W]`, meaning that satisfying the body results in a penalty of W units. The ASP solver returns the solutions that meet the hard-constraints and minimize the total penalty due to the soft-constraints. In this work, we use the ASP solver `clingo` (Gebser et al., 2011), whose underlying algorithms are based on state-of-the-art Boolean satisfiability solving techniques (Biere et al., 2009).

Figure 4 shows the ASP encoding of our learning algorithm. The predicate `node(X)` in rule 1 represents that X is a node. The predicates `line(X,Y,I)` and `arrow(X,Y,I)` represent that there is an undirected and directed edge from X to Y after having intervened on the node I. The observational regime corresponds to $I = 0$. The rules 2-3 encode a non-deterministic guess of the edges for the observational regime, which means that the ASP solver with implicitly consider all possible graphs during the search, hence the exactness of the search. The edges under the observational regime are used in the rules 4-6 to define the edges in the graph after having intervened on I, following the description in Section 5.1. Therefore, the algorithm assumes continuous random variables and additive normal noise when the input contains interventions. It makes no assumption though when the input consists of just observations. The rules 7-8 enforce the fact that undirected edges are symmetric and that there is at most one directed edge between two nodes. The predicate `ancestor(X,Y)` represents that X is an ancestor of Y. The rules 9-11 enforce that the graph has no directed cycles. The predicates in the rules 12-13 represent whether a node X is or is not in a set of nodes C. The rules 14-25 encode the separation criterion 2 in Section 3. The predicate `con(X,Y,C,I)` in rules 26-29 represents that there is a connecting route between X and Y given C after having intervened on I. The rule 30 enforces that each dependence in the input must correspond to a connecting route. The rule 31 represents that each independence in the input that is not represented implies a penalty of W units. The rules 32-33 represent a penalty of 1 unit per edge. Other penalty rules can be added similarly.

Figure 6 shows the ASP encoding of all the (in)dependences in the probability distribution at hand, e.g. as determined by some available data. Specifically, the predicate `nodes(3)` represents that there are three nodes in the domain at hand, and the predicate `set(0..7)` represents that there are eight sets of nodes, indexed from 0 (empty set) to 7 (full set). The predicate `indep(X,Y,C,I,W)` (respectively `dep(X,Y,C,I,W)`) represents that the nodes X and Y are conditionally independent (respectively dependent)

```
% input predicates
% nodes(N): N is the number of nodes
% set(X): X is the index of a set of nodes
% dep(X,Y,C,I,W) (resp. indep(X,Y,C,I,W)): the nodes X and Y are dependent (resp.
%                                          independent) given the set of nodes C
%                                          after having intervened on the node I

% nodes
node(X) :- nodes(N), X=1..N.                                                    % rule 1

% edges
{ line(X,Y,0) } :- node(X), node(Y), X != Y.                                    %      2
{ arrow(X,Y,0) } :- node(X), node(Y), X != Y.                                   %      3
line(X,Y,I) :- line(X,Y,0), node(I), X != I, Y != I, I > 0.                     %      4
line(X,Y,I) :- line(X,I,0), line(I,Y,0), node(I), X != Y, I > 0.
arrow(X,Y,I) :- arrow(X,Y,0), node(I), Y != I, I > 0.                           %      6
line(X,Y,I) :- line(Y,X,I).                                                     %      7
:- arrow(X,Y,I), arrow(Y,X,I).                                                  %      8

% directed acyclity
ancestor(X,Y) :- arrow(X,Y,0).                                                  %      9
ancestor(X,Y) :- ancestor(X,Z), ancestor(Z,Y).
:- ancestor(X,Y), arrow(Y,X,0).                                                 %      11

% set membership
inside_set(X,C) :- node(X), set(C), 2**(X-1) & C != 0.                          %      12
outside_set(X,C) :- node(X), set(C), 2**(X-1) & C = 0.                          %      13

% end_line/head/tail(X,Y,C,I) means that there is a connecting route
% from X to Y given C that ends with an line/arrowhead/arrowtail

% single edge route
end_line(X,Y,C,I) :- line(X,Y,I), outside_set(X,C).                             %      14
end_head(X,Y,C,I) :- arrow(X,Y,I), outside_set(X,C).
end_tail(X,Y,C,I) :- arrow(Y,X,I), outside_set(X,C).

% connection through non-collider
end_line(X,Y,C,I) :- end_line(X,Z,C,I), line(Z,Y,I), outside_set(Z,C).
end_line(X,Y,C,I) :- end_tail(X,Z,C,I), line(Z,Y,I), outside_set(Z,C).
end_head(X,Y,C,I) :- end_line(X,Z,C,I), arrow(Z,Y,I), outside_set(Z,C).
end_head(X,Y,C,I) :- end_head(X,Z,C,I), arrow(Z,Y,I), outside_set(Z,C).
end_head(X,Y,C,I) :- end_tail(X,Z,C,I), arrow(Z,Y,I), outside_set(Z,C).
end_tail(X,Y,C,I) :- end_tail(X,Z,C,I), arrow(Y,Z,I), outside_set(Z,C).

% connection through collider
end_line(X,Y,C,I) :- end_head(X,Z,C,I), line(Z,Y,I), inside_set(Z,C).
end_tail(X,Y,C,I) :- end_line(X,Z,C,I), arrow(Y,Z,I), inside_set(Z,C).
end_tail(X,Y,C,I) :- end_head(X,Z,C,I), arrow(Y,Z,I), inside_set(Z,C).         %      25

% derived non-separations
con(X,Y,C,I) :- end_line(X,Y,C,I), X != Y, outside_set(Y,C).                    %      26
con(X,Y,C,I) :- end_head(X,Y,C,I), X != Y, outside_set(Y,C).
con(X,Y,C,I) :- end_tail(X,Y,C,I), X != Y, outside_set(Y,C).
con(X,Y,C,I) :- con(Y,X,C,I).                                                   %      29

% satisfy all dependences
:- dep(X,Y,C,I,W), not con(X,Y,C,I).                                            %      30

% maximize the number of satisfied independences
:~ indep(X,Y,C,I,W), con(X,Y,C,I). [W,X,Y,C,I]                                  %      31

% minimize the number of lines/arrows
:~ line(X,Y,0), X < Y. [1,X,Y,1]                                                %      32
:~ arrow(X,Y,0). [1,X,Y,2]                                                      %      33

% show results
#show. #show line(X,Y) : line(X,Y,0), X < Y. #show arrow(X,Y) : arrow(X,Y,0).
```

Figure 4: ASP encoding of the learning algorithm.

```
{ biarrow(X,Y,0) } :- node(X), node(Y), X != Y.
:- biarrow(X,Y,0), line(Z,W,0).
biarrow(X,Y,I) :- biarrow(X,Y,0), node(I), X != I, Y != I, I > 0.
biarrow(X,Y,I) :- biarrow(Y,X,I).

end_head(X,Y,C,I) :- biarrow(X,Y,I), outside_set(X,C).
end_head(X,Y,C,I) :- end_tail(X,Z,C,I), biarrow(Z,Y,I), outside_set(Z,C).
end_head(X,Y,C,I) :- end_head(X,Z,C,I), biarrow(Z,Y,I), inside_set(Z,C).

:~ biarrow(X,Y,0), X < Y. [1,X,Y,3]

#show biarrow(X,Y) : biarrow(X,Y,0), X < Y.
```

Figure 5: Additional ASP encoding for learning original ADMGs, in addition to ours.

```
nodes(3).   % three nodes
set(0..7).  % all subsets of three nodes

% observations
dep(1,2,0,0,1).
dep(1,2,4,0,1).
dep(2,3,0,0,1).
dep(2,3,1,0,1).
dep(1,3,0,0,1).
dep(1,3,2,0,1).

% interventions on the node 3
dep(1,2,4,0,3,1).
indep(2,3,0,3,1).
indep(2,3,1,3,1).
indep(1,3,0,3,1).
indep(1,3,2,3,1).
```

Figure 6: ASP encoding of the (in)dependences in the domain.

given the set index C after having intervened on the node I. Observations correspond to $I = 0$. The penalty for failing to represent an (in)dependence is W. The penalty for failing to represent a dependence is actually superfluous in our algorithm since, recall, rule 30 in Figure 4 enforces that all the dependences in the input are represented. Note also that it suffices to specify all the (in)dependences between pair of nodes, because these identify uniquely the rest of the independences in the probability distribution (Studený, 2005, Lemma 2.2). Note also that we do not assume that the probability distribution at hand is faithful to some ADMG or satisfies the composition property, as it is the case in most heuristic learning algorithms.

By calling the ASP solver with the encodings of the learning algorithm and the (in)dependences in the domain, the solver will essentially perform an exhaustive search over the space of graphs, and will output the graphs with the smallest penalty. Specifically, when only the observations are used (i.e. the last five lines of Figure 6 are removed), the learning algorithm finds 37 optimal models. Among them, we have UGs such as `line(1,2) line(1,3) line(2,3)`, DAGs such as `arrow(3,1) arrow(1,2) arrow(3,2)`, AMP CGs such as `line(1,2) arrow(3,1) arrow(3,2)`, and ADMGs such as `line(1,2) line(2,3) arrow(1,2)` or `line(1,2) line(1,3) arrow(2,3)`. When all the observations and interventions available are used, the learning algorithm finds 18 optimal models. These are the models out the 37 models found before that have no directed edge coming out of the node 3. This is the expected result given the last four lines in Figure 6. Note that the output still includes the ADMGs mentioned before.

Finally, the ASP code can easily be extended as shown in Figure 5 to learn not only our ADMGs but also original ADMGs. Note that the second line forbids graphs with both undirected and bidirected edges. This results in 34 optimal models: The 18 previously found plus 16 original ADMGs, e.g. `biarrow(1,2) biarrow(1,3) arrow(1,2)` or `biarrow(1,2) biarrow(1,3) arrow(2,3)`.

7 DISCUSSION

In this work, we have introduced ADMGs as an extension of AMP CGs by (i) relaxing the semidirected acyclity constraint so that only directed cycles are forbidden, and (ii) allowing up to two edges between any pair of nodes. We have introduced and proved the equivalence of global, and ordered local and pairwise Markov properties for the new models. We have also shown that when the random variables are continuous, the new models can be interpreted as systems of structural equations with correlated errors. This has enabled us to adapt Pearl's *do*-calculus to them. We have shown that our models complement those used in Pearl's *do*-calculus, as there are cases where the identification of the causal effect of an intervention is not possible with the latter but is possible with the former, and vice versa. Finally, we have described an exact algorithm for learning the new models from observational and interventional data. Next, we plan to unify the original and our ADMGs to allow directed, undirected and bidirected edges.

References

Andersson, S. A., Madigan, D. and Perlman, M. D. Alternative Markov Properties for Chain Graphs. *Scandinavian Journal of Statistics*, 28:33-85, 2001.

Bareinboim, E. and Tian, J. Recovering Causal Effects From Selection Bias. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 3475-3481, 2015.

Cox, D. R. and Wermuth, N. *Multivariate Dependencies - Models, Analysis and Interpretation*. Chapman & Hall, 1996.

Evans, R. J. and Richardson, T. S. Marginal log-linear Parameters for Graphical Markov Models. *Journal of the Royal Statistical Society B*, 75:743-768, 2013.

Biere, A., Heule, M., van Maaren, H. and Walsh, T. (editors). *Handbook of Satisfiability*. IOS Press, 2009.

Gebser, M., Kaufmann, B., Kaminski, R., Ostrowski, M., Schaub, T., Schneider, M. Potassco: The Potsdam Answer Set Solving Collection. *AI Communications*, 24:107-124, 2011.

Gelfond, M. and Lifschitz, V. The Stable Model Semantics for Logic Programming. In *Proceedings of 5th Logic Programming Symposium*, 1070-1080, 1988.

Kang, C. and Tian, J. Markov Properties for Linear Causal Models with Correlated Errors. *Journal of Machine Learning Research*, 10:41-70, 2009.

Koster, J. T. A. On the Validity of the Markov Interpretation of Path Diagrams of Gaussian Structural Equations Systems with Correlated Errors. *Scandinavian Journal of Statistics*, 26:413-431, 1999.

Koster, J. T. A. Marginalizing and Conditioning in Graphical Models. *Bernoulli*, 8:817-840, 2002.

Lauritzen, S. L. *Graphical Models*. Oxford University Press, 1996.

Levitz, M., Perlman M. D. and Madigan, D. Separation and Completeness Properties for AMP Chain Graph Markov Models. *The Annals of Statistics*, 29:1751-1784, 2001.

Niemelä, I. Logic Programs with Stable Model Semantics as a Constraint Programming Paradigm. *Annals of Mathematics and Artificial Intelligence*, 25:241-273, 1999.

Pearl, J. Causal Diagrams for Empirical Research. *Biometrika*, 82:669-688, 1995.

Pearl, J. *Causality: Models, Reasoning, and Inference* (2nd ed.). Cambridge University Press, 2009.

Pearl, J. and Wermuth, N. When Can Association Graphs Admit a Causal Explanation ? In *Proceedings of the 4th International Workshop on Artificial Intelligence and Statistics*, 141-150, 1993.

Richardson, T. Markov Properties for Acyclic Directed Mixed Graphs. *Scandinavian Journal of Statistics*, 30:145-157, 2003.

Richardson, T. and Spirtes, P. Ancestral Graph Markov Models. *The Annals of Statistics*, 30:962-1030, 2002.

Sadeghi, K. and Lauritzen, S. L. Markov Properties for Mixed Graphs. *Bernoulli*, 20:676-696, 2014.

Simons, P., Niemelä, I. and Soininen, T. Extending and Implementing the Stable Model Semantics. *Artificial Intelligence*, 138:181-234, 2002.

Sonntag, D. and Peña, J. M. Chain Graph Interpretations and their Relations Revisited. *International Journal of Approximate Reasoning*, 58:39-56, 2015.

Sonntag, D. and Peña, J. M. On Expressiveness of the Chain Graph Interpretations. *International Journal of Approximate Reasoning*, 68:91-107, 2016.

Studený, M. *Probabilistic Conditional Independence Structures*. Springer, 2005.