Learning from Pairwise Marginal Independencies

Johannes Textor
Theoretical Biology & Bioinformatics
Utrecht University, The Netherlands
johannes.textor@gmx.de

Alexander Idelberger
Theoretical Computer Science
University of Lübeck, Germany
alex@pirx.de

Maciej Liśkiewicz
Theoretical Computer Science
University of Lübeck, Germany
liskiewi@tcs.uni-luebeck.de

Abstract

We consider graphs that represent pairwise marginal independencies amongst a set of variables (for instance, the zero entries of a covariance matrix for normal data). We characterize the directed acyclic graphs (DAGs) that faithfully explain a given set of independencies, and derive algorithms to efficiently enumerate such structures. Our results map out the space of faithful causal models for a given set of pairwise marginal independence relations. This allows us to show the extent to which causal inference is possible without using conditional independence tests.

1 INTRODUCTION

DAGs and other graphical models encode conditional independence (CI) relationships in probability distributions. Therefore, CI tests are a natural building block of algorithms that infer such models from data. For example, the PC algorithm for learning DAGs (Kalisch and Bühlmann, 2007) and the FCI (Spirtes et al., 2000) and RFCI (Colombo et al., 2012) algorithms for learning maximal ancestral graphs are all based on CI tests.

CI testing is still an ongoing research topic, to which the UAI community is contributing (e.g. Zhang et al., 2011; Doran et al., 2014). But at least for continuous variables, CI testing will always remain more difficult than testing marginal independence for quite fundamental reasons (Bergsma, 2004). Intuitively, the difficulty is that two variables x and y could be dependent "almost nowhere", e.g., for only a few values of the conditioning variable z. This suggests a two-staged approach to structure learning: first try to learn as much as possible from simpler independence tests before applying CI tests. Here, we present a theoretical basis for extracting as much information as possible from the simplest kind of stochastic independence – pairwise marginal independence.

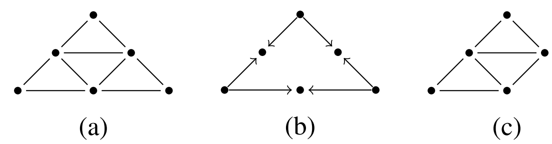


Figure 1: (a) A *marginal independence graph* $\mathcal{U}$ whose missing edges represent pairwise marginal independencies. (b) A *faithful DAG* $\mathcal{G}$ entailing the same set of pairwise marginal independencies as $\mathcal{U}$. (c) A graph for which no such faithful DAG exists.

More precisely, we will consider the following problem. We are given the set of pairwise marginal independencies that hold amongst some variables of interest. Such sets can be represented as graphs whose missing edges correspond to independencies (Figure 1a). We call such graphs *marginal independence graphs*. We wish to find DAGs on the same variables that entail exactly the given set of pairwise marginal independencies (Figure 1b). We call such DAGs *faithful*. Sometimes no such DAGs exist (e.g., Figure 1c). Else, we are interested in finding the set of *all* faithful DAGs, hoping that this set will be substantially smaller than the set of all possible DAGs on the same variables. Those candidate DAGs could then be probed further by using joint marginal or conditional independence tests.

Other authors have represented marginal (in)dependencies using bidirected graphs (Drton and Richardson, 2003; Richardson, 2003; Drton and Richardson, 2008b), instead of undirected graphs like we do here. We hope that the reader is compensated for this small departure from community standards by the lower amount of clutter in our figures, and the greater ease to link our work to standard graph theoretical results. We also emphasize that we model only pairwise, and not higher-order joint dependencies. However, for Gaussian data, pairwise independence entails joint independence. In that case, our marginal independence graphs are equivalent to *covariance graphs* (Cox and Wermuth, 1993; Pearl and Wermuth, 1994; Drton and Richardson, 2003, 2008a; Peña, 2013), whose missing edges represent zero covariances.

Our results generalize the work of Pearl and Wermuth (1994) who showed (but did not prove) how to find *some* faithful DAGs for a given covariance graph. We review these and other connections to related work in Section 3 where we also link our problem to the theory of partially ordered sets (posets). This connection allows us to identify certain maximal and minimal faithful DAGs. Based on these "boundary DAGs" we then derive a characterization of all faithful DAGs (Section 4), and construct related enumeration algorithms (Section 5). We use these algorithms to explore the combinatorial structure of faithful DAG models (Section 6) which leads, among other things, to a quantification of how much pairwise marginal independencies reduce structural causal uncertainty. Finally, we ask what happens when a set of independencies can *not* be explained by any DAG. How many additional variables will we need? We prove that this problem is NP-hard (Section 7).

Preliminary versions of many of the results presented in this paper were obtained in the Master's thesis of the second author (Idelberger, 2014).

2 PRELIMINARIES

In this paper we use the abbreviation *iff* for the connective "if and only if". A graph $\mathcal{G} = (V, E)$ consists of a set of nodes (variables) V and set of edges E. We consider undirected graphs (which we simply refer to as graphs), directed graphs, and mixed graphs that can have both undirected edges (denotes as $x - y$) and directed edges (denoted as $x \to y$). Two nodes are *adjacent* if they are linked by any edge. A *clique* in a graph is a node set $C \subseteq V$ such that all $u, v \in C$ are adjacent. Conversely, an *independent set* is a node set $I \subseteq V$ in which no two nodes $u, v \in I$ are adjacent. A *maximal clique* is a clique for which no proper superset of nodes is also a clique. For any $v \in V$, the *neighborhood* $N(v)$ is the set of nodes adjacent to v and the *boundary* $Bd(v)$ is the neighborhood of v including v, i.e. $Bd(v) = N(v) \cup \{v\}$. A node v is called *simplicial* if $Bd(v)$ is a clique. Equivalently, v is simplicial iff $\mathrm{Bd}(v) \subseteq \mathrm{Bd}(w)$ for all $w \in N(v)$ (Kloks et al., 2000). A clique that contains simplicial nodes is called a *simplex*. Every simplex is a maximal clique, and every simplicial node belongs to exactly one simplex. The *degree* $d(v)$ of a node v is $|N(v)|$. If for two graphs $\mathcal{G} = (V, E(\mathcal{G}))$ and $\mathcal{G}' = (V, E(\mathcal{G}'))$ we have $E(\mathcal{G}) \subseteq E(\mathcal{G}')$, then $\mathcal{G}$ is an *edge subgraph* of $\mathcal{G}'$ and $\mathcal{G}'$ is an *edge supergraph* of $\mathcal{G}$. The *skeleton* of a directed graph $\mathcal{G}$ is obtained by replacing every edge $u \to v$ by an undirected edge $u - v$.

A *path* of length $n - 1$ is a sequence of n distinct nodes in which successive nodes are pairwise adjacent. A *directed path* $x \to \ldots \to y$ consists of directed edges that all point towards y. In a directed graph, a node u is an *ancestor* of another node v if $u = v$ or if there is a directed path $u \to \cdots \to v$. For each edge $u \to v$, we say that u is a *parent* of v and v is a *child* of u. If two nodes u, v in a directed graph have a common ancestor w (which can be u or v), then the path $u \leftarrow \ldots \leftarrow w \to \ldots \to v$ is called a *trek* connecting u and v. A DAG is called *transitive* if, for all $u \neq v$, it contains an edge $u \to v$ whenever there is a directed path from u to v. Given a DAG $\mathcal{G}$, the *transitive closure* is the unique transitive graph that implies the same ancestor relationships as $\mathcal{G}$, whereas the *transitive reduction* is the unique edge-minimal graph that implies the same ancestor relationships.

In this paper we encounter several well-known graph classes, e.g., chordal graphs and trivially perfect graphs. We will give brief definitions when appropriate, but we direct the reader to the excellent survey by Brandstädt et al. (1999) for further details.

3 SIMPLE MARGINAL INDEPENDENCE GRAPHS

In this section we define the class of graphs which can be explained using a directed acyclic graph (DAG) on the same variables. We will refer to such graphs as *simple marginal independence graphs* (SMIGs).

Definition 3.1. *A graph $\mathcal{U} = (V, E(\mathcal{U}))$ is called the* simple marginal independence graph *(SMIG), or* marginal independence graph *of a DAG $\mathcal{G} = (V, E(\mathcal{G}))$ if for all $v, w \in V$, $v - w \in E(\mathcal{U})$ iff v and w have a common ancestor in $\mathcal{G}$. If $\mathcal{U}$ is the marginal independence graph of $\mathcal{G}$ then we also say that $\mathcal{G}$ is* faithful *to $\mathcal{U}$.* $\mathbf{SMIG}$ *is the set of all graphs $\mathcal{U}$ for which there exists a faithful DAG $\mathcal{G}$. Note that each DAG has exactly one marginal independence graph.*

Again, we point out that marginal independence graphs are often called (and drawn as) *bidirected graphs* in the literature, though the term "marginal independence graph" has also been used by various authors (e.g. Tan et al., 2014).

3.1 SMIGs and Dependency Models

In this subsection we recall briefly the general setting for modeling (in)dependencies proposed by Pearl and Verma (1987) and show the relationship between that model and SMIGs. In the definitions below V denotes a set of variables and X, Y and Z are three disjoint subsets of V.

Definition 3.2 (Pearl and Verma (1987))**.** *A* dependency model *$\mathcal{M}$ over V is any subset of triplets (X, Z, Y) which represent independencies, that is, $(X, Z, Y) \in \mathcal{M}$ asserts that X is independent of Y given Z.*

A probabilistic dependency model *$\mathcal{M}_P$ is defined in terms of a probability distribution P over V. By definition $(X, Z, Y) \in \mathcal{M}_P$ iff for any instantiation $\hat{x}$, $\hat{y}$ and $\hat{z}$ of the variables in these subsets $P(\hat{x} \mid \hat{y}\ \hat{z}) = P(\hat{x} \mid \hat{z})$.*

A directed acyclic graph dependency model $\mathcal{M}_\mathcal{G}$ *is defined in terms of a DAG* $\mathcal{G}$. *By definition* $(X, Z, Y) \in \mathcal{M}_\mathcal{G}$ *iff* X *and* Y *are d-separated by* Z *in* $\mathcal{G}$ *(for a definition of d-separation by a set* Z *see Pearl and Verma (1987)).*

We define a *marginal* dependency model, resp. marginal probabilistic and marginal DAG dependency model, analogously as Pearl and Verma (1987) with the restriction that the second component of any triple (X, Z, Y) is the empty set. Thus, such marginal dependency models are sets of pairs (X, Y). It is easy to see that the following properties are satisfied.

Lemma 3.3. *Let* $\mathcal{M}$ *be a marginal probabilistic dependency model or a marginal DAG dependency model. Then* $\mathcal{M}$ *is closed under:*
Symmetry: $(X, Y) \in \mathcal{M} \Leftrightarrow (Y, X) \in \mathcal{M}$ *and*
Decomposition: $(X, Y \cup W) \in \mathcal{M} \Rightarrow (X, Y) \in \mathcal{M}$.
Moreover, if $\mathcal{M}$ *is a marginal DAG dependency model then it is also closed under*
Union: $(X, Y), (X, W) \in \mathcal{M} \Rightarrow (X, Y \cup W) \in \mathcal{M}$.

The marginal probabilistic dependency model is not closed under union in general. For instance, consider two independent, uniformly distributed binary variables y and w and let $x = y \oplus w$, where $\oplus$ denotes xor of two bits. For the model $\mathcal{M}_P$ defined in terms of probability over x, y, w we have that $(\{x\}, \{y\})$ and $(\{x\}, \{w\})$ belong to $\mathcal{M}_P$ but $(\{x\}, \{y, w\})$ does not.

In this paper we will *not* assume that the marginal independencies in the data are closed under union. Instead, we only consider pairwise independencies, which we formalize as follows.

Definition 3.4. *Let* $\mathcal{M}$ *be a marginal probabilistic dependency model over* V. *Then the simple marginal independence graph* $\mathcal{U} = (V, E(\mathcal{U}))$ *of* $\mathcal{M}$ *is the graph in which* $x - y \in E(\mathcal{U})$ *iff* $(\{x\}, \{y\}) \notin \mathcal{M}$.

Thus, in general, marginal independence graphs do not contain any information on higher-order *joint* independencies present in the data. However, under certain common parametric assumptions, dependency models would be closed under union as well. This holds, for instance, if the data are normally distributed. In that case, marginal independence is equivalent to zero covariance, pairwise independence implies joint independence, and marginal independence graphs become covariance graphs.

The following is not difficult to see.

Proposition 3.5. *A marginal dependency model* $\mathcal{M}$ *which is closed under symmetry, decomposition, and union coincides with the transitive closure of* $\{(\{x\}, \{y\}) : x, y \in V\} \cap \mathcal{M}$ *over symmetry and union.*

This Proposition entails that if the marginal dependencies in the data are closed under these properties, then the entire marginal dependency model is represented by the marginal independence graph.

3.2 SMIGs and Partially Ordered Sets

To reach our aim of a complete and constructive characterization of the DAGs faithful to a given SMIG, it is useful to observe that marginal independence graphs are invariant with respect to the insertion or deletion of transitive edges from the DAG. We formalize this as follows.

Definition 3.6. *A (labelled)* poset $\mathcal{P}$ *is a DAG that is identical to its transitive closure.*

Proposition 3.7. *The marginal independence graphs of a DAG* $\mathcal{G}$ *and its transitive closure* $\mathcal{P}(\mathcal{G})$ *are identical.*

Proof. Two nodes are not adjacent in the marginal independence graph iff they have no common ancestor in the DAG. Transitive edges do not influence ancestral relationships. □

We thus restrict our attention to finding *posets* that are faithful to a given SMIG. Note that faithful DAGs can then be obtained by deleting transitive edges from faithful posets; since no DAG obtained in this way can be an edge subgraph of two different posets, this construction is unique and well-defined. In particular, by deleting *all* transitive edges from a poset, we obtain a sparse graphical representation of the poset as defined below.

Definition 3.8. *Given a poset* $\mathcal{P} = (V, E)$, *its* transitive reduction *is the unique DAG* $\mathcal{G}_\mathcal{P} = (V, E')$ *for which* $\mathcal{P}(\mathcal{G}) = \mathcal{P}$ *and* E' *is the smallest set where* $E' \subseteq E$.

Transitive reductions are also known as *Hasse diagrams*, though Hasse diagrams are usually unlabeled. Different posets can have the same marginal independence graphs, e.g. the posets with Hasse diagrams $\mathcal{P}_1 = x \rightarrow y \rightarrow z$ and $\mathcal{P}_2 = x \leftarrow y \rightarrow z$. Similarly, Markov equivalence is a sufficient but not necessary condition to inducing the same marginal independence graphs (adding an edge $x \rightarrow z$ to $\mathcal{P}_2$ changes the poset and the Markov equivalence class, but not the marginal independence graph).

3.3 Recognizing SMIGs

We first recall existing results that show which graphs admit a faithful DAG at all, and how to find such DAGs if possible. Note that many of these results have been stated without proof (Pearl and Wermuth, 1994), but our connection to posets will make some of these proofs straightforward. The following notion related to posets is required.

Definition 3.9 (Bound graph (McMorris and Zaslavsky, 1982))**.** *For a poset* $\mathcal{P} = (V, E)$, *the* bound graph $\mathcal{B} = (V, E')$ *of* $\mathcal{P}$ *is the graph where* $x - y \in E'$ *iff* x *and* y *share a* lower bound, *i.e., have a common ancestor in* $\mathcal{P}$.

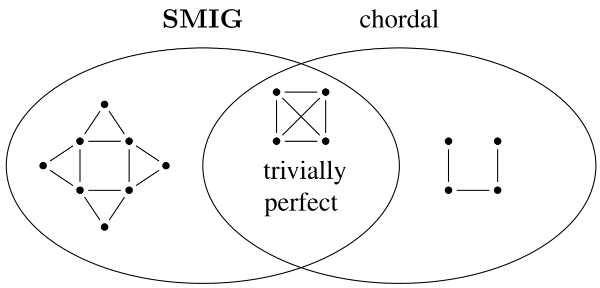


Figure 2: Relation between chordal graphs, trivially perfect graphs, and $\mathbf{SMIG}$. In graph theory, $\mathbf{SMIG}$ is known as the class of (upper/lower) *bound graphs* (Cheston and Jap, 2006).

Theorem 3.10. $\mathbf{SMIG}$ *is the set of all graphs for which every edge is contained in a simplex.*

Proof. This is Theorem 2 in Pearl and Wermuth (1994) (who referred to simplexes as "exterior cliques"). Alternatively, we can observe that the marginal independence graph $\mathcal{U}$ of a poset $\mathcal{P}$ (Definition 3.1) is equal to its bound graph (Definition 3.9). The characterization of bound graphs as "edge simplicial" graphs has been proven by McMorris and Zaslavsky (1982) by noting that simplicial nodes in $\mathcal{U}$ correspond to possible minimal elements in $\mathcal{P}$. We note that this result predates the equivalent statement in Pearl and Wermuth (1994). □

Though all bound graphs have a faithful poset, not all bound graphs have one with the same skeleton; see Figure 1a,b for a counterexample. However, the graphs for which a poset with the same skeleton can be found are nicely characterizable in terms of forbidden subgraphs.

Theorem 3.11 (Pearl and Wermuth (1994))**.** *Given a graph $\mathcal{U}$, a DAG $\mathcal{G}$ that is faithful to $\mathcal{U}$ and has the same skeleton exists iff $\mathcal{U}$ is trivially perfect (i.e., $\mathcal{U}$ has no P_4= — — — nor a C_4= □ as induced subgraph).*

It is known that the trivially perfect graphs are the intersection of the bound graphs and the chordal graphs (Figure 2; Cheston and Jap, 2006).

This nice result begs the question whether a similar characterization is also possible for $\mathbf{SMIG}$. As the following observation shows, that is not the case.

Proposition 3.12. *Every graph $\mathcal{U}$ is an induced subgraph of some graph $\mathcal{U}' \in \mathbf{SMIG}$.*

Proof. Take any graph $\mathcal{U} = (V, E)$ and construct a new graph $\mathcal{U}'$ as follows. For every edge $e = u - v$ in $\mathcal{U}$, add a new node v_e to V and add edges $v_e - u$ and $v_e - v$. Obviously $\mathcal{U}$ is an induced subgraph of $\mathcal{U}'$. To see that $\mathcal{U}'$ is in $\mathbf{SMIG}$, consider the DAG $\mathcal{G}$ consisting of the nodes in $\mathcal{U}'$ and the edges $v \leftarrow v_e \rightarrow u$ and for each newly added node in $\mathcal{U}'$. Then $\mathcal{U}$ is the marginal independence graph of $\mathcal{G}$. □

The graph class characterization implies efficient recognition algorithms for SMIGs.

Theorem 3.13. *It can be tested in polynomial time whether a graph $\mathcal{U}$ is a SMIG.*

Proof. Verifying the graphical condition of Theorem 3.10 amounts to testing whether all edges reside within a simplex. However, knowing that SMIGs are bound graphs, we can apply an efficient algorithm for bound graph recognition that uses radix sort and simplex elimination and achieves a runtime of $\mathcal{O}(n + sm)$ (Skowrońska and Sysło, 1984), where $s \leq n$ is the number of simplexes in the graph. This is typically better than $\mathcal{O}(n^3)$ because large m implies small s and vice versa. Alternatively, we can apply known fast algorithms to find all simplicial nodes (Kloks et al., 2000). □

4 FINDING FAITHFUL POSETS

We now ask how to find faithful DAGs for simple marginal independence graphs. We observed that marginal independence graphs cannot distinguish between transitively equivalent DAGs, so a perhaps more natural question is: which *posets* are faithful to a given graph? As pointed out before, we can obtain all DAGs from faithful posets in a unique manner by removing transitive edges. A further advantage of the poset representation will turn out to be that the "smallest" and "largest" faithful posets can be characterized uniquely (up to isomorphism); as we shall also see, this is not as easy for DAGs, except for marginal independence graphs in a certain subclass.

4.1 Maximal Faithful Posets

Our first aim is to characterize the "upper bound" of the faithful set. That is, we wish to identify those posets for which no edge supergraph is also faithful. We will show that a construction described by Pearl and Wermuth (1994) solves exactly this problem.

Definition 4.1. *For a graph $\mathcal{U} = (V, E(\mathcal{U}))$, the* sink graph *$\mathcal{S}(\mathcal{U}) = (V, E(\mathcal{S}(\mathcal{U})))$ is constructed as follows: for each edge $u - v$ in $\mathcal{U}$, add to $E(\mathcal{S}(\mathcal{U}))$: (1) an edge $u \rightarrow v$ if $Bd(u) \subsetneq Bd(v)$; (2) an edge $u \leftarrow v$ if $Bd(u) \supsetneq Bd(v)$; (3) an edge $u - v$ if $Bd(u) = Bd(v)$.*

For instance, the sink graph of the graph in Figure 1a is the graph in Figure 1b.

Definition 4.2 (Pearl and Wermuth (1994))**.** *A* sink orientation *of a graph $\mathcal{U}$ is any DAG obtained by replacing every undirected edge of $\mathcal{S}(\mathcal{U})$ by a directed edge.*

We first need to state the following.

Lemma 4.3. *Every sink orientation of $\mathcal{U}$ is a poset.*

Proof. Fix a sink orientation $\mathcal{G}$ and consider any chain $x \to y \to z$. By construction, this implies that $\text{Bd}(x) \subsetneq \text{Bd}(z)$. Hence, if x and z are adjacent in the sink graph, then the only possible orientation is $x \to z$. There can be two reasons why x and z are not adjacent in the sink graph: (1) They are not adjacent in $\mathcal{U}$. But then $\mathcal{G}$ would not be faithful, since $\mathcal{G}$ implies the edge $x - z$. (2) The edge was not added to the sink graph. But this contradicts $\text{Bd}(x) \subsetneq \text{Bd}(z)$. □

This Lemma allows us to strengthen Theorem 2 by Pearl and Wermuth (1994) in the sense that we can replace "DAG" by "maximal poset" (emphasized):

Theorem 4.4. *$\mathcal{P}$ is a* maximal poset *faithful to $\mathcal{U}$ iff $\mathcal{P}$ is a sink orientation of $\mathcal{U}$.*

The following is also not hard to see.

Lemma 4.5. *For a SMIG $\mathcal{U}$, every DAG $\mathcal{G}$ that is faithful to $\mathcal{U}$ is a subgraph of some sink orientation of $\mathcal{U}$.*

Proof. Obviously the skeleton of $\mathcal{G}$ cannot contain edges that are not in $\mathcal{U}$. So, suppose $x \to y$ is an edge in $\mathcal{G}$ but conflicts with the sink orientation; that is, the sink graph contains the edge $y \to x$. That is the case only if $\text{Bd}_\mathcal{U}(y)$ is a proper subset of $\text{Bd}_\mathcal{U}(x)$. However, in the marginal independence graph of $\mathcal{G}$, any node that is adjacent to x (has a common ancestor) must also be adjacent to y. Thus, the marginal independence graph of $\mathcal{G}$ cannot be $\mathcal{U}$. □

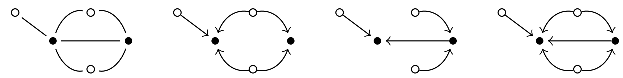
Every maximal faithful poset for $\mathcal{U}$ can be generated by first fixing a topological ordering of $\mathcal{S}(\mathcal{U})$ and then generating the DAG that corresponds to that ordering, an idea that has also been mentioned by Drton and Richardson (2008a). This construction makes it obvious that all maximal faithful posets are isomorphic.

For curiosity of the reader, we note that $\mathcal{S}(\mathcal{U})$ can also be viewed as a *complete partially directed acyclic graph* (CPDAG), which represents the Markov equivalence class of edge-maximal DAGs that are faithful with $\mathcal{U}$. CPDAGs are used in the context of inferring DAGs from data (Spirtes et al., 2000; Chickering, 2003; Kalisch and Bühlmann, 2007), which is only possible up to Markov equivalence.

4.2 Minimal Faithful Posets

A minimal faithful poset to $\mathcal{U}$ is one from which no further relations can be deleted without entailing more independencies than are given by $\mathcal{U}$.

Definition 4.6. *Let $\mathcal{U} = (V, E)$ be a graph and let $I \subseteq V$ be an independent set. Then $I^{\to}_{\mathcal{U}}$ is the poset consisting of the nodes in I, their neighbors in $\mathcal{U}$, and directed edges $i \to j$ for each i, j where $j \in N(i)$.*



(a) (b) (c) (d)

Figure 3: (a) A graph $\mathcal{U}$ with three simplicial nodes I (open circles). (b) Its unique minimal faithful poset $I^{\to}_{\mathcal{U}}$. (c,d) The unique faithful DAGs with minimum (c) or maximum (d) numbers of edges.

For example, Figure 3b shows the unique $I^{\to}_{\mathcal{U}}$ for the graph in Figure 3a.

Theorem 4.7. *Let $\mathcal{U} = (V, E) \in \mathcal{U}$. Then a poset $\mathcal{P}$ is a minimal poset faithful to $\mathcal{U}$ iff $\mathcal{P} = I^{\to}_{\mathcal{U}}$ for a set I consisting of one simplicial vertex for each simplex.*

Proof. We first show that if I is a set consisting of one simplicial node for each simplex, then $I^{\to}_{\mathcal{U}}$ is a minimal faithful poset. Every edge $e \in E(\mathcal{U})$ resides in a simplex, so it is either adjacent to I or both of its endpoints are adjacent to some $i \in I$. In both cases, $I^{\to}_{\mathcal{U}}$ implies e. Also $I^{\to}_{\mathcal{U}}$ does not imply more edges than are in $\mathcal{U}$. Now, suppose we delete an edge $i \to x$ from $I^{\to}_{\mathcal{U}}$. This edge must exist in $\mathcal{U}$, else i was not simplicial. But now $I^{\to}_{\mathcal{U}}$ no longer implies this edge. Thus, $I^{\to}_{\mathcal{U}}$ is minimal. Second, assume that $\mathcal{P}$ is a minimal faithful poset. Assume $\mathcal{P}$ would contain a sequence of two directed edges $x \to y \to z$. Then $\mathcal{P}$ would also contain the edge $x \to z$. But then $y \to z$ could be deleted from $\mathcal{P}$ without changing the dependency graph, and $\mathcal{P}$ was not minimal. So, $\mathcal{P}$ does not contain any directed path of length more than 1. Next, observe that for each simplex in $\mathcal{U}$, the nodes must all have a common ancestor in $\mathcal{P}$. Without paths of length > 1, this is only possible if one node i in the simplex is a parent of all other nodes, and there are no edges among the child nodes of i. Finally, each such i must be a simplicial node in $\mathcal{U}$; otherwise, it would reside in two or more simplexes, and would have to be the unique parent in those simplexes. But then the children of i would form a single simplex in $\mathcal{U}$. □

Like the maximal posets, all minimal posets are thus isomorphic. We point out that the minimal posets contain no transitive edges and therefore, they are also edge-minimal faithful DAGs. However, this does not imply that minimal posets have the smallest possible number of edges amongst all faithful DAGs (Figure 3). There appears to be no straightforward characterization of the DAGs with the smallest number of edges for marginal independence graphs in general. However, a beautiful one exists for the subclass of trivially perfect graphs.

Definition 4.8. *A* tree poset *is a poset whose transitive reduction is a tree (with edges pointing towards the root).*

Theorem 4.9. *A connected SMIG $\mathcal{U}$ has a faithful tree poset iff it is trivially perfect.*

Proof. The bound graph of a tree poset is identical to its *comparability graph* (Brandstädt et al., 1999), which is the skeleton of the poset. Comparability graphs of tree posets coincide with trivially perfect graphs (Wolk, 1965). □

Since no connected graph on n nodes can have fewer edges than the transitive reduction of a tree poset on the same nodes (i.e., $n-1$), tree posets coincide with faithful DAGs having the smallest possible number of edges.

How do we construct a tree for a given trivially perfect graph? Every such graph must have a *central point*, which is a node that is adjacent to all other nodes. We set this node as the sink of the tree, and continue recursively with the subgraphs obtained after removing the central point. Each subgraph is also trivially perfect and can thus be oriented into a tree. After we are done, we link the sinks of the trees of the subgraphs to the original central point to obtain the full tree (Wolk, 1965).

5 FINDING FAITHFUL DAGS

If a given marginal independence graph $\mathcal{U}$ admits faithful DAG models, then it is of interest to enumerate these. A trivial enumeration procedure is the following: start with the sink graph of $\mathcal{U}$, choose an arbitrary edge e, and form all 2 or 3 subgraphs obtained by keeping e (if it is directed), orienting e (if it is undirected), or deleting it. Apply the procedure recursively to these subgraphs. During the recursion, do not touch edges that have been previously chosen. If the current graph is a DAG that is faithful to $\mathcal{U}$, output it; otherwise, stop the recursion.

However, we can do better by exploiting the results of the previous section, which will allow us to derive enumeration algorithms that generate representations of multiple DAGs at each step.

5.1 Enumeration of Faithful DAGs

Having characterized the maximal and minimal faithful posets, we are now ready to construct an enumeration procedure for all DAGs that are faithful to a given graph. We first state the following combination of Theorem 4.4 and Theorem 4.7.

Proposition 5.1. *A DAG $\mathcal{G} = (V, E(\mathcal{G}))$ is faithful to a SMIG $\mathcal{U} = (V, E(\mathcal{U}))$ iff (1) $\mathcal{G}$ is an edge subgraph of some sink orientation of $\mathcal{U}$ and (2) the transitive closure of $\mathcal{G}$ is an edge supergraph of $I_{\mathcal{U}}^{\rightarrow}$ for some node set I consisting of one simplicial node for each simplex.*

From this observation, we can derive our first construction procedure for faithful DAGs.

Proposition 5.2. *A DAG $\mathcal{G}$ is faithful to a SMIG $\mathcal{U} = (V, E(\mathcal{U}))$ iff it can be generated by the following steps. (1) Pick any set $I \subseteq V$ consisting of one simplicial node for each simplex. (2) Generate any DAG on the nodes $V \setminus I$ that is an edge subgraph of some sink orientation of $\mathcal{U}$. (3) Add any subset of edges from $I_{\mathcal{U}}^{\rightarrow}$ such that the transitive closure of the resulting graph contains all edges of $I_{\mathcal{U}}^{\rightarrow}$.*

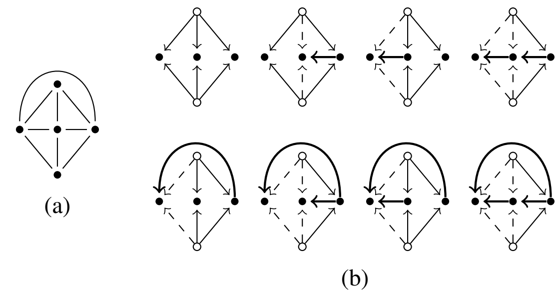


Figure 4: Example of the procedure in Proposition 5.2 that, given a SMIG (a), enumerates all faithful DAGs (b). For brevity, only the graphs that correspond to a fixed topological ordering are displayed. Only one set I (open circles) can be chosen in step (1). Thick edges and filled nodes highlight the DAG $\mathcal{G}$. Mandatory edges (solid) link I to the sources of $\mathcal{G}$; if any such edge was absent, one of the relationships in the poset $I_{\mathcal{U}}^{\rightarrow}$ would be missing. Optional edges (dashed) are transitively implied from the mandatory ones and $\mathcal{G}$.

While step (3) may seem ambiguous, Figure 4 illustrates that after step (2), the edges from $I_{\mathcal{U}}^{\rightarrow}$ decompose nicely into *mandatory* and *optional* ones. This means that we can in fact stop the construction procedure after step (2) and output a "graph pattern", in which some edges are marked as optional. This is helpful in light of the potentially huge space of faithful models, because every graph pattern can represent an exponential number of DAGs.

5.2 Enumeration of Faithful Posets

The DAGs resulting from the procedure in Proposition 5.2 are in general redundant because no care is taken to avoid generating transitive edges. By combining Propositions 5.1 and 5.2, we obtain an algorithm that generates sparse, non-redundant representations of the faithful DAGs.

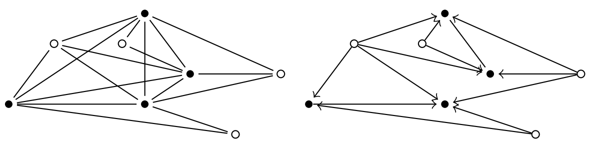
Theorem 5.3. *A poset $\mathcal{P}$ is faithful to $\mathcal{U} = (V, E(\mathcal{U}))$ iff it can be generated by the following steps. (1) Pick any set $I \subseteq V$ consisting of one simplicial node for each simplex. (2) Generate a poset $\mathcal{P}$ on the nodes $V \setminus I$ that is an edge subgraph of some sink orientation of $\mathcal{U}$. (3) Add $I_{\mathcal{U}}^{\rightarrow}$ to $\mathcal{P}$.*

A nice feature of this construction is that step (3) is unambiguous: every choice for I in step (1) and $\mathcal{P}$ in step (2) yields exactly one poset. Figure 5 gives an explicit pseudocode for an algorithm that uses Theorem 5.3 to enumerate all faithful posets.

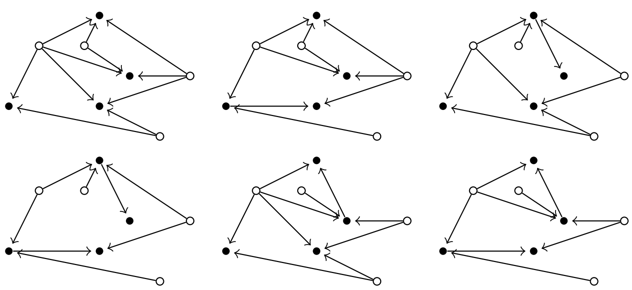
Our algorithm is efficient in the sense that at every inter-

function FAITHFULPOSETS($\mathcal{U} = (V(\mathcal{U}), E(\mathcal{U}))$)
  function LISTPOSETS($\mathcal{G}, \mathcal{S}, R, I_{\mathcal{U}}^{\rightarrow}$)
    if $\mathcal{G}$ is acyclic and atransitive **then**
      Output $\mathcal{G} \cup I_{\mathcal{U}}^{\rightarrow}$
      if skeleton of $\mathcal{G} \subsetneq$ skeleton of $\mathcal{S}$ **then**
        $e \leftarrow$ some edge consistent with $E(\mathcal{S}) \setminus R$
        LISTPOSETS($\mathcal{G}, \mathcal{S}, R \cup \{e\}, I_{\mathcal{U}}^{\rightarrow}$)
        $E(\mathcal{G}) \leftarrow E(\mathcal{G}) \cup \{e\}$
        LISTPOSETS($\mathcal{G}, \mathcal{S}, R \cup \{e\}, I_{\mathcal{U}}^{\rightarrow}$)
  for all node sets I of $\mathcal{U}$ consisting of one simplicial node per simplex **do**
    $\mathcal{G} \leftarrow$ empty graph on nodes of $V(\mathcal{U}) \setminus I$
    $\mathcal{S} \leftarrow$ sink graph of $\mathcal{U}$ on nodes of $V(\mathcal{U}) \setminus I$
    LISTPOSETS($\mathcal{G}, \mathcal{S}, \emptyset, I_{\mathcal{U}}^{\rightarrow}$)

Figure 5: Enumeration algorithm for faithful posets.



(a)



(b)

Figure 6: (a) A graph $\mathcal{U}$ and its sink graph. (b) Transitive reductions of all 6 faithful posets that are generated by Algorithm FAITHFULPOSETS for the input graph (a).

nal node in its recursion tree, it outputs a faithful poset. At every node we need to evaluate whether the current $\mathcal{G}$ is acyclic and atransitive (i.e., contains no transitive edges), which can be done in polynomial time. Also simplexes and their simplicial vertices can be found in polynomial time Kloks et al. (2000). Thus, our algorithm is a *polynomial delay enumeration algorithm* similar to the ones used to enumerate adjustment sets for DAGs (Textor and Liśkiewicz, 2011; van der Zander et al., 2014). Figure 6 shows an example output for this algorithm.

6 EXAMPLE APPLICATIONS

In this section, we apply the previous results to explore some explicit combinatorial properties of SMIGs and their faithful DAGs.

n	connected graphs	conn. SMIGs	unique DAG
2	1	1	0
3	2	2	1
4	6	4	1
5	21	10	2
6	112	27	4
7	853	88	10
8	11,117	328	27
9	261,080	1,460	90
10	11,716,571	7,799	366

Table 1: Comparison of the number of unlabeled connected graphs with n nodes to the number of such graphs that are also SMIGs. For $n = 13$ (not shown), non-SMIGs outnumber SMIGs by more than $10^7 : 1$.

6.1 Counting SMIGs

We revisit the question: when can a marginal independence graph allow a causal interpretation (Pearl and Wermuth, 1994)? More precisely, we ask *how many* marginal independence graphs on n variables are SMIGs. We reformulate this question into a version that has been investigated in the context of poset theory. Let the *height* of a poset $\mathcal{P}$ be the length of a longest path in $\mathcal{P}$. The following is an obvious implication of Theorem 4.7.

Corollary 6.1. *The number $M(n)$ of non-isomorphic SMIGs with n nodes is equal to the number of non-isomorphic posets on n variables of height 1.*

Enumeration of posets is a highly nontrivial problem, and an intensively studied one. The online encyclopedia of integer sequences (OEIS) tabulates $M(n)$ for n up to 40 (Wambach, 2015). We give the first 10 entries of the sequence in Table 1 and compare it to the number of graphs in general (up to isomorphism). As we observe, the fraction of graphs that admit a DAG on the same variables decreases swiftly as n increases.

6.2 Graphs with a Unique Faithful DAG

From a causal inference viewpoint, the best we can hope for is a SMIG to which only single, unique DAG is faithful. The classical example is the graph $\cdot - \cdot - \cdot$, which for more than 3 nodes generalizes to a "star" graph. However, for 5 or more nodes there are graphs other than the star which also induce a single unique DAG. Combining Lemma 4.5 and Theorem 4.7 allows for a simple characterization of all such SMIGs.

Corollary 6.2. *A SMIG $\mathcal{U}$ with n nodes has a unique faithful DAG iff each of its simplexes contains only one simplicial node and its sink orientation equals $I_{\mathcal{U}}^{\rightarrow}$.*

Based on this characterization, we computed the number of

n	posets with n nodes	faithful to C_n
1	1	1
2	3	2
3	19	9
4	219	76
5	4,231	1,095
6	130,023	25,386
7	6,129,859	910,161
8	431,723,379	49,038,872
9	44,511,042,511	3,885,510,411
10	6,611,065,248,783	445,110,425,110

Table 2: Possible labelled posets on n variables before and after observing a complete SMIG C_n.

SMIGs with unique DAGs for n up till 9 (Table 1). Interestingly, this integer sequence does not seem to correspond to any known one.

6.3 Information Content of a SMIG

How much information does a marginal independence graph contain? Let us denote the number of posets on n variables by $P(n)$. After observing a marginal independence graph $\mathcal{U}$, the number of models that are still faithful to the data reduces to size $P(n) - k(\mathcal{U})$, where $k(\mathcal{U}) \leq P(n)$ (indeed, quite often $k(\mathcal{U}) = P(n)$ as we can see in Table 1). Of course, the number $k(\mathcal{U})$ strongly depends on the structure of the SMIG $\mathcal{U}$. But even in the worst case when $\mathcal{U}$ is a complete graph, the space of possible models is still reduced because not all DAGs entail a complete marginal independence graph.

Thus, the following simple consequence of Theorem 4.7 helps to derive a worst-case bound on how much a SMIG reduces structural uncertainty with respect to the model space of posets with n variables.

Corollary 6.3. *The number of faithful posets with respect to a complete graph with n nodes is n times the number of posets with $n - 1$ nodes.*

Table 2 lists the number of possible posets before and after observing a complete SMIG for up to 10 variables. In this sense, at $n = 10$, the uncertainty is reduced about 15-fold.

We note that a similar but more technical analysis is possible for uncertainty reduction with respect to DAGs instead of posets. We omit this due to space limitations.

7 MODELS WITH LATENT VARIABLES

In this section we consider situations in which a graph $\mathcal{U}$ is not a SMIG (which can be detected using the algorithm in Theorem 3.13). Similarly to the definition proposed in Pearl and Verma (1987) for the general dependency models, to obtain faithful DAGs for such graphs we will extend the DAGs with some auxiliary nodes. We generalize Definition 3.1 as follows.

Definition 7.1. *Let $\mathcal{U} = (V, E(\mathcal{U}))$ be a graph and let Q, with $Q \cap V = \emptyset$, be a set of auxiliary nodes. A DAG $\mathcal{G} = (V \cup Q, E(\mathcal{G}))$ is faithful to $\mathcal{U}$ if for all $v, w \in V$, $v - w \in E(\mathcal{U})$ iff v and w have a common ancestor in $\mathcal{G}$.*

The result below follows immediately from Proposition 3.12.

Proposition 7.2. *For every graph $\mathcal{U}$ there exists a faithful DAG $\mathcal{U}$ with some auxiliary nodes.*

Obviously, if $\mathcal{U} \in \mathbf{SMIG}$ then there exists a faithful DAG to $\mathcal{U}$ with $Q = \emptyset$. For $\mathcal{U} \notin \mathbf{SMIG}$, from the proof of Proposition 3.12 it follows that there exists a set Q of at most $|E(\mathcal{U})|$ nodes and a DAG $\mathcal{G}$ such that $\mathcal{G}$ is faithful to $\mathcal{U}$ with auxiliary nodes Q. But the problem arises to minimize the cardinality of Q.

Theorem 7.3. *The problem to decide if for a given graph $\mathcal{U}$ and an integer k, there exists a faithful DAG with at most k auxiliary nodes, is NP-complete.*

Proof. It is easy to see that the problem is in NP. To prove that it is NP-hard, we show a polynomial time reduction from the edge clique cover problem, that is known to be NP-complete (Karp, 1972). Recall that the problem edge clique cover is to decide if for a graph $\mathcal{U}$ and an integer k there exist a set of k subgraphs of $\mathcal{U}$, such that each subgraph is a clique and each edge of $\mathcal{U}$ is contained in at least one of these subgraphs?

Let $\mathcal{U} = (V, E)$ and k be an instance of the edge clique cover problem, with $V = \{v_1, \dots, v_n\}$. We construct the marginal independence graph $\mathcal{U}'$ as follows. Let $W = \{w_1, \dots, w_n\}$. Then $V(\mathcal{U}') = V \cup W$ and $E(\mathcal{U}') = E \cup \{v_i - w_i : i = 1, \dots, n\}$. Obviously, $\mathcal{U}'$ can be constructed from $\mathcal{U}$ in polynomial time. We claim that $\mathcal{U} = (V, E)$ can be covered by $\leq k$ cliques iff for $\mathcal{U}'$ there exists a faithful DAG $\mathcal{G}$ with at most k auxiliary nodes.

Assume first that $\mathcal{U} = (V, E)$ can be covered by at most k cliques, let us say $C_1, \dots, C_{k'}$, with $k' \leq k$. Then we can construct a faithful DAG $\mathcal{G}$ for $\mathcal{U}'$ with k' auxiliary nodes as follows. Its set of nodes is $V(\mathcal{G}) = V \cup W \cup Q$, where $Q = \{q_1, \dots, q_{k'}\}$. The edges $E(\mathcal{G})$ can be defined as

$$\{w_i \to v_i : i = 1, \dots, n\} \cup \bigcup_j \{q_j \to v : v \in C_j\}.$$

It is easy to see that $\mathcal{G}$ is faithful to $\mathcal{U}'$.

Now assume that a DAG $\mathcal{G}$, with at most k auxiliary nodes Q, is faithful to $\mathcal{U}'$. From the construction of $\mathcal{U}'$ it follows that for all different nodes $v_i, v_j \in V$ there is no directed path from v_i to v_j in $\mathcal{G}$. If such a path exists, then v_i is an ancestor of v_j in $\mathcal{G}$. Since $v_i - w_i$ is an edge of $\mathcal{U}'$, the nodes v_i and w_i have a common ancestor in $\mathcal{G}$, which must be also a common ancestor of w_i and v_j – a contradiction because w_i and v_j are not incident in $\mathcal{U}'$. Thus, all treks connecting pairs of nodes from V in $\mathcal{G}$ must contain auxiliary nodes.

Next, we slightly modify $\mathcal{G}$: for each w_i we remove all incident edges and add the new edge $w_i \to v_i$. The resulting graph $\mathcal{G}'$, is a DAG which remains faithful to $\mathcal{U}'$. Indeed, we cannot obtain a directed cycle in the $\mathcal{G}'$ since no w_i has an in-edge and the original $\mathcal{G}$ was a DAG. To see that the obtained DAG remains faithful to $\mathcal{U}'$ note first that after the modifications, w_i and v_i have a common ancestor in $\mathcal{G}$ whereas w_i and v_j, with $i \neq j$, do not. Otherwise, it would imply a directed path from v_i to v_j since w_i is the only possible ancestor of both nodes – a contradiction. Finally, note that any trek connecting v_i and v_j in $\mathcal{G}$ cannot contain a node from W. Similarly, no trek between v_i and v_j in $\mathcal{G}'$ contains a node from W. We get that v_i and v_j have a common ancestor in $\mathcal{G}$ iff they have a common ancestor in $\mathcal{G}'$.

Thus, in $\mathcal{G}'$ the auxiliary nodes Q are incident to V, but not to nodes from W. Below we modify $\mathcal{G}'$ further and obtain a DAG $\mathcal{G}''$, in which every auxiliary node is incident with a node in V via an out-edge only. To this aim we remove from $\mathcal{G}'$ all edges going out from a node in V to a node in Q.

Obviously, if v_i and v_j have a common ancestor in $\mathcal{G}''$, then they also have a common ancestor in $\mathcal{G}'$, because $E(\mathcal{G}'') \subseteq E(\mathcal{G}')$. The opposite direction follows from the fact we have shown at the beginning of this proof that for all different nodes $v_i, v_j \in V$ there is no directed path from v_i to v_j in $\mathcal{G}$. This is true also for $\mathcal{G}'$. Thus, if v_i and v_j have a common ancestor, say x, in $\mathcal{G}'$ then $x \in Q$ and there exist directed paths $x \to y_1 \to \dots y_r \to v_i$ and $x \to y'_1 \to \dots y'_{r'} \to v_j$ such that also all $y_1, \dots, y_r$ and $y'_1, \dots, y'_{r'}$ belong to Q. But from the construction of $\mathcal{G}''$ it follows that both paths belong also to $\mathcal{G}''$.

Since $\mathcal{G}''$ is faithful to $\mathcal{U}$, for every auxiliary node Q the subgraph induced by its children $Ch(Q) \cap V$ in $\mathcal{G}''$ is a clique in $\mathcal{U}'$. Moreover every edge $v_i - v_j$ of the graph $\mathcal{U}$ belongs to at least one such clique. Thus the subgraphs induced by $Ch(q_1) \cap V, \dots, Ch(q_{k'}) \cap V$, with $k' \leq k$, are cliques that cover $\mathcal{U}$. □

8 DISCUSSION

Given a graph that represents a set of pairwise marginal independencies, which causal structures on the same variables might have generated this graph? Here we characterized all these structures, or alternatively, all maximal and minimal ones. Furthermore, we have shown that it is possible to deduce how many exogenous variables (which correspond to simplicial nodes) the causal structure might have, and even to tell whether it might be a tree. For graphs that do not admit a DAG on the same variables, we have studied the problem of explaining the data with as few additional variables as possible, and proved it to be NP-hard. This may be surprising; the related problem of finding a mixed graph that is Markov equivalent to a bidirected graph and has as few bidirected edges as possible is efficiently solvable (Drton and Richardson, 2008a).

The connection to posets emphasizes that sets of faithful DAGs have complex combinatorics. Indeed, if there are no pairwise independent variables, then we obtain the classical poset enumeration problem (Brinkmann and McKay, 2002). Our current, unoptimized implementation of the algorithm in Figure 5 allows us to deal with dense graphs up to about 12 nodes (sparse graphs are easier to deal with). We point out that our enumeration algorithms operate with a "template graph", i.e., the sink orientation. It is possible to incorporate certain kinds of background knowledge, like a time-ordering of the variables, into this template graph by deleting some edges. Such further constraints could greatly reduce the search space. Another additional constraint that could be used for linear models is the precision matrix (Cox and Wermuth, 1993; Pearl and Wermuth, 1994), though finding DAGs that explain a given precision matrix is NP-hard in general (Verma and Pearl, 1993),

We observed that the pairwise marginal independencies substantially reduce structural uncertainty even in the worst case (Table 1). Causal inference algorithms could exploit this to reduce the number of CI tests. The PC algorithm (Kalisch and Bühlmann, 2007), for instance, forms the marginal independence graph as a first stage before performing any CI tests. At that stage, it could be immediately tested if the resulting graph is a SMIG, and if not, the algorithm can terminate as no faithful DAG exists.

In summary, we have mapped out the space of causal structures that are faithful to a given set of pairwise marginal independencies using constructive criteria that lead to well-structured enumeration procedures. The central idea underlying our results is that faithful models for marginal independencies are better described by posets than by DAGs. Our results allow to quantify how much our uncertainty about a causal structure is reduced when we invoke the faithfulness assumption and observe a set of marginal independencies.

It future work, it would be interesting to extend our approach to small (instead of empty) conditioning sets, which would cover cases where we only wish to perform CI tests with low dimensionality.

References

W. P. Bergsma. Testing conditional independence for continuous random variables. Technical Report 2004-049, EURANDOM, 2004.

A. Brandstädt, J. P. Spinrad, et al. *Graph classes: a survey*, volume 3. Siam, 1999.

G. Brinkmann and B. D. McKay. Posets on up to 16 points. *Order*, 19(2):147–179, 2002.

G. A. Cheston and T. Jap. A survey of the algorithmic properties of simplicial, upper bound and middle graphs. *Journal of Graph Algorithms and Applications*, 10(2): 159–190, 2006.

D. M. Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2003.

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

D. R. Cox and N. Wermuth. Linear dependencies represented by chain graphs. *Statistical Science*, 8(3):204–283, 1993.

G. Doran, K. Muandet, K. Zhang, and B. Schölkopf. A permutation-based kernel conditional independence test. In *Proceedings of UAI 2014*, pages 132–141, 2014.

M. Drton and T. S. Richardson. A new algorithm for maximum likelihood estimation in gaussian graphical models for marginal independence. In *Proceedings of UAI 2003*, pages 184–191, 2003.

M. Drton and T. S. Richardson. Graphical methods for efficient likelihood inference in gaussian covariance models. *Journal of Machine Learning Research*, 9:893–914, 2008a.

M. Drton and T. S. Richardson. Binary models for marginal independence. *Journal of the Royal Statistical Society, Ser. B*, 70(2):287–309, 2008b.

A. Idelberger. Generating causal diagrams from stochastic dependencies (in German). Master's thesis, Universität zu Lübeck, Germany, 2014.

M. Kalisch and P. Bühlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, 8:613–636, 2007.

R. M. Karp. *Reducibility among combinatorial problems*. Springer, 1972.

T. Kloks, D. Kratsch, and H. Müller. Finding and counting small induced subgraphs efficiently. *Information Processing Letters*, 74:115–121, 2000.

F. McMorris and T. Zaslavsky. Bound graphs of a partially ordered set. *Journal of Combinatorics, Information & System Sciences*, 7:134–138, 1982. ISSN 0250-9628; 0976-3473/e.

J. Pearl and T. Verma. The logic of representing dependencies by directed graphs. In *Proceedings of AAAI 1987 – Volume 1*, pages 374–379. AAAI Press, 1987.

J. Pearl and N. Wermuth. *When Can Association Graphs Admit A Causal Interpretation?*, volume 89 of *Lecture Notes in Statistics*, pages 205–214. Springer, 1994.

J. M. Peña. Reading dependencies from covariance graphs. *International Journal of Approximate Reasoning*, 54(1): 216–227, 2013.

T. S. Richardson. Markov properties for acyclic directed mixed graphs. *The Scandinavian Journal of Statistics*, 30(1):145–157, 2003.

M. Skowrońska and M. M. Sysło. An algorithm to recognize a middle graph. *Discrete Applied Mathematics*, 7 (2):201–208, 1984. ISSN 0166-218X.

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

K. M. Tan, P. London, K. Mohan, S.-I. Lee, M. Fazel, and D. Witten. Learning Graphical Models With Hubs. *Journal of Machine Learning Research*, 15:3297–3331, Oct 2014.

J. Textor and M. Liśkiewicz. Adjustment criteria in causal diagrams: An algorithmic perspective. In *Proceedings of UAI 2011*, pages 681–688. AUAI Press, 2011.

B. van der Zander, M. Liśkiewicz, and J. Textor. Constructing separators and adjustment sets in ancestral graphs. In *Proceedings of UAI 2014*, pages 907–916, 2014.

T. Verma and J. Pearl. Deciding morality of graphs is NP-complete. In *Proceedings of UAI 1993*, pages 391–399, 1993.

G. Wambach. The On-Line Encyclopedia of Integer Sequences. http://oeis.org/A007776, 2015. Number of connected posets with n elements of height 1. Accessed in March 2015.

E. S. Wolk. A note on the comparability graph of a tree. *Proceedings of the American Mathematical Society*, 16: 17–20, 1965.

K. Zhang, J. Peters, D. Janzing, and B. Schölkopf. Kernel-based conditional independence test and application in causal discovery. In *Proceedings of UAI 2011*, pages 804–8013, 2011.