Journal of Machine Learning Research 13 (2012) 2409-2464 Submitted 4/11; Revised 2/12; Published 8/12

Characterization and Greedy Learning of Interventional Markov Equivalence Classes of Directed Acyclic Graphs

Alain Hauser HAUSER@STAT.MATH.ETHZ.CH
Peter Bühlmann BUHLMANN@STAT.MATH.ETHZ.CH
Seminar für Statistik
ETH Zürich
8092 Zürich, Switzerland

Editor: Max Chickering

Abstract

The investigation of directed acyclic graphs (DAGs) encoding the same Markov property, that is the same conditional independence relations of multivariate observational distributions, has a long tradition; many algorithms exist for model selection and structure learning in Markov equivalence classes. In this paper, we extend the notion of Markov equivalence of DAGs to the case of interventional distributions arising from *multiple* intervention experiments. We show that under reasonable assumptions on the intervention experiments, interventional Markov equivalence defines a finer partitioning of DAGs than observational Markov equivalence and hence improves the identifiability of causal models. We give a graph theoretic criterion for two DAGs being Markov equivalent under interventions and show that each interventional Markov equivalence class can, analogously to the observational case, be uniquely represented by a chain graph called *interventional essential graph* (also known as *CPDAG* in the observational case). These are key insights for deriving a generalization of the Greedy Equivalence Search algorithm aimed at structure learning from interventional data. This new algorithm is evaluated in a simulation study.

Keywords: causal inference, interventions, graphical model, Markov equivalence, greedy equivalence search

1. Introduction

Directed acyclic graphs (or DAGs for short) are commonly used to model causal relationships between random variables; in such models, parents of some vertex in the graph are understood as "causes", and edges have the meaning of "causal influences". The causal influences between random variables imply conditional independence relations among them. However, those independence relations, or the corresponding Markov properties, do *not* identify the corresponding DAG completely, but only up to Markov equivalence. To put it simple, the skeleton of an underlying DAG is completely determined by its Markov property, whereas the *direction* of the arrows (which is crucial for causal interpretation) is in general not encoded in the Markov property for the observational distribution.

Interventions can help to overcome those limitations in identifiability. An *intervention* is realized by forcing the value of one or several random variables of the system to chosen values, destroying their original causal dependencies. The ensemble of both the observational and interventional distributions can greatly improve the identifiability of the causal structure of the system, the underlying DAG.

This paper has two main contributions. The first one is an algorithmically tractable graphical representation of Markov equivalence classes under a given set of interventions (possibly affecting several variables) from which the identifiability of causal models can be read off. This is of general interest for computation and algorithms dealing with structure (DAG) learning from an ensemble of observational and interventional data such as MCMC. The second contribution is a generalization of the Greedy Equivalence Search (GES) algorithm of Chickering (2002b), yielding an algorithm called Greedy Interventional Equivalence Search (GIES) which can be used for regularized maximum likelihood estimation in such an interventional setting.

In Section 2, we establish a criterion for two DAGs being Markov equivalent under a given intervention setting. We then generalize the concept of essential graphs, a graph theoretic representation of Markov equivalence classes, to the interventional case and characterize the properties of those graphs in Section 3. In Section 4, we elaborate a set of algorithmic operations to efficiently traverse the search space of interventional essential graphs and finally present the GIES algorithm. An experimental evaluation thereof is given in Section 5. We postpone all proofs to Appendix B, while Appendix A contains a review on graph theoretic concepts and definitions. An implementation of the GIES algorithm will be available in the next release of the R package `pcalg` (Kalisch et al., 2012); meanwhile, a prerelease version is available upon request from the first author.

1.1 Related Work

The investigation of Markov equivalence classes of directed graphical models has a long tradition, perhaps starting with the criterion for two DAGs being Markov equivalent by Verma and Pearl (1990) and culminating in the graph theoretic characterization of essential graphs (also called *CPDAGs*, "completed partially directed acyclic graphs") representing Markov equivalence classes by Andersson et al. (1997). Several algorithms for estimating essential graphs from observational data exist, such as the PC algorithm (Spirtes et al., 2000) or the Greedy Equivalence Search (GES) algorithm (Meek, 1997; Chickering, 2002b); a more complete overview is given in Brown et al. (2005) and Murphy (2001).

Different approaches to incorporate interventional data for learning causal models have been developed in the past. The Bayesian procedures of Cooper and Yoo (1999) or Eaton and Murphy (2007) address the problem of calculating a posterior (and also a likelihood) of an ensemble of observational and interventional data but do not address questions of identifiability or Markov equivalence: allowing different posteriors for Markov equivalent models can be intended in Bayesian methods (and realized by giving the corresponding models different priors). Since the number of DAGs with p variables grows super-exponentially with p (Robinson, 1973), the computation of a *full* posterior is intractable. For this reason, the mentioned Bayesian approaches are limited to computing posterior probabilities for certain features of a DAG; such a feature could be an edge from a vertex a to another vertex b, or a directed path from a to b visiting additional vertices. Approaches based on active learning (He and Geng, 2008; Tong and Koller, 2001; Eberhardt, 2008) propose an iterative line of action, estimating the essential graph with observational data in a first step and using interventional data in a second step to orient beforehand unorientable edges. He and Geng (2008) present a greedy procedure in which interventional data is uniquely used for deciding about edge orientations; this is not favorable from a statistical point of view since interventional data can also help to improve the estimation of the skeleton (or, more generally, the observational essential graph). Tong and Koller (2001) avoid this problem by using a Bayesian framework, but do not address the issue of Markov equivalence therewith. Eberhardt et al. (2005) and Eberhardt (2008) provide algorithms for choosing intervention targets that *completely* identify *all* causal models of p variables uniformly, but neither address the question of partial identifiability under a limited number of interventions nor provide an algorithm for learning the causal structure from data. Eberhardt et al. (2010) present an algorithm for learning *cyclic* linear causal models, but focus on complete identifiability; identifiability results for cyclic models only imply *sufficient*, but not *necessary*, conditions for the identifiability of acyclic models.

Probably the most advanced result concerning identifiability of causal models under single-variable interventions so far is given in the work of Tian and Pearl (2001). Although they do not provide a characterization of equivalence classes as a whole (as this paper does), they present a necessary and sufficient graph theoretic criterion for two models being indistinguishable under a set of single-variable interventions as well as a learning algorithm based on the detection of changes in marginal distributions.

2. Model

We consider p random variables $(X_1, \ldots, X_p) =: X$ which take values in some product measure space $(\mathcal{X}, \mathcal{A}, \mu) = (\prod_{i=1}^{p} \mathcal{X}_i, \bigotimes_{i=1}^{p} \mathcal{A}_i, \bigotimes_{i=1}^{p} \mu_i)$ with $\mathcal{X}_i \subset \mathbb{R}\ \forall\, i$. Each σ-algebra $\mathcal{A}_i$ is assumed to contain at least two disjoint sets of positive measure to avoid pathologies, and X is assumed to have a strictly positive joint density w.r.t. the measure μ on $\mathcal{X}$. We denote the set of all positive densities on $\mathcal{X}$ by $\mathcal{M}$. For any subset of component indices $A \subset [p] := \{1, \ldots, p\}$, we use the notation $\mathcal{X}_A := \prod_{a \in A} \mathcal{X}_a$, $X_A := (X_a)_{a \in A}$ and the convention $X_\emptyset \equiv 0$. Lowercase symbols like x_A represent a value in $\mathcal{X}_A$.

The model we are considering is built upon Markov properties with respect to DAGs. By convention, all graphs appearing in the paper shall have the vertex set $[p]$, representing the p random variables $X_1, \ldots, X_p$. Our notation and definitions related to graphs are summarized in Appendix A.1.

2.1 Causal Calculus: A Short Review

We start by summarizing important facts and fixing our notation concerning Markov properties and intervention calculus.

Definition 1 (Markov property; Lauritzen, 1996) *Let D be a DAG. Then we say that a probability density $f \in \mathcal{M}$ **obeys the Markov property of** D if $f(x) = \prod_{i=1}^{p} f(x_i | x_{\mathrm{pa}_D(i)})$. The set of all positive densities obeying the Markov property of D is denoted by $\mathcal{M}(D)$.*

Definition 1 is the most straightforward translation of independence relations induced from structural equations, the historical origin of directed graphical models (Wright, 1921). Related notions like local and global Markov properties exist and are equivalent to the factorization property of Definition 1 for positive densities (Lauritzen, 1996).

Definition 2 (Markov equivalence; Andersson et al., 1997) *Let D_1 and D_2 be two DAGs. D_1 and D_2 are called **Markov equivalent** (notation: $D_1 \sim D_2$) if $\mathcal{M}(D_1) = \mathcal{M}(D_2)$.*

Theorem 3 (Verma and Pearl, 1990) *Two DAGs D_1 and D_2 are Markov-equivalent if and only if they have the same skeleton and the same v-structures.*

Directed graphical models allow for an obvious causal interpretation. For a density f that obeys the Markov properties of some DAG D, we can think of a random variable X_a being the *direct cause* of another variable X_b if a is a parent of b in D.

Definition 4 (Causal model) *A* ***causal model*** *is a pair (D, f), where D is a DAG on the vertex set $[p]$ and $f \in \mathcal{M}(D)$ is a density obeying the Markov property of D: D is called the* ***causal structure*** *of the model, and f the* ***observational density****.*

Causality is strongly linked to interventions. We consider **stochastic interventions** (Korb et al., 2004) modeling the effect of setting or forcing one or several random variables X_I, where $I \subset [p]$ is called the **intervention target**, to the value of *independent* random variables U_I, called **intervention variables**. The joint product density of U_I on $\mathcal{X}_I$, called **level density**, is denoted by $\tilde{f}$. Extending the do() operator (Pearl, 1995) to stochastic interventions, we denote the density of X under such an intervention by $f(x|\operatorname{do}_D(X_I = U_I))$. Using truncated factorization and the assumption of independent intervention variables, this **interventional density** can be written as

$$f(x \mid \operatorname{do}_D(X_I = U_I)) = \prod_{i \notin I} f(x_i|x_{\mathrm{pa}_D(i)}) \prod_{i \in I} \tilde{f}(x_i) \; . \qquad (1)$$

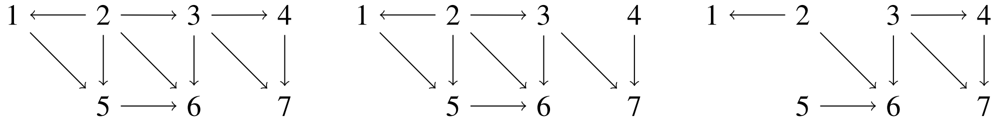
By denoting with $I = \emptyset$ and using the convention $f(x|\operatorname{do}(X_\emptyset = U_\emptyset)) = f(x)$, we also encompass the observational case as an intervention target.

Definition 5 (Intervention graph) *Let $D = ([p], E)$ be a DAG with vertex set $[p]$ and edge set E (see Appendix A.1), and $I \subset [p]$ an intervention target. The* ***intervention graph*** *of D is the DAG $D^{(I)} = ([p], E^{(I)})$, where $E^{(I)} := \{(a,b) \mid (a,b) \in E, b \notin I\}$.*

For a causal model (D, f), an interventional density $f(\cdot|\operatorname{do}_D(X_I = U_I))$ obeys the Markov property of $D^{(I)}$: the Markov property of the observational density is inherited. Figure 1 shows an example of a DAG and two corresponding intervention graphs.

As foreshadowed in the introduction, we are interested in causal inference based on data sets originating from *multiple* interventions, that means from a set of the form $\mathcal{S} = \{(I_j, \tilde{f}_j)\}_{j=1}^J$, where $I_j \subset [p]$ is an intervention target and $\tilde{f}_j$ a level density on $\mathcal{X}_{I_j}$ for $1 \leq j \leq J$. We call such a set an **intervention setting**, and the corresponding (multi)set of intervention targets $\mathcal{I} = \{I_j\}_{j=1}^J$ a **family of targets**. We often use the family of targets as an index set, for example to write a corresponding intervention setting as $\mathcal{S} = \{(I, \tilde{f}_I)\}_{I \in \mathcal{I}}$.

We consider **interventional data** of sample size n produced by a causal model (D, f) under an intervention setting $\mathcal{S} = \{(I, \tilde{f}_I)\}_{I \in \mathcal{I}}$. We assume that the n samples $X^{(1)}, \ldots, X^{(n)}$ are independent, and write them as usual as rows of a **data matrix X**. However, they are *not* identically distributed



(a) D (b) $D^{(\{4\})}$ (c) $D^{(\{3,5\})}$

Figure 1: A DAG D and the corresponding intervention graphs $D^{(\{4\})}$ and $D^{(\{3,5\})}$.

as they arise from *different* interventions. The interventional data set is fully specified by the pair $(\mathcal{T}, \mathbf{X})$,

$$\mathcal{T} = \begin{pmatrix} T^{(1)} \\ \vdots \\ T^{(n)} \end{pmatrix} \in \mathcal{I}^n, \quad \mathbf{X} = \begin{pmatrix} —X^{(1)}— \\ \vdots \\ —X^{(n)}— \end{pmatrix} , \tag{2}$$

where for each $i \in [n]$, $T^{(i)}$ denotes the intervention target under which the sample $X^{(i)}$ was produced. This data set can potentially contain observational data as well, namely if $\emptyset \in \mathcal{I}$. To summarize, we consider the statistical model

$$\begin{aligned} &X^{(1)}, X^{(2)}, \ldots, X^{(n)} \text{ independent,} \\ &X^{(i)} \sim f\big(\cdot \mid \mathrm{do}_D(X^{(i)}_{T^{(i)}} = U_{T^{(i)}})\big),\ U_{T^{(i)}} \sim \tilde{f}_{T^{(i)}}, \quad i = 1, \ldots, n , \end{aligned} \tag{3}$$

and we assume that each target $I \in \mathcal{I}$ appears at least once in the sequence $\mathcal{T}$.

2.2 Interventional Markov Equivalence: New Concepts and Results

An intervention at some target $a \in [p]$ destroys the original causal influence of other variables of the system on X_a. Interventional data thereof can hence not be used to determine the causal parents of X_a in the (undisturbed) system. To be able to estimate at least the complete skeleton of a causal structure (as in the observational case), an intervention experiment has to be performed based on a *conservative* family of targets:

Definition 6 (Conservative family of targets) *A family of targets $\mathcal{I}$ is called* ***conservative*** *if for all $a \in [p]$, there is some $I \in \mathcal{I}$ such that $a \notin I$.*

In this paper, we restrict our considerations to *conservative* families of targets; see Section 2.3 for a more detailed discussion. Note that every experiment in which we also measure observational data corresponds to a conservative family of targets.

If a family of targets $\mathcal{I}$ contains more than one target, interventional data as in Equation (3) are *not* identically distributed. Whereas the distribution of observational data is determined by a *single* density, we need *tuples* of densities as in the following definition to specify the distribution of interventional data.

Definition 7 *Let D be a DAG on $[p]$, and let $\mathcal{I}$ be a family of targets. Then we define*

$$\begin{aligned} \mathcal{M}_{\mathcal{I}}(D) := \big\{ (f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}^{|\mathcal{I}|} \,\big|\, &\forall I \in \mathcal{I} : f^{(I)} \in \mathcal{M}(D^{(I)}), \textit{ and} \\ &\forall I, J \in \mathcal{I}, \forall a \notin I \cup J : f^{(I)}(x_a | x_{\mathrm{pa}_D(a)}) = f^{(J)}(x_a | x_{\mathrm{pa}_D(a)}) \big\} . \end{aligned}$$

Although the do() operator does not appear in Definition 7, the elements in $\mathcal{M}_{\mathcal{I}}(D)$ are exactly the tuples $(f(\cdot | \mathrm{do}_D(X_I = U_I)))_{I \in \mathcal{I}}$ that can be realized as interventional densities of some causal model (D, f). The first condition in the definition reflects the fact that an intervention at a target I generates a density obeying the Markov property of $D^{(I)}$; the second condition is a consequence of the truncated factorization in Equation (1). These considerations are formalized in the following lemma and motivate Definition 9 of interventional Markov equivalence in analogy to the observational case. Note that for $\mathcal{I} = \{\emptyset\}$, Definition 7 equals its observational counterpart: $\mathcal{M}_{\{\emptyset\}}(D) = \mathcal{M}(D)$ (see Definition 1).

Lemma 8 *Let D be a DAG on $[p]$, and $\mathcal{I}$ a conservative family of targets.*

(i) Let (D, f) be a causal model (that is, $f \in \mathcal{M}(D)$), $\mathcal{S} = \{(I, \tilde{f}_I)\}_{I \in \mathcal{I}}$ an intervention setting and $U_I \sim \tilde{f}_I$ intervention variables for $I \in \mathcal{I}$. Then, we have

$$\big(f(\cdot \mid \mathrm{do}(X_I = U_I))\big)_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(D) \ .$$

(ii) Let $(f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(D)$. Then there is some positive density $f \in \mathcal{M}(D)$ and an intervention setting $\mathcal{S} = \{(I, \tilde{f}_I)\}_{I \in \mathcal{I}}$ such that $f(\cdot | \mathrm{do}(X_I = U_I)) = f^{(I)}(\cdot)$ for random variables U_I with density $\tilde{f}_I$, for all $I \in \mathcal{I}$.

Definition 9 (Interventional Markov equivalence) *Let D_1 and D_2 be DAGs, and $\mathcal{I}$ a family of targets. D_1 and D_2 are called **$\mathcal{I}$-Markov equivalent** (notation: $D_1 \sim_{\mathcal{I}} D_2$) if $\mathcal{M}_{\mathcal{I}}(D_1) = \mathcal{M}_{\mathcal{I}}(D_2)$. The $\mathcal{I}$-Markov equivalence class of a DAG D is denoted by $[D]_{\mathcal{I}}$.*

Alternatively, we will also use the term “interventionally Markov equivalent” when it is clear which family of targets is meant. For the simplest conservative family of targets, $\mathcal{I} = \{\emptyset\}$, we get back Definition 2 for the observational case. We now generalize Theorem 3 for the interventional case in order to get a purely graph theoretic criterion for interventional Markov equivalence of two given DAGs, the main result of this section.

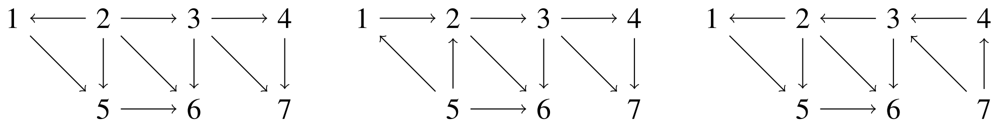
Theorem 10 *Let D_1 and D_2 be two DAGs on $[p]$, and $\mathcal{I}$ a conservative family of targets. Then, the following statements are equivalent:*

(i) $D_1 \sim_{\mathcal{I}} D_2$;
(ii) for all $I \in \mathcal{I}$, $D_1^{(I)} \sim D_2^{(I)}$ (in the observational sense);
(iii) for all $I \in \mathcal{I}$, $D_1^{(I)}$ and $D_2^{(I)}$ have the same skeleton and the same v-structures;
(iv) D_1 and D_2 have the same skeleton and the same v-structures, and $D_1^{(I)}$ and $D_2^{(I)}$ have the same skeleton for all $I \in \mathcal{I}$.

2.3 Discussion

Throughout this paper, we always assume the observational density f of a causal model to be strictly positive. This assumption makes sure that the conditional densities in Equation (1) are well-defined. The requirement of a strictly positive density can, however, be a restriction for example for discrete models (where the density is with respect to the counting measure). In the observational case, the notion of Markov equivalence remains the same when we also allow densities that are not strictly positive (Lauritzen, 1996). We conjecture that the notion of interventional Markov equivalence (Definition 9 and Theorem 10) also remains valid for such densities; corresponding proofs would, however, require more caution to avoid the aforementioned problems with (truncated) factorization.

To illustrate the importance of a conservative family of targets for structure identification, let us consider the simplest non-trivial example of a causal model with 2 variables X_1 and X_2. Under observational data, we can distinguish two Markov equivalence classes: one in which the variables are independent (represented by the empty DAG D_0), and one in which they are not independent (represented by the DAGs $D_1 := 1 \longrightarrow 2$ and $D_2 := 1 \longleftarrow 2$). D_1 and D_2 can be distinguished if we can measure data from an intervention at one of the vertices in addition to observational data; this experimental setting corresponds to the (conservative) family of targets $\mathcal{I} = \{\emptyset, \{1\}\}$. However, an



(a) D (b) D_1 (c) D_2

Figure 2: Three DAGs having equal skeletons and a single v-structure, $3 \longrightarrow 6 \longleftarrow 5$, hence being observationally Markov equivalent. For $\mathcal{I} = \{\emptyset, \{4\}\}$, we have $D \sim_{\mathcal{I}} D_1$, but $D \not\sim_{\mathcal{I}} D_2$ since the skeletons of $D^{(\{4\})}$ (Figure 1(b)) and $D_2^{(\{4\})}$ do not coincide.

intervention at, say, X_1 *alone* (that is, in the *absence* of observational data), corresponding to the non-conservative family $\mathcal{I} = \{\{1\}\}$, only allows a distinction between the models D_2 and D_0 on the one hand (which do not show dependence between X_1 and X_2 under the intervention) and D_1 on the other hand (which does show dependence between X_1 and X_2 under the intervention). Note that the two indistinguishable models D_0 and D_2 do not even have the same skeleton, and that it is impossible to determine the influence of X_2 on X_1 in the undisturbed system. In this setting, it would be more natural to consider the intervened variable X_1 as an external parameter rather than a random variable of the system, and to perform regression to detect or determine the influence of X_1 on X_2. Note, however, that full identifiability of the models does *not* require observational data; interventions at X_1 and X_2 (corresponding to the conservative family $\mathcal{I} = \{\{1\}, \{2\}\}$ in our notation) are also sufficient.

Theorem 10 is of great importance for the description of Markov equivalence classes under interventions. It shows that two DAGs which are interventionally Markov equivalent under some conservative family of targets are also observationally Markov equivalent:

$$D_1 \sim_{\mathcal{I}} D_2 \Rightarrow D_1 \sim D_2. \tag{4}$$

This implication is *not* true anymore for non-conservative families of targets. This is an explanation for the term "conservative": a conservative family of targets yields a finer partitioning of DAGs into equivalence classes compared to observational Markov equivalence, but it preserves the "borders" of observational Markov equivalence classes. Figure 2 shows three DAGs that are observationally Markov equivalent, but which fall into two different interventional Markov equivalence classes under the family of targets $\mathcal{I} = \{\emptyset, \{4\}\}$.

Theorem 10 agrees with Theorem 3 of Tian and Pearl (2001) for single-variable interventions. While we also make a statement about interventions at *several* variables, they prove their theorem for perturbations of the system at single variables only, but for a wider class of perturbations called **mechanism changes** that go beyond our notion of interventions. While an *intervention* destroys the causal dependence of a variable from its parents (and hence replaces a conditional density by a marginal one in the Markov factorization, see Equation (1)), a *mechanism change* (also known as "imperfect" or "soft" interventions; see Eaton and Murphy, 2007) alters the functional form of this dependence (and hence replaces a Markov factor by a different one which is still a conditional distribution). The fact that Theorem 10 is true for mechanism changes on single variables motivates the conjecture that it also holds for mechanism changes on *several* variables.

3. Essential Graphs

Theorem 10 represents a computationally fast criterion for deciding whether two DAGs are interventionally Markov equivalent or not. However, given some DAG D, it does not provide a possibility for quickly finding *all* equivalent ones, and hence does not specify the equivalence class as a whole. In this section, we give a characterization of graphs that uniquely represent an interventional Markov equivalence class (Theorem 18). Our characterization of these *interventional essential graphs* is inspired by and similar to the one developed by Andersson et al. (1997) for the observational case and allows for handling equivalence classes algorithmically. Furthermore, we present a linear time algorithm for constructing a representative of the equivalence class corresponding to an interventional essential graph (Proposition 16 and discussion thereafter), as well as a polynomial time algorithm for constructing the interventional essential graph of a given DAG (Algorithm 1). Throughout this section, $\mathcal{I}$ always stands for a conservative family of targets.

3.1 Definitions and Motivation

All DAGs in an $\mathcal{I}$-Markov equivalence class share the same skeleton; however, arrow orientations may vary between different representatives (Theorem 10). Varying and common arrow orientations are represented by undirected and directed edges, respectively, in $\mathcal{I}$-essential graphs.

Definition 11 ($\mathcal{I}$-essential graph) *Let D be a DAG. The **$\mathcal{I}$-essential graph** of D is defined as $\mathcal{E}_{\mathcal{I}}(D) := \bigcup_{D' \in [D]_{\mathcal{I}}} D'$. (The union is meant in the graph theoretic sense, see Appendix A.1).*

When the family of targets $\mathcal{I}$ in question is clear from the context, we will also use the term **interventional essential graph**, while "observational essential graph" shall refer to the concept of essential graphs as introduced by Andersson et al. (1997) in the observational case. Simply speaking of "essential graphs", we mean interventional or observational essential graphs in the following.

Definition 12 ($\mathcal{I}$-essential arrow) *Let D be a DAG. An edge $a \longrightarrow b \in D$ is **$\mathcal{I}$-essential** in D if $a \longrightarrow b \in D' \; \forall \, D' \in [D]_{\mathcal{I}}$.*

An $\mathcal{I}$-essential graph typically contains directed as well as undirected edges. Directed ones correspond to arrows that are $\mathcal{I}$-essential in every representative of the equivalence class; in other words, $\mathcal{I}$-essential arrows are those whose direction is identifiable. A first sufficient criterion for an edge to be $\mathcal{I}$-essential follows immediately from Lemma 47 (Appendix B.1).

Corollary 13 *Let D be a DAG with $a \longrightarrow b \in D$. If there is an intervention target $I \in \mathcal{I}$ such that $|\{a, b\} \cap I| = 1$, then $a \longrightarrow b$ is $\mathcal{I}$-essential.*

The investigation of essential graphs has a long tradition in the observational case (Andersson et al., 1997; Chickering, 2002a). Due to increased identifiability of causal structures, Markov equivalence classes shrink in the interventional case; Equation (4) implies $\mathcal{E}_{\mathcal{I}}(D) \subset \mathcal{E}_{\{\emptyset\}}(D)$ for any conservative family of targets $\mathcal{I}$ (see also Figure 8 in Section 5). Essential graphs, interventional as well as observational ones, are mainly interesting because of two reasons:

- It is important to know which arrow directions of a causal model are identifiable and which are not since arrow directions are relevant for the causal interpretation.

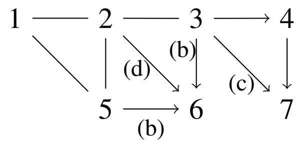
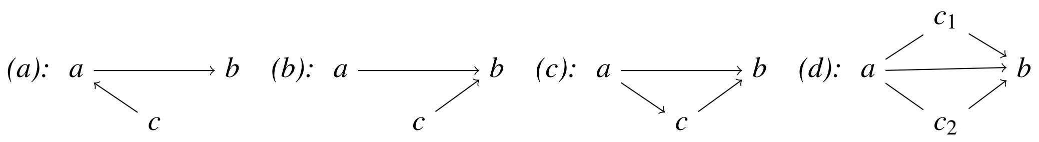


Figure 3: A graph with six arrows. Four of them are strongly $\mathcal{I}$-protected for any conservative family of targets $\mathcal{I}$ (in parentheses: arrow configurations according to Definition 14). Arrows $3 \longrightarrow 4$ and $4 \longrightarrow 7$ are strongly $\mathcal{I}$-protected for $\mathcal{I} = \{\emptyset, \{4\}\}$, but not for $\mathcal{I} = \{\emptyset\}$.

- Markov equivalent DAGs encode the same statistical model. Hence the space of DAGs is no suitable "parameter" or search space for statistical inference and computation. The natural search space is given by the set of the equivalence classes, the objects that can be distinguished from data. Essential graphs uniquely represent these equivalence classes and are efficiently manageable in algorithms.

The characterization of $\mathcal{I}$-essential graphs (Theorem 18) relies on the notion of strongly $\mathcal{I}$-protected arrows (Definition 14) which reproduces the corresponding definition of Andersson et al. (1997) for $\mathcal{I} = \{\emptyset\}$; an illustration is given in Figure 3.

Definition 14 (Strong protection) *Let G be a graph. An arrow $a \longrightarrow b \in G$ is **strongly $\mathcal{I}$-protected** in G if there is some $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$, or the arrow $a \longrightarrow b$ occurs in at least one of the following four configurations as an induced subgraph of G:*



We will see in Theorem 18 that every arrow of an $\mathcal{I}$-essential graph (that is, every edge corresponding to an $\mathcal{I}$-essential arrow in the representative DAGs) is strongly $\mathcal{I}$-protected. The configurations in Definition 14 guarantee the identifiability of the edge orientation between a and b: if there is a target $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$, turning the arrow would change the skeleton of the intervention graph $D^{(I)}$ (see also Corollary 13); in configuration (a), reversal would create a new v-structure; in (b), reversal would destroy a v-structure; in (c), reversal would create a cycle; an in (d) finally, at least one of the arrows between a and c_1 or c_2 must point away from a in each representative, hence turning the arrow $a \longrightarrow b$ would create a cycle. We refer to Andersson et al. (1997) for a more detailed discussion of the configurations (a) to (d).

3.2 Characterization of Interventional Essential Graphs

As in the observational setting, we can show that interventional essential graphs are chain graphs with chordal chain components (see Appendix A.1). For the observational case $\mathcal{I} = \{\emptyset\}$, Propositions 15 and 16 below correspond to Propositions 4.1 and 4.2 of Andersson et al. (1997).

Proposition 15 *Let D be a DAG on $[p]$. Then:*

(i) $\mathcal{E}_{\mathcal{I}}(D)$ is a chain graph.

(ii) For each chain component $T \in \mathbf{T}(\mathcal{E}_{\mathcal{I}}(D))$, the induced subgraph $\mathcal{E}_{\mathcal{I}}(D)[T]$ is chordal.

Proposition 16 *Let D be a DAG. A digraph D' is acyclic and $\mathcal{I}$-equivalent to D if and only if D' can be constructed by orienting the edges of every chain component of $\mathcal{E}_{\mathcal{I}}(D)$ according to a perfect elimination ordering.*

This proposition is not only of theoretic, but also of algorithmic interest. According to the explanation in Appendix A.2, perfect elimination orderings on the (chordal) chain components of $\mathcal{E}_{\mathcal{I}}(D)$ can be generated with LEXBFS (Algorithm 6); doing this for all chain components yields computational complexity $O(|E|+p)$, where E denotes the edge set of $\mathcal{E}_{\mathcal{I}}(D)$ (see Appendix A.2).

As an immediate consequence of Proposition 16, interventional essential graphs are in one-to-one correspondence with interventional Markov equivalence classes. We will therefore also speak about "representatives of $\mathcal{I}$-essential graphs", where we mean representatives (that is, DAGs) of the corresponding equivalence class. Propositions 15 and 16 give the justification for the following definition; note that in order to generate a representative of some $\mathcal{I}$-essential graph, the family of targets $\mathcal{I}$ need not be known.

Definition 17 *Let G be the $\mathcal{I}$-essential graph of some DAG. The set of representatives of G is denoted by $\mathbf{D}(G)$:*

$$\begin{aligned}\mathbf{D}(G) :=\{&D \text{ a DAG} \mid D \subset G, D^u = G^u, D[T] \text{ oriented according to some} \\ &\text{perfect elimination ordering for each chain component } T \in \mathbf{T}(G)\}.\end{aligned}$$

Here, D^u denotes the skeleton of D (Appendix A.1). We can now state the main result of this section, a graph theoretic characterization of $\mathcal{I}$-essential graphs. For the observational case $\mathcal{I} = \{\emptyset\}$, this theorem corresponds to Theorem 4.1 of Andersson et al. (1997).

Theorem 18 *A graph G is the $\mathcal{I}$-essential graph of a DAG D if and only if*

(i) G is a chain graph;
(ii) for each chain component $T \in \mathbf{T}(G)$, $G[T]$ is chordal;
(iii) G has no induced subgraph of the form $a \longrightarrow b \frac{\quad}{} c$;
(iv) G has no line $a \frac{\quad}{} b$ for which there exists some $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$;
(v) every arrow $a \longrightarrow b \in G$ is strongly $\mathcal{I}$-protected.

The graph G of Figure 3 satisfies points (i) to (iii) of Theorem 18. For $\mathcal{I} = \{\emptyset, \{4\}\}$, it also fulfills points (iv) and (v); in this case, it is the $\mathcal{I}$-essential graph $\mathcal{E}_{\mathcal{I}}(D)$ of the DAG D of Figure 1(a) by Proposition 16.

3.3 Construction of Interventional Essential Graphs

In this section, we show that there is a simple way to construct the $\mathcal{I}$-essential graph $\mathcal{E}_{\mathcal{I}}(D)$ of a DAG D: we need to successively convert arrows that are not strongly $\mathcal{I}$-protected into lines (Algorithm 1). By doing this, we get a sequence of partial $\mathcal{I}$-essential graphs.

Definition 19 (Partial $\mathcal{I}$-essential graph) *Let D be a DAG. A graph G with $D \subset G \subset \mathcal{E}_{\mathcal{I}}(D)$ is called a **partial $\mathcal{I}$-essential graph** of D if $a \longrightarrow b \frac{\quad}{} c$ does not occur as an induced subgraph of G.*

The following lemma can be understood as a motivation for looking at such graphs. Note that due to the condition $G \subset \mathcal{E}_{\mathcal{I}}(D)$, and because G and $\mathcal{E}_{\mathcal{I}}(D)$ have the same skeleton, every arrow of $\mathcal{E}_{\mathcal{I}}(D)$ is also present in G, hence statement (ii) below makes sense.

Lemma 20 *Let D be a DAG. Then:*

(i) D and $\mathcal{E}_{\mathcal{I}}(D)$ are partial $\mathcal{I}$-essential graphs of D.

(ii) Let G be a partial $\mathcal{I}$-essential graph of D. Every arrow $a \longrightarrow b \in \mathcal{E}_{\mathcal{I}}(D)$ is strongly $\mathcal{I}$-protected in G.

(iii) Let G be a partial $\mathcal{I}$-essential graph of two DAGs D_1 and D_2. Then, $D_1 \sim_{\mathcal{I}} D_2$.

Algorithm 1 constructs the $\mathcal{I}$-essential graph G from a partial $\mathcal{I}$-essential graph of any DAG $D \in \mathbf{D}(G)$. The algorithm is indeed valid and calculates $\mathcal{E}_{\mathcal{I}}(D)$, since the graph produced in each iteration is a partial $\mathcal{I}$-essential graph of D (Lemma 21), and the only partial $\mathcal{I}$-essential graph that has only strongly $\mathcal{I}$-protected arrows is $\mathcal{E}_{\mathcal{I}}(D)$ (Lemma 22).

Lemma 21 *Let D be a DAG and G a partial $\mathcal{I}$-essential graph of D. Assume that $a \longrightarrow b \in G$ is not strongly $\mathcal{I}$-protected in G, and let $G' := G + (b,a)$ (that is, the graph we get by replacing the arrow $a \longrightarrow b$ by a line $a \frac{\quad}{} b$; see Appendix A.1). Then G' is also a partial $\mathcal{I}$-essential graph of D.*

Lemma 22 *Let D be a DAG. There is exactly one partial $\mathcal{I}$-essential graph of D in which every arrow is strongly $\mathcal{I}$-protected, namely $\mathcal{E}_{\mathcal{I}}(D)$.*

To construct $\mathcal{E}_{\mathcal{I}}(D)$ from some DAG $D = ([p], E)$, we must, in the worst case, execute the iteration of Algorithm 1 for every arrow in the DAG; at each step, we must check every 4-tuple of vertices to see whether some arrow occurs in configuration (d) of Definition 14. Therefore Algorithm 1 has at most complexity $O(|E| \cdot p^4)$; by exploiting the partial order $\preceq_G$ on $\mathbf{T}(G)$ (see Appendix A.1), more efficient implementations are possible. Note that some checks only need to be done once. If, for example, an edge $a \longrightarrow b$ is part of a v-structure (configuration (b) of Definition 14), or if there is some $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$ in the first iteration of Algorithm 1, this will also be the case in every later iteration.

3.4 Example: Identifiability under Interventions

A simple example illustrates how much identifiability can be gained with a single intervention. We consider a linear chain as observational essential graph:

$$G = \mathcal{E}_{\{\emptyset\}}(D) : 1 \frac{\quad}{} 2 \frac{\quad}{} 3 \frac{\quad}{} \cdots \frac{\quad}{} p\ .$$

We can easily count the number of representatives of G using the following lemma.

```
Input  : $G$: partial $\mathcal{I}$-essential graph of some DAG $D$ (not known)
Output: $\mathcal{E}_{\mathcal{I}}(D)$
while $\exists\ a \longrightarrow b \in G$ s.t. $a \longrightarrow b$ not strongly $\mathcal{I}$-protected in $G$ do
    $G \leftarrow G + (b,a)$;
return $G$;
```

Algorithm 1: REPLACEUNPROTECTED$(\mathcal{I}, G)$. Iterative construction of an $\mathcal{I}$-essential graph

Lemma 23 (Source lemma) *Let G be a connected, chordal, undirected graph, and let $D \subset G$ be a DAG without v-structures and with $D^u = G$. Then D has exactly one source.*

Proof Let σ be a topological ordering of D; then, $\sigma(1)$ is a source, see Appendix A.1. It remains to show that there is at most one such source. Assume, for the sake of contradiction, that there are two different sources u and v. Since G is connected, there is a shortest u-v-path $\gamma = (a_0 \equiv u, a_1, \ldots, a_k \equiv v)$. Let $a_i \longleftarrow a_{i+1} \in D$ be the first arrow that points away from v in the chain γ in D (note $i \geq 1$ since $u \longrightarrow a_1 \in D$ by assumption). The v-structure $a_{i-1} \longrightarrow a_i \longleftarrow a_{i+1}$ is not allowed as an induced subgraph of D, hence a_{i-1} and a_{i+1} must be adjacent in D and in G; however, γ is then no *shortest* u-v-path, a contradiction. ■

For our linear chain G and any $s \in [p]$, there is exactly one DAG $D \in \mathbf{D}(G)$ that has the (unique) source s, namely the DAG we get by orienting *all* edges of G away from s; other edge orientations would produce a v-structure. We conclude G has p representatives.

Assume that the true causal model producing the data is (D, f), and denote the source of D by $s \in [p]$. Consider the conservative family of targets $\mathcal{I} = \{\emptyset, \{v\}\}$ with $v \in [p]$. If $v < s$, the interventional essential graph $\mathcal{E}_{\mathcal{I}}(D)$ is

$$1 \longleftarrow 2 \longleftarrow \ldots \longleftarrow v+1 \longrightarrow \ldots \longrightarrow p \; ,$$

and $|\mathbf{D}(\mathcal{E}_{\mathcal{I}}(D))| = p - v$ by the same arguments as above; analogously, if $v > s$, we find $|\mathbf{D}(\mathcal{E}_{\mathcal{I}}(D))| = v - 1$. On the other hand, if $v = s$, all edges of D are strongly $\mathcal{I}$-protected: those incident to s because of the intervention target, all others because they are in configuration (a) of Definition 14; therefore, we have $\mathcal{E}_{\mathcal{I}}(D) = D$.

In the best case, all edge orientations in the chain can be identified by a single intervention, while the observational essential graph $\mathcal{E}_{\{\emptyset\}}(D)$ that is identifiable from observational data alone contains p representatives. However, this needs an intervention at the a priori unknown source s. Choosing the central vertex $\lceil \frac{p}{2} \rceil$ as intervention target ensures that at least half of the edges become directed in $\mathcal{E}_{\mathcal{I}}(D)$, independent of the position s of the source.

4. Greedy Interventional Equivalence Search

Different algorithms have been proposed to estimate essential graphs from observational data. One of them, the Greedy Equivalence Search (GES) (Meek, 1997; Chickering, 2002b), is particularly interesting because of two properties:

- It is score-based; it greedily maximizes some score function for given data over essential graphs. It uses no tuning-parameter; the score function alone measures the quality of the estimate. Chickering (2002b) chose the BIC score because of consistency; technically, any score equivalent and decomposable function (see Definition 24) is adequate.

- It traverses the space of essential graphs which is the natural search space for model inference (see Section 3). We will see in Section 5 that a greedy search over *equivalence classes* yields much better estimation results than a naïve greedy search over *DAGs*.

GES greedily optimizes the score function in two phases (Chickering, 2002b):

- In the **forward phase**, the algorithm starts with the empty essential graph, $G_0 := ([p], \emptyset)$. It then sequentially steps from one essential graph G_i to a *larger* one, G_{i+1}, for which there are representatives $D_i \in \mathbf{D}(G_i)$ and $D_{i+1} \in \mathbf{D}(G_{i+1})$ such that D_{i+1} has exactly one arrow more than D_i.

- In the **backward phase**, the sequence $(G_i)_i$ is continued by gradually stepping from one essential graph G_i to a *smaller* one, G_{i+1}, for which there are representatives $D_i \in \mathbf{D}(G_i)$ and $D_{i+1} \in \mathbf{D}(G_{i+1})$ such that D_{i+1} has exactly one arrow less than D_i.

In both phases, the respective candidate with maximal score is chosen, or the phase is aborted if no candidate scores higher than the current essential graph G_i.

We introduce in addition a new turning phase which proved to enhance estimation (see Section 5). Here, the sequence $(G_i)_i$ is elongated by gradually stepping from one essential graph G_i to a new one with the same number of edges, denoted by G_{i+1}, for which there are representatives $D_i \in \mathbf{D}(G_i)$ and $D_{i+1} \in \mathbf{D}(G_{i+1})$ such that D_{i+1} can be constructed from D_i by turning exactly one arrow. As before, we choose the highest scoring candidate. Such a turning phase had already been proposed, but not characterized or implemented, by Chickering (2002b).

Because GES is an optimization algorithm working on the space of observational essential graphs, and because the characterization of *interventional* essential graphs is similar to that of observational ones (Theorem 18), GES can indeed be generalized to handle interventional data as well by operating on interventional instead of observational essential graphs. We call this generalized algorithm *Greedy Interventional Equivalence Search* or GIES. An overview is shown in Algorithm 2: the forward, backward and turning phase are repeatedly executed in this order until none of them can augment the score function any more.

A naïve search strategy would perhaps traverse the space of DAGs instead of essential graphs, greedily adding, removing or turning single arrows from DAGs. It is well-known in the observational case that such an approach performs markedly worse than one accounting for Markov equivalence (Chickering, 2002b; Castelo and Kočka, 2003), and we will see in our simulations (Section 5.2) that the same is true in the interventional case as long as few interventions are made. Ignoring Markov equivalence cuts down the search space of successors at haphazard; since all DAGs in a Markov equivalence class represent the same statistical model, there is no justification for considering neighbors (that is, DAGs that can be reached by adding, removing or turning an arrow) of one of the representatives but not of the other ones.

GIES can be used with general score functions. It goes without saying that the chosen score function should be a "reasonable" one which has favorable statistical properties such as consistency. We denote the score of a DAG D given interventional data $(\mathcal{T}, \mathbf{X})$ by $S(D; \mathcal{T}, \mathbf{X})$, and we assume that S is **score equivalent**, that is, it assigns the same score to $\mathcal{I}$-equivalent DAGs; $\mathcal{I}$ always stands for a conservative family of targets in this section. Furthermore, we require S to be decomposable.

Definition 24 *A score function S is called* ***decomposable*** *if for each DAG D, S can be written as a sum*

$$S(D; \mathcal{T}, \mathbf{X}) = \sum_{i=1}^{p} s(i, \mathrm{pa}_D(i); \mathcal{T}, \mathbf{X}),$$

where the ***local score*** *s depends on* $\mathbf{X}$ *only via* $\mathbf{X}_{\bullet i}$ *and* $\mathbf{X}_{\bullet\, \mathrm{pa}_D(i)}$*, with* $\mathbf{X}_{\bullet i}$ *denoting the* i^{th} *column of* $\mathbf{X}$ *and* $\mathbf{X}_{\bullet\, \mathrm{pa}_D(i)}$ *the submatrix of* $\mathbf{X}$ *corresponding to the columns with index in* $\mathrm{pa}_D(i)$.

Throughout the rest of this section, S always denotes a score equivalent and decomposable score function. Such a score function needs only be evaluated at one single representative of some interventional Markov equivalence class. Indeed, a key ingredient for the efficiency of the observational GES as well as our interventional GIES is an implementation that computes the greedy steps to the next equivalence class in a local fashion without enumerating all corresponding DAG members. Chickering (2002b) found a clever way to do that in the forward and backward phase of the observational GES. In Sections 4.1 and 4.2, we generalize his methods to the interventional case, and in Section 4.3, we propose an efficient implementation of the new turning phase.

4.1 Forward Phase

A step in the forward phase of GIES can be formalized as follows: for an $\mathcal{I}$-essential graph G_i, find the next one $G_{i+1} := \mathcal{E}_{\mathcal{I}}(D_{i+1})$, where

$$\begin{aligned} D_{i+1} &:= \underset{D' \in \mathbf{D}^+(G_i)}{\arg\max}\, S(D'; \mathcal{T}, \mathbf{X}), \text{ and} \\ \mathbf{D}^+(G_i) &:= \{D' \text{ a DAG} \mid \exists \text{ an arrow } u \longrightarrow v \in D' : D' - (u,v) \in \mathbf{D}(G_i)\} . \end{aligned}$$

If no candidate DAG $D' \in \mathbf{D}^+(G_i)$ scores higher than G_i, abort the forward phase.

We denote the set of candidate *$\mathcal{I}$-essential graphs* by $\boldsymbol{\mathcal{E}}^+_{\mathcal{I}}(G_i) := \{\mathcal{E}_{\mathcal{I}}(D') \mid D' \in \mathbf{D}^+(G_i)\}$. In the next proposition, we show that each graph $G' \in \boldsymbol{\mathcal{E}}^+_{\mathcal{I}}(G_i)$ can be characterized by a triple (u, v, C), where $u \longrightarrow v$ is the arrow that has to be added to a representative D of G_i in order to get a representative D' of G', and C specifies the edge orientations of D within the chain component of v in G.

```
Input  : $(\mathcal{T}, \mathbf{X})$: interventional data for family of targets $\mathcal{I}$
Output: $\mathcal{I}$-essential graph
$G \leftarrow ([p], \emptyset)$;
repeat
    DoContinue ← FALSE;
    repeat
        $G_{\text{old}} \leftarrow G$;
        $G \leftarrow$ FORWARDSTEP$(G; \mathcal{T}, \mathbf{X})$ ;                    // See Algorithm 3
    until $G_{\text{old}} = G$;
    repeat
        $G_{\text{old}} \leftarrow G$;
        $G \leftarrow$ BACKWARDSTEP$(G; \mathcal{T}, \mathbf{X})$ ;                   // See Algorithm 4
        if $G_{\text{old}} \neq G$ then DoContinue ← TRUE;
    until $G_{\text{old}} = G$;
    repeat
        $G_{\text{old}} \leftarrow G$;
        $G \leftarrow$ TURNINGSTEP$(G; \mathcal{T}, \mathbf{X})$ ;                    // See Algorithm 5
        if $G_{\text{old}} \neq G$ then DoContinue ← TRUE;
    until $G_{\text{old}} = G$;
until ¬DoContinue;
```

Algorithm 2: GIES$(\mathcal{T}, \mathbf{X})$. Greedy Interventional Equivalence Search. The steps of the different phases of the algorithms are described in Algorithms 3–5.

Proposition 25 *Let G be an $\mathcal{I}$-essential graph, let u and v be two non-adjacent vertices of G, and let $C \subset \mathrm{ne}_G(v)$. Then there is a DAG $D \in \mathbf{D}(G)$ with $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D\} = C$ such that $D' := D + (u,v) \in \mathbf{D}^+(G)$ if and only if*

(i) C is a clique in $G[T_G(v)]$;
(ii) $N := \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u) \subset C$;
(iii) and every path from v to u in G has a vertex in C.

For given G, u, v and C determine D' uniquely up to $\mathcal{I}$-equivalence.

Note that points (i) and (ii) imply in particular that N is a clique in $G[T_G(v)]$. Proposition 25 has already been proven for the case of observational data (Chickering, 2002b, Theorem 15); it is not obvious, however, to see that this characterization of a forward step is also valid for interventional essential graphs, so we give a new proof in Appendix B.3 using the results developed in Sections 2 and 3.

The DAGs D and D' in Proposition 25 only differ in the edge (u,v); v is the only vertex whose parents are different in D and D'. Since the score function S is assumed to be decomposable, the score difference between D and D' can be expressed by the local score change at vertex v, as stated in the following corollary.

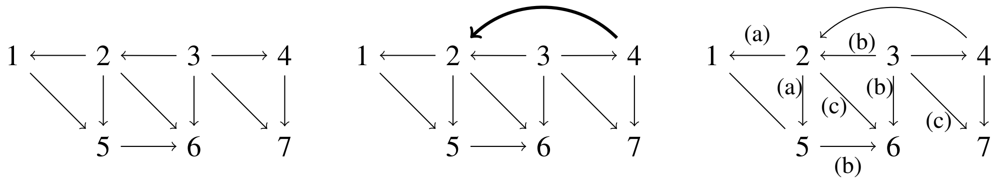
Corollary 26 *Let G, u, v, C, D and D' be as in Proposition 25. The score difference $\Delta S := S(D'; \mathcal{T}, \mathbf{X}) - S(D; \mathcal{T}, \mathbf{X})$ can be calculated as follows:*

$$\Delta S = s(v, \mathrm{pa}_G(v) \cup C \cup \{u\}; \mathcal{T}, \mathbf{X}) - s(v, \mathrm{pa}_G(v) \cup C; \mathcal{T}, \mathbf{X}).$$

In the observational case, this corollary corresponds to Corollary 16 of Chickering (2002b).

```
Input  : $G = ([p], E)$: $\mathcal{I}$-essential graph; $(\mathcal{T}, \mathbf{X})$: interventional data for $\mathcal{I}$
Output: $G' \in \boldsymbol{\mathcal{E}}^+_{\mathcal{I}}(G)$, or $G$
$\Delta S_{\max} \leftarrow 0$;
foreach $v \in [p]$ do
    foreach $u \in [p] \setminus \mathrm{ad}_G(v)$ do
        $N \leftarrow \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u)$;
        foreach clique $C \subset \mathrm{ne}_G(v)$ with $N \subset C$ do          // Proposition 25(i) and (ii)
            if $\nexists$ path from $v$ to $u$ in $G[[p] \setminus C]$ then          // Proposition 25(iii)
                $\Delta S \leftarrow s(v, \mathrm{pa}_G(v) \cup C \cup \{u\}; \mathcal{T}, \mathbf{X}) - s(v, \mathrm{pa}_G(v) \cup C; \mathcal{T}, \mathbf{X})$;
                if $\Delta S > \Delta S_{\max}$ then
                    $\Delta S_{\max} \leftarrow \Delta S$;
                    $(u_{\max}, v_{\max}, C_{\max}) \leftarrow (u, v, C)$;
if $\Delta S_{\max} > 0$ then
    $\sigma \leftarrow$ LexBFS$((C_{\max}, v_{\max}, \ldots), E[T_G(v_{\max})])$;
    Orient edges of $G[T_G(v_{\max})]$ according to $\sigma$;
    Insert edge $(u_{\max}, v_{\max})$ into $G$;
    return ReplaceUnprotected$(\mathcal{I}, G)$ ;                              // See Algorithm 1
else return $G$;
```

Algorithm 3: FORWARDSTEP$(G; \mathcal{T}, \mathbf{X})$. One step of the forward phase of GIES.



(a) D (b) D' (c) $\mathcal{E}_{\mathcal{I}}(D')$

Figure 4: DAGs D, D' and $\mathcal{E}_{\mathcal{I}}(D')$ illustrating a possible forward step of GIES for the family of targets $\mathcal{I} = \{\emptyset, \{4\}\}$, applied to the $\mathcal{I}$-essential graph G of Figure 3 for the parameters $(u,v,C) = (4,2,\{3\})$ (notation according to Proposition 25). In parentheses in Figure (c): arrow configurations according to Definition 14; arrows incident to 4 are strongly $\mathcal{I}$-protected by the intervention target $\{4\}$.

The most straightforward way to construct an $\mathcal{I}$-essential graph $G' \in \boldsymbol{\mathcal{E}}^+_{\mathcal{I}}(G)$ characterized by the triple (u,v,C) as defined in Proposition 25 would be to create a representative $D \in \mathbf{D}(G)$ by orienting the edges of $T_G(v)$ as indicated by the set C, add the arrow $u \longrightarrow v$ to get D', and finally construct $\mathcal{E}_{\mathcal{I}}(D')$ with Algorithm 1. The next lemma suggests a novel shortcut to this procedure: it is sufficient to orient the edges of the chain component $T_G(v)$ *only* to get a partial $\mathcal{I}$-essential graph of D' after adding the arrow $u \longrightarrow v$.

Lemma 27 *Let G, u, v, C, D and D' be as in Proposition 25. Let H be the graph that we get by orienting all edges of $T_G(v)$ as in D (leaving other chain components unchanged) and inserting the arrow (u,v). Then H is a partial $\mathcal{I}$-essential graph of D'.*

Algorithm 3 shows our implementation of the forward phase of GIES, summarizing the results of Proposition 25, Corollary 26 and Lemma 27. Figure 4 illustrates one forward step, applied to the $\mathcal{I}$-essential graph G (for $\mathcal{I} = \{\emptyset, \{4\}\}$) of Figure 3 and characterized by the triple $(u,v,C) = (4,2,\{3\})$. Note that this triple is indeed valid in the sense of Proposition 25: $\{3\}$ is clearly a clique (point (i)), $\mathrm{ne}_G(2) \cap \mathrm{ad}_G(4) = \{3\}$ (point (ii)), and there is no path from 2 to 4 in $G[[p] \setminus C]$ (point (iii)).

4.2 Backward Phase

In analogy to the forward phase, one step of the backward phase can be formalized as follows: for an $\mathcal{I}$-essential graph G_i, find its successor $G_{i+1} := \mathcal{E}_{\mathcal{I}}(D_{i+1})$, where

$$D_{i+1} := \underset{D' \in \mathbf{D}^-(G_i)}{\arg\max}\, S(D'; \mathbf{X}), \text{ and}$$

$$\mathbf{D}^-(G_i) := \{D' \text{ a DAG} \mid \exists\, D \in \mathbf{D}(G_i), u \longrightarrow v \in D : D' = D - (u,v)\} .$$

If no candidate DAG $D' \in \mathbf{D}^+(G_i)$ scores higher than G_i, the backward phase is aborted.

Whenever we have some representative $D \in \mathbf{D}(G)$ of an $\mathcal{I}$-essential graph G, we get a DAG in $\mathbf{D}^-(G)$ by removing any arrow of D. This is in contrast to the forward phase where we do not necessarily get a DAG in $\mathbf{D}^+(G)$ by adding an arbitrary arrow to D. By adding arrows, new directed cycles could be created, something which is not possible by removing arrows. This is the reason why the backward phase is generally simpler to implement than the forward phase.

In Proposition 28 (corresponding to Theorem 17 of Chickering (2002b) for the observational case), we show that we can, similarly to the forward phase, characterize an $\mathcal{I}$-essential graph of $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{-}(G) := \{\mathcal{E}_{\mathcal{I}}(D') \mid D' \in \mathbf{D}^{-}(G)\}$ by a triple (u,v,C), where C is a clique in $\mathrm{ne}_G(v)$. As in the forward phase, we see that the score difference of D and D' is determined by the local score change at the vertex v (Corollary 29), and that lines in chain components other than $T_G(v)$ remain lines in $G' = \mathcal{E}_{\mathcal{I}}(D')$ (Lemma 30). Algorithm 4 summarizes the results of the propositions in this section.

Proposition 28 *Let $G = ([p],E)$ be an $\mathcal{I}$-essential graph with $(u,v) \in E$ (that is, $u - v \in G$ or $u \longrightarrow v \in G$), and let $C \subset \mathrm{ne}_G(v)$. There is a DAG $D \in \mathbf{D}(G)$ with $u \longrightarrow v \in D$ and $\{a \in \mathrm{ne}_G(v) \setminus \{u\} \mid a \longrightarrow v \in D\} = C$ such that $D' := D - (u,v) \in \mathbf{D}^{-}(G)$ if and only if*

(i) C is a clique in $G[T_G(v)]$;
(ii) $C \subset N := \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u)$.

Moreover, u, v and C determine D' uniquely up to $\mathcal{I}$-equivalence for a given G.

Corollary 29 *Let G, u, v, C, D and D' be as in Proposition 28. The score difference $\Delta S := S(D';\mathcal{T},\mathbf{X}) - S(D;\mathcal{T},\mathbf{X})$ is:*

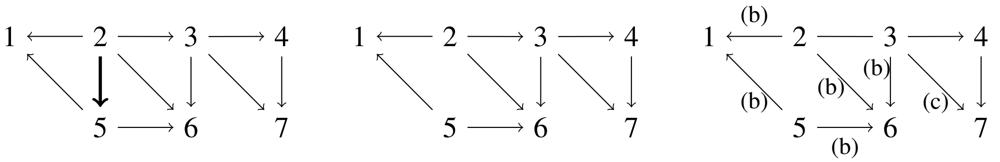
$$\Delta S = s(v,(\mathrm{pa}_G(v) \cup C) \setminus \{u\};\mathcal{T},\mathbf{X}) - s(v,\mathrm{pa}_G(v) \cup C \cup \{u\};\mathcal{T},\mathbf{X}).$$

In the observational case, this corresponds to Corollary 18 in Chickering (2002b). The analogue to Lemma 27 for a computational shortcut in the forward phase reads as follows:

Lemma 30 *Let G, u, v, C, D and D' be as in Proposition 28. Let H be the graph that we get by orienting all edges of $T_G(v)$ as in D and removing the arrow (u,v). Then H is a partial $\mathcal{I}$-essential graph of D'.*

```
Input  : $G = ([p],E)$: $\mathcal{I}$-essential graph; $(\mathcal{T},\mathbf{X})$: interventional data for $\mathcal{I}$
Output: $G' \in \boldsymbol{\mathcal{E}}_{\mathcal{I}}^{-}(G)$, or $G$
$\Delta S_{\max} \leftarrow 0$;
foreach $v \in [p]$ do
    foreach $u \in \mathrm{ne}_G(v) \cup \mathrm{pa}_G(v)$ do
        $N \leftarrow \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u)$;
        foreach clique $C \subset N$ do
            $\Delta S \leftarrow s(v,(\mathrm{pa}_G(v) \cup C) \setminus \{u\};\mathcal{T},\mathbf{X}) - s(v,\mathrm{pa}_G(v) \cup C \cup \{u\};\mathcal{T},\mathbf{X})$;
            if $\Delta S > \Delta S_{\max}$ then
                $\Delta S_{\max} \leftarrow \Delta S$;
                $(u_{\max}, v_{\max}, C_{\max}) \leftarrow (u,v,C)$;
if $\Delta S_{\max} > 0$ then
    if $u_{\max} \in \mathrm{ne}_G(v_{\max})$ then $\sigma \leftarrow$ LEXBFS$((C_{\max}, u_{\max}, v_{\max}, \ldots), E[T_G(v_{\max})])$;
    else $\sigma \leftarrow$ LEXBFS$((C_{\max}, v_{\max}, \ldots), E[T_G(v_{\max})])$;
    Orient edges of $G[T_G(v_{\max})]$ according to $\sigma$;
    Remove edge $(u_{\max}, v_{\max})$ from $G$;
    return REPLACEUNPROTECTED$(\mathcal{I}, G)$ ;                // See Algorithm 1
else return G;
```

Algorithm 4: BACKWARDSTEP$(G;\mathcal{T},\mathbf{X})$. One step of the backward phase of GIES.



(a) D (b) D' (c) $\mathcal{E}_{\mathcal{I}}(D')$

Figure 5: DAGs D, D' and $\mathcal{E}_{\mathcal{I}}(D')$ illustrating a possible backward step of GIES for the family of targets $\mathcal{I} = \{\emptyset, \{4\}\}$, applied to the $\mathcal{I}$-essential graph G of Figure 3 for the parameters $(u,v,C) = (2,5,\emptyset)$ (notation according to Proposition 28). Figure (c), in parentheses: arrow configurations according to Definition 14.

A backward step of GIES is summarized in Algorithm 4 and illustrated in Figure 5. The triple $(u,v,C) = (2,5,\emptyset)$ used there to characterize the backward step obviously fulfills the requirements of Proposition 28.

4.3 Turning Phase

Finally, we characterize a step of the turning phase of GIES, in which we want to find the successor $G_{i+1} := \mathcal{E}_{\mathcal{I}}(D_{i+1})$ for an $\mathcal{I}$-essential graph G_i by the rule

$$\begin{aligned} D_{i+1} &:= \operatorname*{arg\,max}_{D' \in \mathbf{D}^{\circlearrowleft}(G_i)} S(D'; \mathcal{T}, \mathbf{X}), \text{ where} \\ \mathbf{D}^{\circlearrowleft}(G_i) &:= \{D' \text{ a DAG} \mid D' \notin \mathbf{D}(G_i), \text{ and } \exists \text{ an arrow } u \longrightarrow v \in D' : \\ &\qquad D' - (u,v) + (v,u) \in \mathbf{D}(G_i)\} \ . \end{aligned}$$

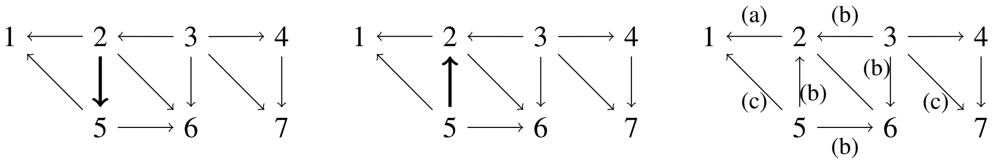
When the score cannot be augmented anymore, the turning phase is aborted. The additional condition "$D' \notin \mathbf{D}(G_i)$" is not necessary in the definitions of $\mathbf{D}^+(G_i)$ and $\mathbf{D}^-(G_i)$; when adding or removing an arrow from a DAG, the skeleton changes, hence the new DAG is certainly not $\mathcal{I}$-equivalent to the previous one. However, when *turning* an arrow, the skeleton remains the same, and the danger of staying in the same equivalence class exists.

Again, we are looking for an efficient method to find a representative D' for each $G' \in \boldsymbol{\mathcal{E}}_{\mathcal{I}}^{\circlearrowleft}(G_i) := \{\mathcal{E}_{\mathcal{I}}(D') \mid D' \in \mathbf{D}^{\circlearrowleft}(G_i)\}$. It makes sense to distinguish whether the arrow that should be turned in a representative $D \in \mathbf{D}(G_i)$ is $\mathcal{I}$-essential or not. We start with the case where we want to turn an arrow which is *not* $\mathcal{I}$-essential.

Proposition 31 *Let G be an $\mathcal{I}$-essential graph with $u - v \in G$, and let $C \subset \mathrm{ne}_G(v) \setminus \{u\}$. Define $N := \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u)$. Then there is a DAG $D \in \mathbf{D}(G)$ with $u \longleftarrow v \in D$ and $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D\} = C$ such that $D' := D - (v,u) + (u,v) \in \mathbf{D}^{\circlearrowleft}(G)$ if and only if*

(i) C is a clique in $G[T_G(v)]$;
(ii) $C \setminus N \neq \emptyset$;
(iii) $C \cap N$ separates $C \setminus N$ and $N \setminus C$ in $G[\mathrm{ne}_G(v)]$.

For a given G, u, v and C determine D' up to $\mathcal{I}$-equivalence.



(a) D

(b) D'

(c) $\mathcal{E}_{\mathcal{I}}(D')$

Figure 6: DAGs D, D' and $\mathcal{E}_{\mathcal{I}}(D')$ illustrating a possible turning step of GIES applied to the $\mathcal{I}$-essential graph G ($\mathcal{I} = \{\emptyset, \{4\}\}$) of Figure 3 for the parameters $(u,v,C) = (5,2,\{3\})$ (notation of Proposition 31). The arrow $2 \longrightarrow 5$ is not $\mathcal{I}$-essential in D. Figure (c): arrow configurations in parentheses, see Definition 14.

There are now *two* vertices that have different parents in the DAGs D and D', namely u and v; thus the calculation of the score difference between D and D' involves two local scores instead of one.

Corollary 32 *Let G, u, v, C, D and D' be as in Proposition 31. Then the score difference $\Delta S := S(D'; \mathcal{T}, \mathbf{X}) - S(D; \mathcal{T}, \mathbf{X})$ can be calculated as follows:*

$$\begin{aligned}\Delta S = {} & s(v, \mathrm{pa}_G(v) \cup C \cup \{u\}; \mathcal{T}, \mathbf{X}) + s(u, \mathrm{pa}_G(u) \cup (C \cap N); \mathcal{T}, \mathbf{X}) \\ & - s(v, \mathrm{pa}_G(v) \cup C; \mathcal{T}, \mathbf{X}) - s(u, \mathrm{pa}_G(u) \cup (C \cap N) \cup \{v\}; \mathcal{T}, \mathbf{X}).\end{aligned}$$

Lemma 33 *Let G, u, v, C, D and D' be as in Proposition 31. Let H be the graph that we get by orienting all edges of $T_G(v)$ as in D and turning the arrow (v,u). Then H is a partial $\mathcal{I}$-essential graph of D'.*

A possible turning step is illustrated in Figure 6, where a non-$\mathcal{I}$-essential arrow (for $\mathcal{I} = \{\emptyset, \{4\}\}$) of a representative of the graph G of Figure 3 is turned. The step is characterized by the triple $(u,v,C) = (5,2,\{3\})$ which satisfies the conditions of Proposition 31: $\{3\}$ is obviously a clique (point (i)), $C \setminus N = C$ since $N = \{1\}$ (point (ii)), and $C \setminus N = \{3\}$ and $N \setminus C = \{1\}$ are separated in $G[\mathrm{ne}_G(2)]$ (point (iii)). In contrast, the triple $(u,v,C) = (5,2,\{1\})$ fulfills points (i) and (iii) of Proposition 31, but not point (ii). There is a DAG $D \in \mathbf{D}(G)$ with $\{a \in \mathrm{ne}_G(2) \mid a \longrightarrow 2 \in D\} = \{1\}$, and turning the arrow $2 \longrightarrow 5$ in D yields another DAG D' (that is, does not create a new cycle). This new DAG D', however, is $\mathcal{I}$-equivalent to D, and hence not a member of $\mathbf{D}^{\circlearrowleft}(G)$ (see the discussion above).

We now proceed to the case where an $\mathcal{I}$-essential arrow of a representative of G is turned; here there is no danger to remain in the same Markov equivalence class. The characterization of this case is similar to the forward phase.

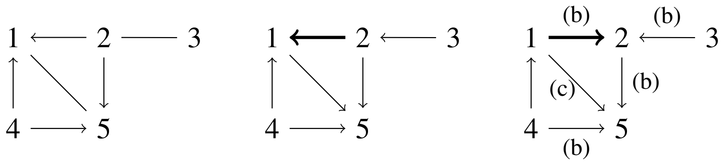
Proposition 34 *Let G be an $\mathcal{I}$-essential graph with $u \longleftarrow v \in G$, and let $C \subset \mathrm{ne}_G(v)$. Then there is a DAG $D \in \mathbf{D}(G)$ with $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D\} = C$ such that $D' := D - (v,u) + (u,v) \in \mathbf{D}^{\circlearrowleft}(G)$ if and only if*

(i) C is a clique;

(ii) $N := \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u) \subset C$;

(iii) every path from v to u in G except (v,u) has a vertex in $C \cup \mathrm{ne}_G(u)$.

Moreover, u, v and C determine D' up to $\mathcal{I}$-equivalence.



(a) G (b) D (c) $D' = \mathcal{E}_\mathcal{I}(D')$

Figure 7: Graphs G, D, D' and $\mathcal{E}_\mathcal{I}(D')$ illustrating a possible turning step of GIES for the family of targets $\mathcal{I} = \{\emptyset, \{4\}\}$ and the parameters $(u, v, C) = (1, 2, \{3\})$ (notation of Proposition 34). The arrow $2 \longrightarrow 1$ is $\mathcal{I}$-essential in D. Figure (c): arrow configurations in parentheses, see Definition 14.

Chickering (2002a) has already proposed a turning step for essential arrows in the observational case; however, he did not provide necessary and sufficient conditions specifying all possible turning steps as Proposition 34 does.

Lemma 35 *Let G, u, v, C, D and D' be as in Proposition 34, and let H be the graph that we get by orienting all edges of $T_G(v)$ and $T_G(u)$ as in D and by turning the edge (v, u). Then H is a partial $\mathcal{I}$-essential graph of D'.*

To construct a $G' \in \boldsymbol{\mathcal{E}}_\mathcal{I}^{\circlearrowleft}(G)$ out of G, we must possibly orient *two* chain components of G instead of one (Lemma 35). In the example of Figure 7, we see that it is indeed not sufficient to orient the edges of $T_G(v)$ alone in order to get a partial $\mathcal{I}$-essential graph of G'. The arrow $1 \longrightarrow 5$ is not $\mathcal{I}$-essential in D, hence $5 \in T_G(1)$. However, the same arrow is $\mathcal{I}$-essential in D' and hence also present in $\mathcal{E}_\mathcal{I}(D')$.

Despite the fact that we need to orient the edges of $T_G(v)$ and $T_G(u)$ to get a partial $\mathcal{I}$-essential graph of D', $\mathcal{E}_\mathcal{I}(D')$ is nevertheless determined by the orientation of edges adjacent to v (determined by the clique C) alone. This comes from the fact that in D, defined as in Proposition 34, all arrows of $D[T_G(u)]$ must point away from u.

Corollary 36 *Let G, u, v, C, D and D' be as in Proposition 34. Then the score difference $\Delta S := S(D'; \mathcal{T}, \mathbf{X}) - S(D; \mathcal{T}, \mathbf{X})$ can be calculated as follows:*

$$\begin{aligned}\Delta S &= s(v, \mathrm{pa}_G(v) \cup C \cup \{u\}; \mathcal{T}, \mathbf{X}) + s(u, \mathrm{pa}_G(u) \setminus \{v\}; \mathcal{T}, \mathbf{X}) \\ &\quad - s(v, \mathrm{pa}_G(v) \cup C; \mathcal{T}, \mathbf{X}) - s(u, \mathrm{pa}_G(u); \mathcal{T}, \mathbf{X}).\end{aligned}$$

The entire turning step, for essential and non-essential arrows, is shown in Algorithm 5.

4.4 Discussion

Every step in the forward, backward and turning phase of GIES is characterized by a triple (u, v, C), where u and v are different vertices and C is a clique in the neighborhood of v. To identify the highest scoring movement from one $\mathcal{I}$-essential graph G to a potential successor in $\boldsymbol{\mathcal{E}}_\mathcal{I}^{+}(G)$, $\boldsymbol{\mathcal{E}}_\mathcal{I}^{-}(G)$ or $\boldsymbol{\mathcal{E}}_\mathcal{I}^{\circlearrowleft}(G)$, respectively, one potentially has to examine all cliques in the neighborhood $\mathrm{ne}_G(v)$ of all vertices $v \in [p]$. The time complexity of any (forward, backward or turning) step applied to an $\mathcal{I}$-essential graph G hence highly depends on the size of the largest clique in the chain components of G. By restricting GIES to $\mathcal{I}$-essential graphs with a bounded vertex degree, the time complexity of a step of GIES is polynomial in p; otherwise, it is in the worst case exponential. We believe, however, that GIES is in practice much more efficient than this worst-case complexity suggests. Some evidence for this claim is provided by the runtime analysis of our simulation study, see Section 5.2.

A heuristic approach to guarantee polynomial runtime of a greedy search has been proposed by Castelo and Kočka (2003) for the observational case. Their Hill Climber Monte Carlo (HCMC) algorithm operates in DAG space, but to account for Markov equivalence, the neighborhood of a number of randomly chosen DAGs equivalent to the current one is scanned in each greedy step.

```
Input  : $G = ([p], E)$: $\mathcal{I}$-essential graph; $(\mathcal{T}, \mathbf{X})$: interventional data for $\mathcal{I}$
Output: $G' \in \boldsymbol{\mathcal{E}}_{\mathcal{I}}^{\circlearrowleft}$, or $G$
$\Delta S_{\max} \leftarrow 0$;
foreach $v \in [p]$ do
    foreach $u \in \mathrm{ne}_G(v)$ do     // Consider arrows that are not $\mathcal{I}$-essential for turning
        $N \leftarrow \mathrm{ne}_G(u) \cap \mathrm{ad}_G(v)$;
        foreach clique $C \subset \mathrm{ne}_G(v) \setminus \{u\}$ do                    // Proposition 31(i)
            if $C \setminus N \neq \emptyset$ and $\{u, v\}$ separates $C$ and $N \setminus C$ in $G[T_G(v)]$ then
                                                             // Proposition 31(ii) and (iii)
                $\Delta S \leftarrow s(v, \mathrm{pa}_G(v) \cup C \cup \{u\}; \mathcal{T}, \mathbf{X}) + s(u, \mathrm{pa}_G(u) \cup (C \cap N); \mathcal{T}, \mathbf{X})$;
                $\Delta S \leftarrow \Delta S - s(v, \mathrm{pa}_G(v) \cup C; \mathcal{T}, \mathbf{X}) - s(u, \mathrm{pa}_G(u) \cup (C \cap N) \cup \{v\}; \mathcal{T}, \mathbf{X})$;
                if $\Delta S > \Delta S_{\max}$ then
                    $\Delta S_{\max} \leftarrow \Delta S$;
                    $(u_{\max}, v_{\max}, C_{\max}) \leftarrow (u, v, C)$;
    foreach $u \in \mathrm{ch}_G(v)$ do                 // Consider $\mathcal{I}$-essential arrows for turning
        $N \leftarrow \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u)$;
        foreach clique $C \subset \mathrm{ne}_G(v)$ with $N \subset C$ do             // Proposition 34(i) and (ii)
            if $\nexists$ path from $v$ to $u$ in $G[[p] \setminus (C \cup \mathrm{ne}_G(u))] - (v, u)$ then    // Proposition 34(iii)
                $\Delta S \leftarrow s(v, \mathrm{pa}_G(v) \cup C \cup \{u\}; \mathcal{T}, \mathbf{X}) + s(u, \mathrm{pa}_G(u) \setminus \{v\}; \mathcal{T}, \mathbf{X})$;
                $\Delta S \leftarrow \Delta S - s(v, \mathrm{pa}_G(v) \cup C; \mathcal{T}, \mathbf{X}) - s(u, \mathrm{pa}_G(u); \mathcal{T}, \mathbf{X})$;
                if $\Delta S > \Delta S_{\max}$ then
                    $\Delta S_{\max} \leftarrow \Delta S$;
                    $(u_{\max}, v_{\max}, C_{\max}) \leftarrow (u, v, C)$;
if $\Delta S_{\max} > 0$ then
    if $v_{\max} \longrightarrow u_{\max} \in G$ then
        $\sigma_u := \mathrm{LexBFS}((u_{\max}, \ldots), E[T_G(u_{\max})])$;
        Orient edges of $G[T_G(u_{\max})]$ according to $\sigma$;
        $\sigma_v := \mathrm{LexBFS}((C_{\max}, v_{\max}, \ldots), E[T_G(v_{\max})])$;
    else $\sigma_v := \mathrm{LexBFS}((C_{\max}, v_{\max}, u_{\max}, \ldots), E[T_G(v_{\max})])$;
    Orient edges of $G[T_G(v_{\max})]$ according to $\sigma_v$;
    Turn edge $(v_{\max}, u_{\max})$ in $G$;
    return $\mathrm{ReplaceUnprotected}(\mathcal{I}, G)$ ;                          // See Algorithm 1
else return $G$;
```

Algorithm 5: $\mathrm{TurningStep}(G; \mathcal{T}, \mathbf{X})$. One step of the turning phase of GIES.

The equivalence class of the current DAG is explored by randomly turning "covered arrows", that is, arrows whose reversal does not change the Markov property. In our (interventional) notation, an arrow is covered if and only if it is *not* strongly $\mathcal{I}$-protected (Definition 14). By limiting the number of covered arrow reversals, a polynomial runtime is guaranteed at the cost of potentially lowering the probability of investigating a particular successor in $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{+}(G)$, $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{-}(G)$ or $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{\circlearrowleft}(G)$, respectively. HCMC hence enables a fine tuning of the trade-off between exploration of the search space and runtime, or between greediness and randomness.

The order of executing the backward and the turning phase seems somewhat arbitrary. In the analysis of the steps performed by GIES in our simulation study (Section 5.2), we saw that the turning phase can generally only augment the score when very few backward steps were executed before. For this reason, we believe that changing the order of the backward and the turning phase would have little effect on the overall performance of GIES.

As already discussed by Chickering (2002b) for the observational case, caching techniques can markedly speed up GES; the same holds for GIES. The basic idea is the following: in a forward step, the algorithm evaluates a lot of triples (u, v, C) to choose the best one, $(u_{\max}, v_{\max}, C_{\max})$ (lines 1 to 9 in Algorithm 3). After performing the forward move corresponding to $(u_{\max}, v_{\max}, C_{\max})$, many of the triples evaluated in the step before are still valid candidates for next step in the sense of Proposition 25 and lead to the same score difference as before (see Corollary 26). Caching those values avoids unnecessary reevaluation of possible forward steps. The same holds for the backward and the turning phase; since the forward step is most frequently executed, a caching strategy in this phase yields the highest speed-up though.

We emphasize that the characterization of "neighboring" $\mathcal{I}$-essential graphs in $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{+}(G)$, $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{-}(G)$ or $\boldsymbol{\mathcal{E}}_{\mathcal{I}}^{\circlearrowleft}(G)$, respectively, by triples (u, v, C) is of more general interest for structure learning algorithms, for example for the design of sampling steps of an MCMC algorithm. Also the beforementioned HCMC algorithm could be extended to interventional data by generalizing the notion of "covered arcs" using Definition 14.

The prime example of a score equivalent and decomposable score function is the Bayesian information criterion (BIC) (Schwarz, 1978) which we used in our simulations (Section 5). It penalizes the complexity of causal models by their number of free parameters (ℓ_0 penalization); this number is the sum of free parameters of the conditional densities in the Markov factorization (Definition 1), which explains the decomposability of the score. Using different penalties, for example, ℓ_2 penalization, can lead to a non-decomposable score function. GIES can also be adapted to such score functions; the calculation of score differences becomes computationally more expensive in this case since it cannot be done in a local fashion as in Corollaries 26, 29, 32 and 36.

GIES only relies on the notion of interventional Markov equivalence, and on a score function that can be evaluated for a given class of causal models. As we mentioned in Section 2.1, we believe that interventional Markov equivalence classes remain unchanged for models that do not have a strictly positive density. For this reason it should be safe to also apply GIES to such a model class.

5. Experimental Evaluation

We evaluated the GIES algorithm on simulated interventional data (Section 5.2) and on *in silico* gene expression data sets taken from the DREAM4 challenge (Marbach et al., 2010) (Section 5.3).

In both cases, we restricted our considerations to *Gaussian* causal models as summarized in Section 5.1.

5.1 Gaussian Causal Models

Consider a causal model (D, f) with a Gaussian density of the form $\mathcal{N}(0, \Sigma)$. The observational Markov property of such a model translates to a set of *linear* structural equations

$$X_i = \sum_{j=1}^{p} \beta_{ij} X_j + \varepsilon_i,\ \varepsilon_i \overset{\text{indep.}}{\sim} \mathcal{N}(0, \sigma_i^2), \quad 1 \leq i \leq p\ , \tag{5}$$

where $\beta_{ij} = 0$ if $j \notin \mathrm{pa}_D(i)$. When the DAG structure D is known, the covariance matrix Σ can be parameterized by the **weight matrix**

$$B := (\beta_{ij})_{i,j=1}^{p} \in \mathbf{B}(D) := \{A = (\alpha_{ij}) \in \mathbb{R}^{p \times p} \mid \alpha_{ij} = 0 \text{ if } j \notin \mathrm{pa}_D(i)\}$$

that assigns a weight β_{ij} to each arrow $j \longrightarrow i \in D$, and the vector of error covariances $\sigma^2 := (\sigma_1^2, \ldots, \sigma_p^2)$:

$$\Sigma = \mathrm{Cov}(X) = (\mathbb{1} - B)^{-1} \operatorname{diag}(\sigma^2)(\mathbb{1} - B)^{-\mathrm{T}}\ .$$

This is a consequence of Equation (5).

We always assume Gaussian intervention variables U_I (see Section 2.1). In this case, not only the observational density f is Gaussian, but also the interventional densities $f(x \mid \mathrm{do}_D(X_I = U_I))$. An interventional data set $(\mathcal{T}, \mathbf{X})$ as defined in Equation (2) then consists of n independent, but not identically distributed Gaussian samples.

We use the **Bayesian information criterion** (BIC) as score function for GIES:

$$S(D; \mathcal{T}, \mathbf{X}) := \sup\{\ell_D(B, \sigma^2; \mathcal{T}, \mathbf{X}) \mid B \in \mathbf{B}(D), \sigma^2 \in \mathbb{R}_{>0}^{p}\} - \frac{k_D}{2} \log(n)\ ,$$

where ℓ_D denotes the log-likelihood of the density in Equation (3):

$$\begin{aligned} \ell_D(B, \sigma^2; \mathcal{T}, \mathbf{X}) \quad &:= \quad \sum_{i=1}^{n} \log f\big(X^{(i)} \mid \mathrm{do}_D(X^{(i)}_{T^{(i)}} = U_{T^{(i)}})\big) \qquad (6) \\ &= \quad \sum_{i=1}^{n} \Big[\sum_{j \notin T^{(i)}} \log f(X_j^{(i)} \mid X^{(i)}_{\mathrm{pa}_D(j)}) + \sum_{j \in T^{(i)}} \log \tilde{f}(X_j^{(i)}) \Big] \\ &= \quad -\frac{1}{2} \sum_{i=1}^{n} \sum_{j \notin T^{(i)}} \Big[\log \sigma_j^2 + \frac{1}{\sigma_j^2} \Big(X_j^{(i)} - B_{j\bullet} X^{(i)} \Big)^2 \Big] + C \\ &= \quad -\frac{1}{2} \sum_{j=1}^{p} \Big[|\{i \mid j \notin T^{(i)}\}| \log \sigma_j^2 + \frac{1}{\sigma_j^2} \sum_{i : j \notin T^{(i)}} \Big(X_j^{(i)} - B_{j\bullet} X^{(i)} \Big)^2 \Big] + C\ , \end{aligned}$$

where the constant C is independent of the parameters (B, σ^2) of the model. Since Gaussian causal models with structure D are parameterized by $B \in \mathbf{B}(D)$ and $\sigma^2 \in \mathbb{R}^p_{>0}$, we have $k_D = p + |E|$ free parameters, where E denotes the edge set of D. It can be seen in Equation (6) that the **maximum likelihood estimator** (MLE) $(\hat{B}, \hat{\sigma}^2)$, the maximizer of ℓ_D, minimizes the residual sum of squares for the different structural equations; for more details we refer to Hauser and Bühlmann (2012).

The DAG $\hat{D}$ maximizing the BIC yields a *consistent* estimator for the true causal structure D in the sense that $P[\hat{D} \sim_{\mathcal{I}} D] \rightarrow 1$ in the limit $n \rightarrow \infty$ as long as the true density f is **faithful** with respect to D, that is, *every* conditional independence relation of f is encoded in the Markov property of D (Hauser and Bühlmann, 2012). Note that the BIC score is even defined in the high-dimensional setting $p > n$; however, we only consider low-dimensional settings here.

5.2 Simulations

We simulated interventional data from 4000 randomly generated Gaussian causal models as described in Section 5.2.1. In Sections 5.2.2 and 5.2.3, we present our methods for evaluating GIES; the results are discussed in Section 5.2.4. As a rough summary, GIES markedly beat the conceptually simpler greedy search over the space of DAGs as well as the original GES of Chickering (2002b) ignoring the interventional nature of the simulated data sets. Its learning performance could keep up with a provably consistent exponential time dynamic programming algorithm at much lower computational cost.

5.2.1 GENERATION OF GAUSSIAN CAUSAL MODELS

For some number p of vertices, we randomly generated Gaussian causal models parameterized by a structure D, a weight matrix $B \in \mathbf{B}(D)$ and a vector of error covariances $\sigma^2 \in \mathbb{R}^p_{>0}$ by a procedure slightly adapted from Kalisch and Bühlmann (2007):

1. For a given **sparseness parameter** $s \in (0,1)$, draw a DAG D with topological ordering $(1,\ldots,p)$ and binomially distributed vertex degrees with mean $s(p-1)$.

2. Shuffle the vertex indices of D to get a random topological ordering.

3. For each arrow $j \longrightarrow i \in D$, draw $\beta'_{ij} \sim \mathcal{U}([-1,-0.1] \cup [0.1,1])$ using independent realizations; for other pairs of (i,j), set $\beta'_{ij} = 0$ (see Equation (5)). This yields a weight matrix $B' = (\beta'_{ij})_{i,j=1}^p \in \mathbf{B}(D)$ with positive as well as negative entries which are bounded away from 0.

4. Draw error variances $\sigma'^2_i \overset{\text{i.i.d.}}{\sim} \mathcal{U}([0.5,1])$.

5. Calculate the corresponding covariance matrix $\Sigma' = (\mathbb{1} - B')^{-1} \operatorname{diag}(\sigma'^2)(\mathbb{1} - B')^{-\mathrm{T}}$.

6. Set $H := \operatorname{diag}((\Sigma'_{11})^{-1/2},\ldots,(\Sigma'_{pp})^{-1/2})$, and normalize the weights and error variances as follows:
$$B := HB'H^{-1}, \quad (\sigma_1^2,\ldots,\sigma_p^2)^{\mathrm{T}} := H^2(\sigma'^2_1,\ldots,\sigma'^2_p)^{\mathrm{T}} .$$
   It can easily be seen that the corresponding covariance matrix fulfills
$$\Sigma = (\mathbb{1} - B)^{-1} \operatorname{diag}(\sigma^2)(\mathbb{1} - B)^{-\mathrm{T}} = H\Sigma' H ,$$
   ensuring the desired normalization $\Sigma_{ii} = 1$ for all i.

Steps 1 and 3 are provided by the function `randomDAG()` of the R-package `pcalg` (Kalisch et al., 2012).

We considered families of targets of the form $\mathcal{I} = \{\emptyset, I_1,\ldots,I_k\}$, where $I_1,\ldots,I_k$ are k different, randomly chosen intervention targets of size m; the target size m had values between 1 and 4.

For a fixed sample size n, we produced approximately the same number of data samples for each target in the family $\mathcal{I}$ by using a level density $\mathcal{N}((2,\dots,2),(0.2)^2\mathbb{1}_m)$ in each case (see the model in Equation (1)). With this choice and the aforementioned normalization of Σ, the mean values of the intervention levels lay 2 standard deviations above the mean values of the observational marginal distributions. In total, we considered 4000 causal models and simulated 128 observational or interventional data sets from each of them by combining the following simulation parameters:

- $(p,s) \in \{(10,0.2),(20,0.1),(30,0.1),(40,0.1)\}$ with 1000 DAGs each.
- $k = 0, 0.2p, 0.4p, \dots, p$ for each value of p; the first setting is purely observational.
- $m \in \{1,2,4\}$.
- $n \in \{50,100,200,500,1000,2000,5000,10000\}$.

In addition, we generated causal models with $p \in \{50,100,200\}$ (100 DAGs each) and $p = 500$ (20 DAGs) with an expected vertex degree of 4 (which corresponds to a sparseness parameter of $s = 4/(p-1)$) and simulated 6 data sets for the parameters $k = 0.4$ and $n \in \{1000,2000,5000,10000,$ $20000,50000\}$ from each of these models. We only used these additional data sets for the investigation of the runtime of GIES.

5.2.2 Alternative Structure Learning Algorithms

We compare GIES with three alternative greedy search algorithms. The first one is the original GES of Chickering (2002b) which regards the complete interventional data set as observational (that is, ignores the list $\mathcal{T}$ of an interventional data set $(\mathcal{T},\mathbf{X})$ as defined in Equation (2)). The second one, which we call GIES-NT (for “no turning”), is a variant of GIES that stops after the first forward and backward phase and lacks the turning phase. The third algorithm, called GDS for “greedy DAG search”, is a simple greedy algorithm optimizing the same score function as GIES, but working on the space of DAGs instead of the space of $\mathcal{I}$-essential graphs; GDS simply adds, removes or turns arrows of DAGs in the forward, backward and turning phase, respectively. Furthermore, for $p \leq 20$, we compare with a dynamic programming (DP) approach proposed by Silander and Myllymäki (2006), an algorithm that finds a global optimum of any decomposable score function on the space of DAGs. Because of the exponential growth in time and memory requirements, we could not calculate DP estimates for models with $p \geq 30$ variables. For GDS and DP, we examine the $\mathcal{I}$-essential graph of the returned DAGs.

5.2.3 Quality Measures for Estimated Essential Graphs

The **structural Hamming distance** or SHD (Tsamardinos et al., 2006; we use the slightly adapted version of Kalisch and Bühlmann, 2007) is used to measure the distance between an estimated $\mathcal{I}$-essential graph $\hat{G}$ and a true $\mathcal{I}$-essential graph or DAG G. If A and $\hat{A}$ denote the adjacency matrices of G and $\hat{G}$, respectively, the SHD between G and $\hat{G}$ reads

$$\mathrm{SHD}(\hat{G},G) := \sum_{1 \leq i < j \leq p} \left(1 - \mathbb{1}_{\{(A_{ij}=\hat{A}_{ij}) \wedge (A_{ji}=\hat{A}_{ji})\}}\right) .$$

The SHD between $\hat{G}$ and G is the sum of the numbers of false positives of the skeleton, false negatives of the skeleton, and wrongly oriented edges. Those quantities are defined as follows. Two vertices which are adjacent in $\hat{G}$ but not in G count as one false positive, two vertices which are adjacent in G but not in $\hat{G}$ as one false negative. Two vertices which are adjacent in both G and $\hat{G}$, but connected with different edge types (that is, by a directed edge in one graph, by an undirected one in the other; or by directed edges with different orientations in both graphs) constitute a **wrongly oriented** edge.

5.2.4 RESULTS AND DISCUSSION

As we mentioned in Section 3.1, the undirected edges in the $\mathcal{I}$-essential graph $\mathcal{E}_{\mathcal{I}}(D)$ of some causal structure D are the edges with unidentifiable orientation. The number of undirected edges in $\mathcal{E}_{\mathcal{I}}(D)$ analyzed in the next paragraph is therefore a good measure for the identifiability of D. Later on, we study the performance of GIES and compare it to the other algorithms mentioned in Section 5.2.2.

*I*dentifiability under Interventions

In Figure 8, the number of non-$\mathcal{I}$-essential arrows is plotted as a function of the number k of non-empty intervention targets ($k = |\mathcal{I}| - 1$, see Section 5.2.1). With single-vertex interventions at 80% of the vertices, the majority of the DAGs used in the simulation are completely identifiable; with target size $m = 2$ or $m = 4$, this is already the case for $k = 0.6p$ or $k = 0.4p$, respectively. For the small target sizes used, the identifiability under k targets of size m is similar to the identifiability under $k \cdot m$ single-vertex targets.

A certain prudence is advisable when interpreting Figure 8 since the number of orientable edges also reflects the characteristics of the generated DAGs. Nevertheless, the plots show that the iden-

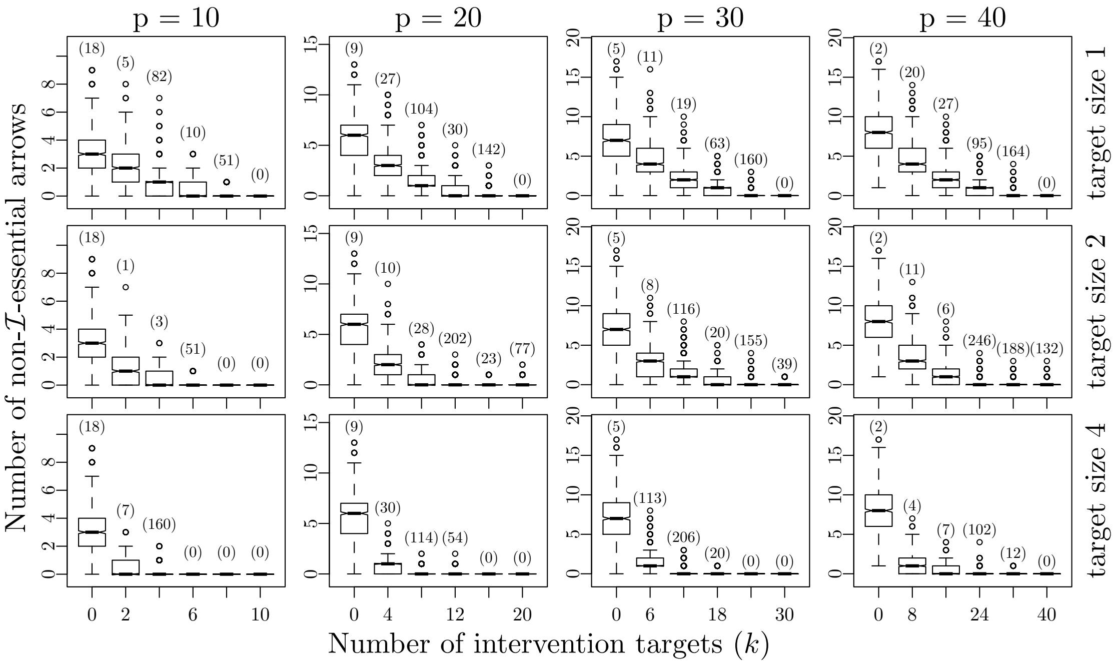


Figure 8: Number of non-$\mathcal{I}$-essential arrows as a function of the number k of intervention vertices. In parentheses: number of outliers in the corresponding boxplot.

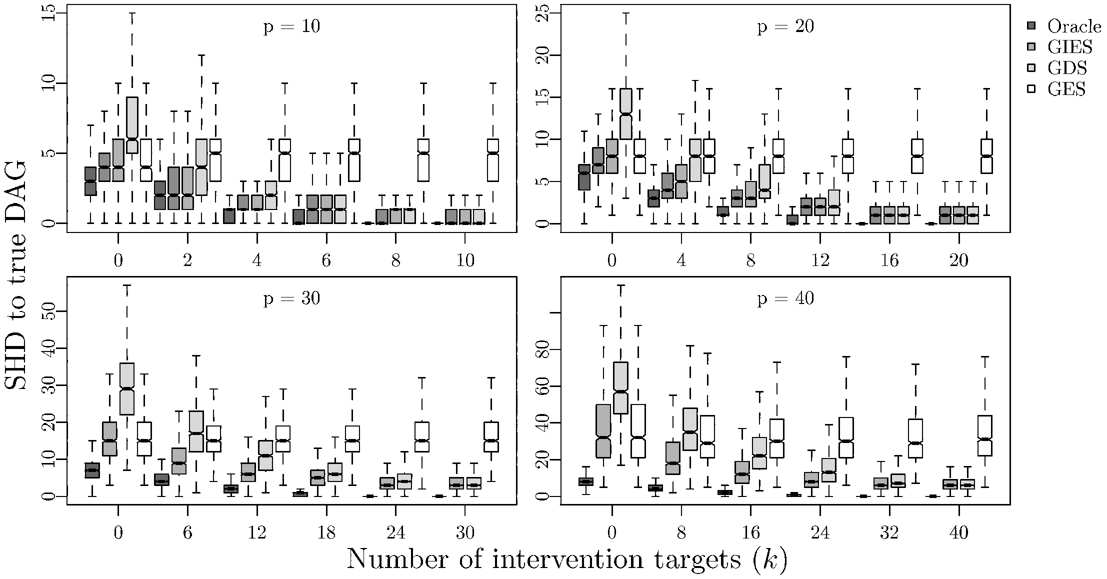


Figure 9: SHD between $\mathcal{I}$-essential graph $\hat{G}$ estimated from $n = 1000$ data points and true DAG D as a function of the number k of single-vertex intervention targets. "Oracle estimates" denote the respective true $\mathcal{I}$-essential graph $\mathcal{E}_{\mathcal{I}}(D)$, the best possible estimate under some family of targets $\mathcal{I}$ (see also Figure 8). DP estimates are missing in the two lower plots.

tifiability of causal models increases quickly even with few intervention targets. In regard of applications this is an encouraging finding since it illustrates that even a small number of intervention experiments can strongly increase the identifiability of causal structures.

*P*erformance of GIES

Figure 9 shows the structural Hamming distance between true DAG D and estimated $\mathcal{I}$-essential graph $\hat{G}$ for different algorithms as a function of the number k of intervention targets. Single-vertex interventions are considered; for larger targets, the overall picture is comparable (data not shown). In 10 out of 12 cases for $p \leq 20$, the median SHD values of GIES and DP estimates are equal; in the remaining cases, too, GIES yields estimates of comparable quality—at much lower computational costs.

In parallel with the identifiability, the estimates produced by the different algorithms improve for growing k. This illustrates that interventional data arising from different intervention targets carry *more* information about the underlying causal model than observational data of the same sample size.

For complete interventions, that is, $k = p$, every DAG is completely identifiable and hence its own $\mathcal{I}$-essential graph. Therefore, GDS and GIES are exactly the same algorithm in this case. With shrinking k, the performance of GDS compared to that of GIES gets worse. On the other hand, GES coincides with GIES in the observational case ($k = 0$). For growing k, the estimation performance of GES stays approximately constant; it can, as opposed to GIES, not make use of the additional information coming from interventions. To sum up, both the price of ignoring interventional Markov equivalence (GDS) and ignoring the interventional nature of the provided data sets (GES) are apparent in Figure 9.

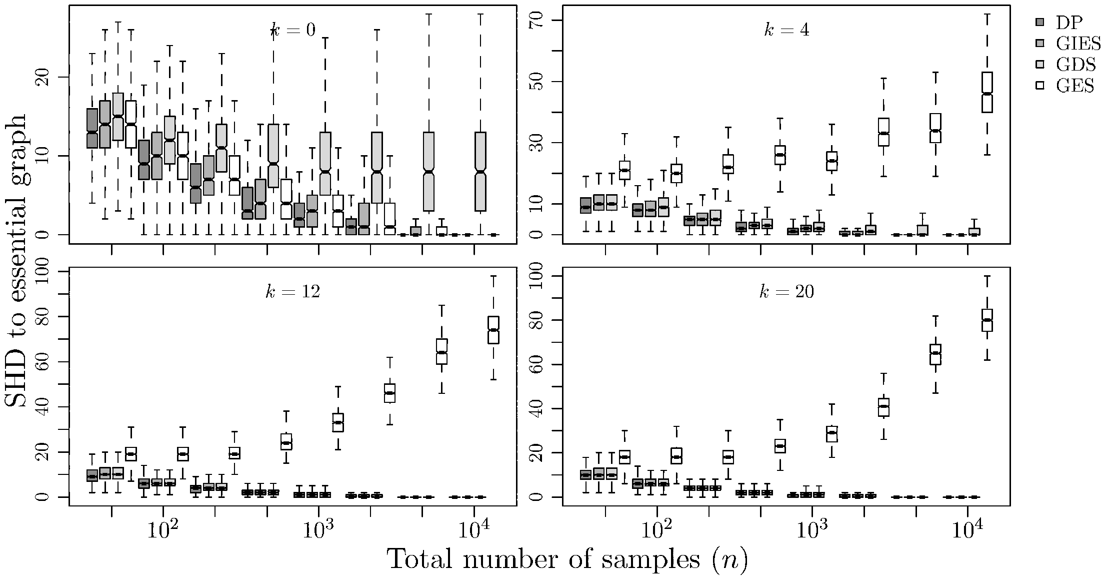


Figure 10: SHD between estimated and true $\mathcal{I}$-essential graph for different numbers k of intervention targets of size $m = 4$ for the DAGs with $p = 20$ vertices. The abscissa denotes the *total* sample size n. For example, a data set with $n = 1000$ and $k = 4$ consists of 200 observational samples and 200 interventional samples each arising from interventions at four different targets, see Section 5.2.1.

The performance of GIES as a function of the sample size n is plotted in Figure 10 for the DAGs with $p = 20$ vertices and intervention targets of size $m = 4$. The quality of the GIES estimates is comparable to that of the DP estimates. The behavior of the SHD values for growing n is a strong hint for the consistency of GIES in the limit $n \to \infty$ (note that the DP algorithm is consistent; Hauser and Bühlmann, 2012). In contrast, the plots for $k = 0$ and $k = 4$ again reveal the weak performance of GDS for small numbers of intervention vertices; the plots suggest that GDS, in contrast to GIES, does not yield a consistent estimator of the $\mathcal{I}$-essential graph due to being stuck in a bad local optimum.

The most striking result in Figure 10 is certainly the fact that the estimation performance of GES heavily decreases with growing n as long as the data is not observational ($k > 0$). This is not an artifact of GES, but a problem of model-misspecification: running DP for an *observational* model (that is, considering all data as observational as GES does) yields SHD values maximally 14% below that of GES (data not shown). For single-vertex interventions, the SHD values of the GES estimates stay approximately constant with growing n; for target size $m = 2$, its SHD values also increase, but not to the same extent as for $m = 4$.

In Figure 11, we compare the SHD between true and estimated $\mathcal{I}$-essential graphs with $p = 30$ vertices for estimates produced by different greedy algorithms; other vertex numbers give a similar picture. In most settings, GIES beats both GDS and GIES-NT. It combines both the advantage of GIES-NT, using the space of interventional Markov equivalence classes as search space, and GDS, the turning phase apparently reducing the risk of getting stuck in local maxima of the score function.

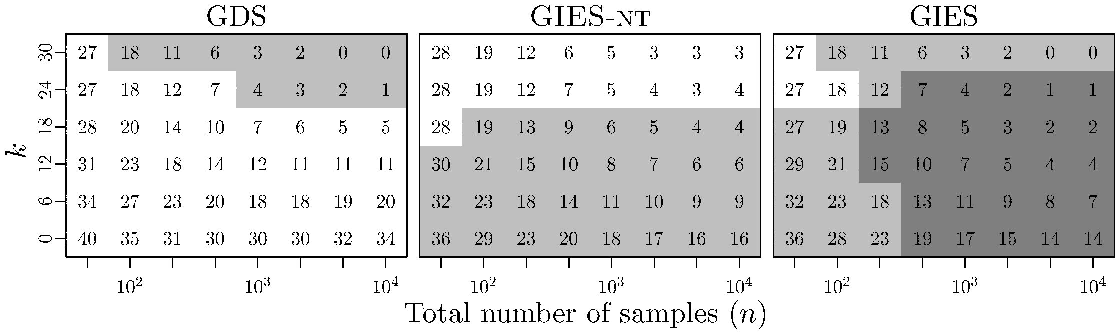


Figure 11: Mean SHD between estimated and true $\mathcal{I}$-essential graph for different greedy algorithms as a function of n and k; data for DAGs with $p = 30$ and single-vertex interventions. Shading: algorithm yielded significantly better estimates than one (▨) or two (■) of its competitors, respectively (paired t-test on a significance level of $\alpha = 5\%$).

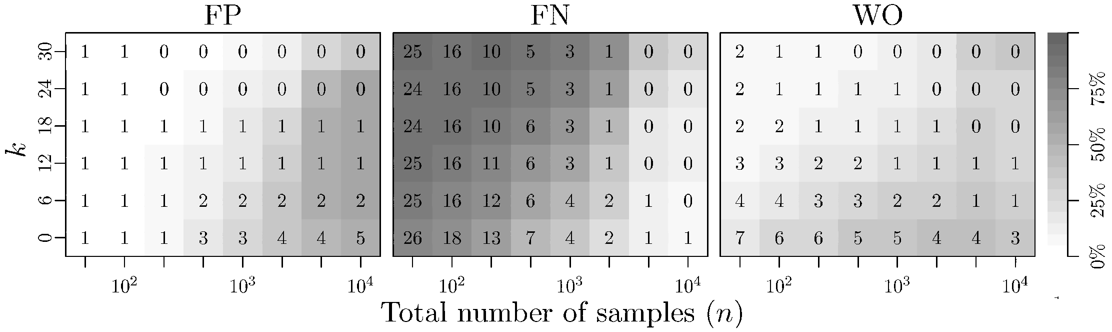


Figure 12: False positives (FP) and false negatives (FN) of the skeleton and wrongly oriented edges (WO; Section 5.2.3) of the GIES estimates compared to the true $\mathcal{I}$-essential graphs with $p = 30$ vertices; mean values as a function of k and n for single-vertex interventions. Shading: ratio of each quantity and the SHD between estimated and true $\mathcal{I}$-essential graph (dark means a large contribution to the SHD).

As noted in Section 5.2.3, the SHD between true and estimated interventional essential graphs can be written as the sum of false positives of the skeleton, false negatives of the skeleton and wrongly oriented edges. Those numbers are shown in Figure 12, again for GIES estimates under single-vertex interventions for DAGs with $p = 30$ vertices. False positives of the skeleton are the main contribution to the SHD values. In 60% of the cases, especially for large n and small k, wrongly oriented edges represent the second-largest contribution.

*R*untime Analysis

All algorithms evaluated in this section were implemented in C++ and compiled into a library using the GNU compiler g++ 4.6.1. The simulations—that is, the generation of data and the library calls—were performed using R 2.13.1. All simulations were run on an AMD Opteron 8380 CPU with 2.5 GHz and 2 GB RAM.

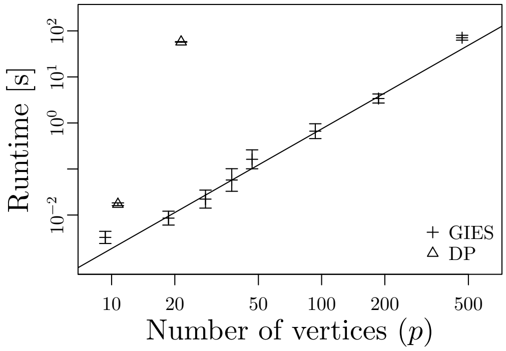


Figure 13: Runtime of GIES and DP as a function of the vertex number.

Figure 13 shows the running times of GIES and DP as a function of the number p of vertices. GDS had running times of the same order of magnitude as GIES; they were actually up to 50% higher since we used a basic implementation of GDS compared to an optimized version of GIES (running times of GDS are not plotted for this reason). The linearity of the GIES values in the log-log plot (see the solid line in Figure 13) indicate a polynomial time complexity of the approximate order $O(p^{2.8})$, in contrast to the exponential complexity of DP; note that GIES also has an exponential *worst case* complexity (see Section 4.4). The multiple linear regression $\log(t) = \beta_0 + \beta_1 \log(p) + \beta_2 \log(|E|) + \varepsilon$, where t denotes the runtime and E the edge set of the true DAG, yields coefficients $\hat{\beta}_1 = 1.01$ and $\hat{\beta}_2 = 0.94$.

5.3 DREAM4 Challenge

We also measured the performance of GIES on synthetic gene expression data sets from the DREAM4 *in silico* challenge (Marbach et al., 2010; Prill et al., 2010). Our goal here was to evaluate predictions of expression levels of gene knockout or knockdown experiments by cross-validation based on the provided interventional data.

5.3.1 DATA

The DREAM4 challenge provides five data sets with an ensemble of interventional and observational data simulated from five biologically plausible, possibly *cyclic* gene regulatory networks with 10 genes (Marbach et al., 2009). The data set of each network consists of

- 11 observational measurements, simulated from random fluctuations of the system parameters (resembling observational data measured in different individuals);
- 10 measurements from single-gene knockdowns, one knockdown per gene;
- 10 measurements from single-gene knockouts, one knockout per gene;
- five time series with 21 time points each, simulated from an unknown change of parameters in the first half (corresponding to measurements under a perturbed chemical environment having unknown effects on the gene regulatory network) and from the unperturbed system in the second half.

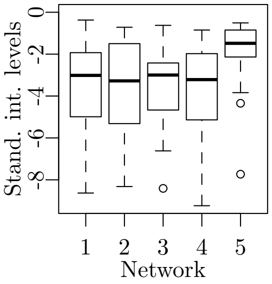


Figure 14: Standardized intervention levels in the different DREAM4 data sets. Data is scaled such that the observational samples have empirical mean 0 and standard deviation 1.

Since our framework can not cope with uncertain interventions (that is, interventions with unknown target), we only used the 50 observational measurements of the second half of the time series. Altogether, we have, from each network, a total of 81 data points, 61 observational and 20 interventional ones. We normalized the data such that the observational samples of each gene have mean 0 and standard deviation 1. In this normalization, 95% of the intervention levels (that is, the expression levels of knocked out or knocked down genes) lie between -8.37 and -0.62 with a mean of -3.30 (Figure 14).

5.3.2 METHODS

We used each interventional measurement (20 per network) as one test data point and predicted its value from a network estimated with training data consisting either of the 80 remaining data points, or the 61 observational measurements alone. We used GIES, GES and PC (Spirtes et al., 2000) to estimate the causal models and evaluated the prediction accuracy by the mean squared error (MSE). We will use abbreviations like "GES(80)" or "PC(61)" to denote GES estimates based on a training set of size 80 or PC estimates based on an observational training set of size 61, respectively.

For a given DAG, we predicted interventional gene expression levels based on the estimated structural equation model after replacing the structural equation of the intervened variable by a constant one; see Section 5.1 for connection between Gaussian causal models and structural equations, especially Equation (5). GES and PC regard all data as observational and yield an observational essential graph. For those algorithms, we enumerated all representative DAGs of the estimated equivalence class using the function `allDags()` of the R package `pcalg` (Kalisch et al., 2012), calculated an expression level with each of them, and took the mean of those predictions. GIES(80) yields a single DAG in each case since the 19 interventional measurements in the training data ensure complete identifiability.

Furthermore, we used the evaluation script provided by the DREAM4 challenge to assess the quality of our network predictions to those sent in to the challenge by participating teams. This evaluation is based on the area under the ROC curve (AUROC) of the true and false positive rate of the edge predictions.

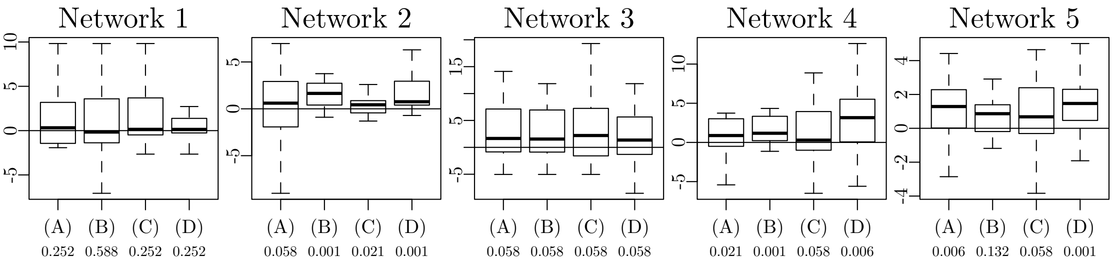


Figure 15: Upper row: MSE values of GIES and competitors; lower row: differences of MSE values as defined in Equation (7); large values indicate a good performance of GIES. (A) GIES(80), (B) PC(80), (C) PC(61), (D) GES(80), (E) GES(61). Numbers below the boxplots: p-values of a one-sided sign test.

5.3.3 RESULTS

Figure 15 shows boxplots of MSE differences between GIES(80) and its competitors; that is, we consider quantities of the form

$$\Delta\mathrm{MSE}_{\mathrm{comp}} := \mathrm{MSE}_{\mathrm{comp}} - \mathrm{MSE}_{\mathrm{GIES}(80)}, \tag{7}$$

where comp stands for one of the competitors. Since the MSE differences showed a skewed distribution in general, we used a sign test for calculating their p-values.

Except for one case (PC(61) in network 1), GIES(80) always yielded the best predictions of all competitors. Although all data sets are dominated by observational data (61 observational measurements versus 20 interventional ones), GIES can make use of the additional information carried by interventional data points to rule out its observational competitors. On the other hand, the dominance of observational data is probably one of the reasons for the fact that GIES does not outperform the observational methods more clearly but has an overall performance which is comparable with that of its competitors. Another reason could be the fact that the underlying networks used for data generation are not acyclic as assumed by GIES. Interestingly, the winning margin of GIES in network 5 was not smaller than in other networks although the corresponding data set has the smallest intervention levels (in absolute values; see Figure 14).

29 teams participated in the DREAM4 challenge. Their AUROC values are available from the DREAM4 website;[1] adding our values gives a data set of 30 evaluations. Among those, our results had overall rank 10, and ranks 8, 4, 21, 10 and 3, respectively, for networks 1 to 5. Except for network 3, we could keep up with the best third of the participating teams despite the beforementioned model misspecification given by the assumption of acyclicity, and despite the fact that we ignored the time series structure and half of the time series data.

6. Conclusion

We gave a definition and a graph theoretic criterion for the Markov equivalence of DAGs under multiple interventions. We characterized corresponding equivalence classes by their *essential graph*,

[1] DREAM4 can be found at `http://wiki.c2b2.columbia.edu/dream/index.php/D4c2`.

defined as the union of all DAGs in an equivalence class in analogy to the observational case. Using those essential graphs as a basis for the algorithmic representation of interventional Markov equivalence classes, we presented a new greedy algorithm (including a new turning phase), GIES, for learning causal structures from data arising from multiple interventions.

In a simulation study, we showed that the number of non-orientable edges in causal structures drops quickly even with a small number of interventions; our description of interventional essential graphs makes it possible to *quantify* the gain in identifiability. For a fixed sample size n, GIES estimates got closer to the true causal structure as the number of intervention vertices grew. For DAGs with $p \leq 20$ vertices, the GIES algorithm could keep up with a consistent, exponential-time DP approach maximizing the BIC score. It clearly beat GDS, a simple greedy search on the space of DAGs, as well as GES which cannot cope with interventional data. Our novel turning phase proved to be an improvement of GES even on observational data, as it was already conjectured by Chickering (2002b). Applying GIES to synthetic data sets from the DREAM4 challenge (Marbach et al., 2010), we got better predictions of gene expression levels of knockout or knockdown experiments than with observational estimation methods.

The accurate structure learning performance of GIES in the limit of large data sets raises the question whether GIES is consistent. Chickering (2002b) proved the consistency of GES on observational data. However, the generalization of his proof for GIES operating on interventional data is not obvious since such data are in general not identically distributed.

Acknowledgments

We are grateful to Markus Kalisch for very carefully reading the proofs and for his helpful comments on the text. Furthermore, we would like to thank the anonymous reviewers for constructive comments.

Appendix A. Graphs

In this appendix, we shortly summarize our notation (mostly following Andersson et al., 1997) and basic facts concerning graphs. All statements about perfect elimination orderings that are used in Sections 3 and 4 are listed or proven in Section A.2.

A.1 Definitions and Notation

A **graph** is a pair $G = (V, E)$, where V is a finite set of vertices and $E \subset E^*(V) := (V \times V) \setminus \{(a,a) | a \in V\}$ is a set of edges. We use graphs to denote causal relationships between random variables $X_1, \ldots, X_p$. To keep notation simple, we always assume $V = \{1, 2, \ldots, p\} =: [p]$, in order to represent each random variable by its index in the graph.

An edge $(a,b) \in E$ with $(b,a) \in E$ is called **undirected** (or a **line**), whereas an edge $(a,b) \in E$ with $(b,a) \notin E$ is called **directed** (or an **arrow**). Consequently, a graph G is called directed (or undirected, resp.) if all its edges are directed (or undirected, resp.); a directed graph is also called

digraph for short. We use the short-hand notation

$$\begin{aligned} a \longrightarrow b \in G \quad &:\Leftrightarrow \quad (a,b) \in E \wedge (b,a) \notin E, \\ a \longrightarrow\!\!\!\!\!\!\!\!- \; b \in G \quad &:\Leftrightarrow \quad (a,b) \in E \wedge (b,a) \in E, \\ a \cdots\!\cdot b \in G \quad &:\Leftrightarrow \quad (a,b) \in E \vee (b,a) \in E. \end{aligned}$$

A **subgraph** of some graph G is a graph $G' = (V', E')$ with the property $V' \subset V$, $E' \subset E$, denoted by $G' \subset G$. For a subset $A \subset V$ of the vertices of G, the **induced subgraph** on A is $G[A] := (A, E[A])$, where $E[A] := E \cap (A \times A)$. A **v-structure** (also called *immorality* by, for example, Lauritzen, 1996) is an induced subgraph of G of the form $a \longrightarrow b \longleftarrow c$. The **skeleton** of a graph G is the undirected graph $G^u := (V, E^u)$, $E^u := \{(a,b) \in V \times V \mid a \cdots\!\cdot b \in G\}$. For two graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ on the same vertex set, we define the union and the intersection as $G_1 \cup G_2 := (V, E_1 \cup E_2)$ and $G_1 \cap G_2 := (V, E_1 \cap E_2)$, respectively. For a graph $G = (V, E)$ and $(a,b) \in E^*(V)$, we use the shorthand notation $G - (a,b) := (V, E \setminus \{(a,b)\})$ and $G + (a,b) := (V, E \cup \{(a,b)\})$.

The following sets describe the local environment of a vertex a in a graph G:

$$\begin{aligned} \mathrm{pa}_G(a) \quad &:= \quad \{b \in V \mid b \longrightarrow a \in G\}, \text{ the **parents** of } a, \\ \mathrm{ch}_G(a) \quad &:= \quad \{b \in V \mid a \longrightarrow b \in G\}, \text{ the **children** of } a, \\ \mathrm{ne}_G(a) \quad &:= \quad \{b \in V \mid a \,\text{—}\, b \in G\}, \text{ the **neighbors** of } a, \\ \mathrm{ad}_G(a) \quad &:= \quad \{b \in V \mid a \cdots\!\cdot b \in G\}, \text{ the vertices **adjacent** to } a. \end{aligned}$$

The subscripts "G" in the above definitions are omitted when it is clear which graph is meant. For a set $A \subset V$ of vertices, we generalize those definitions as follows:

$$\mathrm{pa}_G(A) := \bigcup_{a \in A} \mathrm{pa}_G(a) \setminus A, \quad \mathrm{ne}_G(A) := \bigcup_{a \in A} \mathrm{ne}_G(a) \setminus A, \text{ etc.}$$

The **degree** of a vertex $a \in V$ is defined as $\deg_G(a) := |\,\mathrm{ad}_G(a)|$.

For two distinct vertices a and $b \in V$, a **chain** of length k from a to b is a sequence of distinct vertices $\gamma = (a \equiv a_0, a_1, \ldots, a_k \equiv b)$ such that for each $i = 1, \ldots, k$, either $a_{i-1} \longrightarrow a_i \in G$ or $a_{i-1} \longleftarrow a_i \in G$; if for all i, $(a_{i-1}, a_i) \in E$ (that is, $a_{i-1} \longrightarrow a_i \in G$ or $a_{i-1} \,\text{—}\, a_i \in G$), the sequence γ is called a **path**. If at least one edge $a_{i-1} \longrightarrow a_i$ is directed in a path, the path is called *directed*, otherwise *undirected*. A **(directed) cycle** is defined as a (directed) path with the difference that $a_0 = a_n$. Paths define a preorder on the vertices of a graph: $a \preceq_G b :\Leftrightarrow \exists$ a path γ from a to b in G. Furthermore, $a \approx_G b :\Leftrightarrow (a \preceq_G b) \wedge (b \preceq_G a)$ is an equivalence relation on the set of vertices.

An undirected graph $G = (V, E)$ is **complete** if all pairs of vertices are adjacent. A **clique** is a subset of vertices $C \subset V$ such that $G[C]$ is complete; a vertex $a \in V$ is called **simplicial** if $\mathrm{ne}(a)$ is a clique. An undirected graph G is called **chordal** if every cycle of length $k \geq 4$ contains a **chord**, that means two nonconsecutive adjacent vertices. For pairwise disjoint subsets $A, B, S \subset V$ with $A \neq \emptyset$ and $B \neq \emptyset$, A and B are **separated** by S in G if every path from a vertex in A to a vertex in B contains a vertex in S.

A **directed acyclic graph**, or **DAG** for short, is a digraph that contains no cycle. In the paper, we mostly use the symbol D for DAGs, whereas arbitrary graphs are, as in this appendix, mostly named G. Chain graphs can be viewed as something between undirected graphs and DAGs: a graph $G = (V, E)$ is a **chain graph** if it contains no directed cycle; undirected graphs and DAGs are

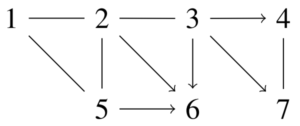


Figure 16: A chain graph G with three chain components $A = T_G(1) = [1]_{\approx_G} = \{1,2,3,5\}$, $B = T_G(6) = \{6\}$ and $C = T_G(4) = \{4,7\}$. The arrows induce the partial order $A \preceq_G B$, $A \preceq_G C$. The graph is no chain graph anymore when we replace the arrow $3 \longrightarrow 4$ by a line since this would create a directed cycle: $(3,7,4,3)$.

special cases of chain graphs. The equivalence classes in V w.r.t. the equivalence relation $\approx_G$ are the connected components of G after removing all directed edges. We denote the quotient set of V by $\mathbf{T}(G) := V/\approx_G$, and its members $T \in \mathbf{T}(G)$ are called **chain components** of G. For a vertex $a \in V$, $T_G(a)$ stands for $[a]_{\approx_G}$. The preorder $\preceq_G$ on V induces in a canonical way a *partial order* on $\mathbf{T}(G)$ which we also denote by $\preceq_G$: $T_G(a) \preceq_G T_G(b) :\Leftrightarrow a \preceq_G b$. An illustration is shown in Figure 16.

An **ordering** of a graph is a bijection $[p] \to V$, hence, since we assume $V = [p]$ here, a permutation $\sigma \in S_p$. An ordering σ canonically induces a total order on V by the definition $a \leq_\sigma b :\Leftrightarrow \sigma^{-1}(a) \leq \sigma^{-1}(b)$. An ordering $\sigma = (v_1, \ldots, v_p)$ is called a **perfect elimination ordering** if for all i, v_i is simplicial in $G^u[\{v_1, \ldots, v_i\}]$. A graph $G = (V,E)$ is a DAG if and only if the previously defined preorder $\preceq_G$ is a partial order; such a partial order can be extended to a total order (Szpilrajn, 1930). Thus every DAG has at least one **topological ordering**, that is an ordering σ whose total order $\leq_\sigma$ extends $\preceq_G$: $a \preceq_G b \Rightarrow a \leq_\sigma b$. For $\sigma \in S_p$, a DAG $D = ([p],E)$ is said to be **oriented according to** σ if σ is a topological ordering of D. In a DAG D with topological ordering σ, the arrows point from vertices with low to vertices with high ordered indices. The vertex $\sigma(1)$ is a **source**, that means all arrows point away from it.

A.2 Perfect Elimination Orderings

Perfect elimination orderings play an important role in the characterization of interventional Markov equivalence classes of DAGs as well as in the implementation of the Greedy Interventional Equivalence Search (GIES). In this section, we provide all results for this topic that are used as auxiliary tools in the proofs of Sections 3 and 4.

Lemma 37 *Let $D = (V,E)$ be a DAG. D has no v-structures if and only if any topological ordering of D is a perfect elimination ordering.*

The proof of this lemma follows easily from the definitions of a v-structure and a perfect elimination ordering. Moreover, if *any* topological ordering of a DAG is a perfect elimination ordering, this is automatically the case for *every* topological ordering.

Proposition 38 (Rose, 1970) *Let $G = (V,E)$ be an undirected graph. Then G is chordal if and only if it has a perfect elimination ordering.*

```
   Input  : An undirected graph $G = (V, E)$
   Output: An ordering $\sigma$ of the vertices $V$, called a LEXBFS-ordering
   $\Sigma \leftarrow (V)$; // Initialize sequence $\Sigma$ of vertex sets to contain the single set $V$ in
      the beginning
   $\sigma \leftarrow ()$; // Initialize output sequence of vertices
3  while $\Sigma \neq \emptyset$ do
4      Remove a vertex $a$ from the first set in the sequence $\Sigma$;
       if first set of $\Sigma$ is empty then remove first set from $\Sigma$;
       Append $a$ to $\sigma$;
       Mark all sets of $\Sigma$ as not visited;
       foreach $b \in \mathrm{ne}_G(a)$ s.t. $b \in S$ for some $S \in \Sigma$ do
           if $S$ not visited then
               Insert empty set $T$ into $\Sigma$ in front of $S$;
               Mark $S$ as visited;
           else let $T$ be the set preceding $S$ in $\Sigma$;
13         Move $b$ from $S$ to $T$;
14         if $S = \emptyset$ then remove $S$ from $\Sigma$;
```

Algorithm 6: $\mathrm{LEXBFS}(V, E)$. Lexicographic breadth-first search in the so-called "partitioning paradigm" (Rose et al., 1976; Corneil, 2004)

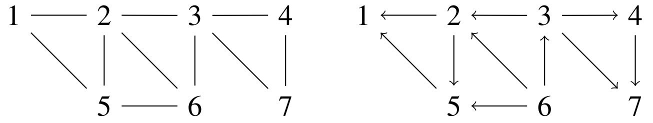
Perfect elimination orderings of chordal graphs can be produced by a variant of the breadth-first search algorithm, the so-called lexicographic breadth-first search (LEXBFS; see Algorithm 6). The term "lexicographic" reflects the fact that the algorithm visits edges in lexicographic order w.r.t. the produced ordering σ.

Proposition 39 (Rose et al., 1976) *Let $G = (V, E)$ be an undirected chordal graph with a* LEXBFS-*ordering σ. Then σ is also a perfect elimination ordering on G.*

Corollary 40 *Let G be an undirected chordal graph with a* LEXBFS-*ordering σ. A DAG $D \subset G$ with $D^u = G$ that is oriented according to σ has no v-structures.*

Corollary 40 is a consequence of Lemma 37 and Proposition 39. According to this corollary, LEXBFS-orderings can be used for constructing representatives of essential graphs (see Proposition 16). Corollary 40 as well as Algorithm 6 are therefore of great importance for the proofs and algorithms of Sections 3 and 4.

Figure 17 shows an undirected chordal graph G and a DAG D that has the skeleton G and is oriented according to a LEXBFS-ordering σ of G. The functioning of Algorithm 6 when producing a LEXBFS-ordering on G is illustrated in Table 1. Note that the "sets" in Σ are written as tuples. We use this notation to ensure that we can always remove the first (leftmost) vertex from the first "set" of Σ (line 3 in Algorithm 6), and that we keep the relative order of vertices when moving them from one set S to the preceding one, T, in Σ (line 12 in Algorithm 6). Throughout the text, we always assume an implementation of Algorithm 6 in which the data structure used to represent the "sets" in the sequence Σ guarantees this "first in, first out" (FIFO) behavior. In particular, the start sequence $(v_1, v_2, \ldots, v_p)$ of the vertices in V provided to the algorithm determines the vertex the LEXBFS-ordering $\sigma := \mathrm{LEXBFS}((v_1, \ldots, v_p), E)$ starts with: $\sigma(1) = v_1$. It is often sufficient



(a) G

(b) D

Figure 17: An undirected, chordal graph $G = ([7], E)$ and the DAG D we get by orienting all edges of G according to the ordering $\sigma := \text{LEXBFS}((6,3,1,2,4,5,7),E)$.

i	Σ	σ
0	$((6,3,1,2,4,5,7))$	$()$
1	$((3,2,5),(1,4,7))$	(6)
2	$((2),(5),(4,7),(1))$	$(6,3)$
3	$((5),(4,7),(1))$	$(6,3,2)$
4	$((4,7),(1))$	$(6,3,2,5)$
5	$((7),(1))$	$(6,3,2,5,4)$
6	$((1))$	$(6,3,2,5,4,7)$
7	$()$	$(6,3,2,5,4,7,1)$

Table 1: State of the sequences Σ and σ after the i^{th} run ($i = 0, \ldots, 7$) of the while loop (lines 2 to 13) of Algorithm 6 applied to the graph G of Figure 17 with start order $(6,3,1,2,4,5,7)$.

to specify the start order of LEXBFS up to arbitrary orderings of some subsets of vertices. For a set $A = \{a_1, \ldots, a_k\} \subset V$ and an additional vertex $v \in V \setminus A$, for example, we use the notation

$$\text{LEXBFS}((A, v, V \setminus (A \cup \{v\})), E), \quad \text{or even} \quad \text{LEXBFS}((A, v, \ldots), E)$$

to denote a LEXBFS-ordering produced from a start order of the form $(a_1, \ldots, a_k, v, \ldots)$, without specifying the orderings of A and $V \setminus (A \cup \{v\})$.

By using appropriate data structures (for example, doubly linked lists for the representation of Σ and its sets, and a pointer at each vertex pointing to the set in Σ in which it is contained), Algorithm 6 has complexity $O(|E| + |V|)$ (Corneil, 2004).

For the rest of this section, we state further consequences of Lemma 37 and Proposition 39 which are relevant for the proofs of Sections 3 and 4.

Corollary 41 *Let $G = (V, E)$ be an undirected chordal graph, and let $a - b \in G$. There exist DAGs D_1 and D_2 with $D_1, D_2 \subset G$ and $D_1^u = D_2^u = G$ without v-structures such that $a \longrightarrow b \in D_1$ and $a \longleftarrow b \in D_2$.*

Proof Set $\sigma_1 := \text{LEXBFS}((a, V \setminus \{a\}), E)$ and $\sigma_2 := \text{LEXBFS}((b, V \setminus \{b\}), E)$, and let D_1 and D_2 be two DAGs with skeleton G and oriented according to σ_1 and σ_2, resp. Then, by Corollary 40, D_1 and D_2 have the requested properties; in particular, all edges point away from a in D_1, whereas all edges point away from b in D_2. ■

Corollary 42 (Andersson et al., 1997) *Let $G = (V,E)$ be an undirected chordal graph, $a \in V$ and $C \subset \mathrm{ne}(a)$. Then there is a DAG $D \subset G$ with $D^u = G$ and $\{b \in \mathrm{ne}(a) \mid b \longrightarrow a \in D\} = C$ that has no v-structures if and only if C is a clique.*

Proof *"$\Rightarrow$":* Assume that there are non-adjacent vertices $b, c \in C$. Then, $b - a - c$ is an induced subgraph of G, and by construction, the same vertices occur in configuration $b \longrightarrow a \longleftarrow c$ in D, which means that D has a v-structure, a contradiction.

"$\Leftarrow$": Let $(c_1, \ldots, c_k)$ be an arbitrary ordering of C. Run LEXBFS on a start order of the form $(c_1, \ldots, c_k, a, \ldots)$. After the first run of the while loop (lines 2 to 13 of Algorithm 6), $\sigma = (c_1)$, and the first set in the sequence Σ contains $(C \cup \{a\}) \setminus \{c_1\}$ as a subset (all vertices in this set are adjacent to c_1), in an unchanged order $c_2, \ldots, c_k, a$ due to our FIFO convention. After the second run of the while loop, $\sigma = (c_1, c_2)$, and the first set in Σ contains $(C \cup \{a\}) \setminus \{c_1, c_2\}$, and so on. In the end, we get a LEXBFS-ordering of the form $\sigma = (c_1, \ldots, c_k, a, \ldots)$. Orienting the edges of G according to σ yields a DAG with the requested properties by Corollary 40. ■

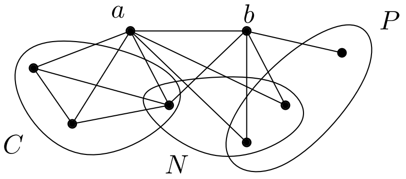


Figure 18: Configuration of vertices in Proposition 43.

Proposition 43 *Let $G = (V,E)$ be an undirected, chordal graph, $a - b \in G$, and $C \subset \mathrm{ne}_G(a) \setminus \{b\}$ a clique. Let $N := \mathrm{ne}_G(a) \cap \mathrm{ne}_G(b)$, and assume that $C \cap N$ separates $C \setminus N$ and $N \setminus C$ in $G[\mathrm{ne}_G(a)]$ (see Figure 18). Then there exists a DAG $D \subset G$ with $D^u = G$ such that*

(i) D has no v-structures;

(ii) all edges in $D[C \cup \{a\}]$ point towards a;

(iii) all other edges of D point away from vertices in $C \cup \{a\}$ (in particular, $a \longrightarrow b \in D$);

(iv) $b \longrightarrow d \in D$ for all $d \in P := \mathrm{ne}_G(b) \setminus (C \cup \{a\})$.

Proof Set $\sigma := \mathrm{LEXBFS}((C, a, b, \ldots), E)$, and let D be the DAG that we get by orienting the edges of G according to σ. As in Corollary 42, properties (i) to (iii) are met.

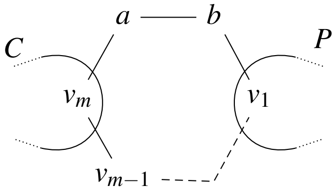
It remains to show that b occurs before any $d \in P$ in σ (that means $b <_\sigma d \,\forall\, d \in P$) in order that D obeys property (iv). W.l.o.g., we can assume $C = \{1, 2, \ldots, k\}$, $a = k+1$ and $b = k+2$. The start order of the vertices for LEXBFS is then $(1, 2, \ldots, p)$. Due to the FIFO convention for the sets of the sequence Σ in Algorithm 6, b always precedes any $d \in P$ whenever they appear in the same set; hence we only must show that the set containing b is never preceded by a set containing some $d \in P$ in Σ.

Suppose, for the sake of contradiction, that this is the case for some $d \in P$; name $v_1 := d$. At the beginning, b is in the same set as v_1 in the sequence Σ; there is some vertex v_2 that forces LEXBFS to move v_1 into the set preceding the one containing b. A careful inspection of Algorithm 6 shows that v_2 is the vertex which is minimal w.r.t. $\leq_\sigma$ in

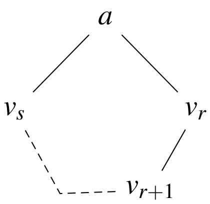
$$S(v_1) := \{v \in V \mid v \in \mathrm{ne}_G(v_1) \setminus \mathrm{ne}_G(b), v <_\sigma b\}.$$

If $v_2 > b$ (that is, if $v_2 \notin C \cup \{a\}$ due to our convention), v_2, as v_1, always follows b whenever they are in the same set in Σ. Therefore, $v_2 <_\sigma b$ implies that there is some vertex v_3 that moves v_2 in the set preceding the one of b in Σ during the execution of LEXBFS; as before, we see that this is the vertex which is minimal w.r.t. $\leq_\sigma$ in $S(v_2)$.

We can now continue to construct this sequence $v_{i+1} := \min S(v_i)$ (always taking the minimum w.r.t. $\leq_\sigma$) until we find some vertex $v_m < b$; this is a vertex in $C \cup \{a\}$. Even more, $v_m \in C \setminus N$, since, by definition of $S(v_{m-1})$, we only consider vertices that are not adjacent to b. We now have constructed a path $\gamma = (v_1, \ldots, v_m)$ of length $m \geq 2$ in G such that $v_1 \in P$, $v_i \notin \mathrm{ne}_G(b)\ \forall\ i > 1$, $v_i > b\ \forall\ i < m$ and $v_m \in C \setminus N$; furthermore, we have $v_m <_\sigma \ldots <_\sigma v_1 <_\sigma b$. The path γ can be elongated to a cycle $(a, v_0 := b, v_1, v_2, \ldots, v_m, a)$:



We now claim that $v_i — a \in G$ for all $0 \leq i \leq m$. This is clearly the case for $i = 0$ and $i = m$ by construction. Assume, for the sake of contradiction, that there is some i, $0 < i < m$, that is not adjacent to a. Let r be the *largest* index smaller than i such that $v_r — a \in G$ and s be the *smallest* index larger than i such that $v_s — a \in G$. Then the following is an *induced* subgraph of G:



Note that a chord between different v_l's, say, a chord of the form $v_l — v_{l+h}$ with $h \geq 2$, would violate the minimality of v_{l+1} in the set $S(v_l)$. This means that G contains an induced cycle of length 4 or more, contradicting the chordality.

This proves the claim that $v_i — a \in G$ for all $0 \leq i \leq m$, or, in other words, $v_i \in \mathrm{ne}_G(a)$ for all $0 \leq i \leq m$. Hence $v_1 \in N \setminus C$, and γ is a path from $N \setminus C$ to $C \setminus N$ in $G[\mathrm{ne}_G(a)]$ that has no vertex in $C \cap N$, in contradiction with the assumption. ■

Proposition 44 *Let $G = (V, E)$ be a chain graph with chordal chain components that does not contain $a \longrightarrow b — c$ as an induced subgraph, and let $D \subset G$ be a digraph with $D^u = G^u$. D is acyclic and has the same v-structures as G if and only if $D[T]$ is oriented according to a perfect elimination ordering for each chain component $T \in \mathbf{T}(G)$.*

Proof *“⇒”*: let $T \in \mathbf{T}(G)$. $G[T]$ obviously does not have any v-structures, hence $D[T]$ has no v-structures, either. It follows from Lemma 37 that $D[T]$ must be oriented according to a perfect elimination ordering.

"$\Leftarrow$": for each $T \in \mathbf{T}(G)$, $D[T]$ is acyclic by construction. Assume that D has some directed cycle γ; this cycle must reach different chain components of G, so it contains at least one edge $a \longrightarrow b$ that is also present in G. Because of $D \subset G$ and $D^u = G^u$, γ is also a cycle in G; and since $a \longrightarrow b \in G$, it is even a *directed* cycle in G, a contradiction. So D is acyclic.

By construction, every v-structure in G is also present in D. Suppose that D has some v-structure $a \longrightarrow b \longleftarrow c$ that G has not. a, b and c cannot belong to the same chain component of G according to Lemma 37. So, w.l.o.g., $a \longrightarrow b — c$ must be an induced subgraph of G, contradicting the assumption. Hence D and G have the same v-structures. ■

Appendix B. Proofs

In this appendix, the technically interested reader finds all proofs that were left out in Sections 2 to 4 for better readability.

B.1 Proofs for Section 2

We start with the proof of Lemma 8 which motivates Definition 7 by showing that, for some DAG D and some (conservative) family of targets $\mathcal{I}$, the elements of $\mathcal{M}_{\mathcal{I}}(D)$ are exactly the density tuples that can be realized as interventional densities of a causal model with structure D. Note that we use the conservativeness of $\mathcal{I}$ only in the proof of point (ii); it can even be proven without assuming conservativeness, although the proof becomes harder.

Proof of Lemma 8

(i) $f(x|\,\mathrm{do}(X_I = U_I))$ obeys the Markov property of $D^{(I)}$ (Section 2.1). Furthermore, for $I, J \in \mathcal{I}$ and $a \notin I \cup J$, we have

$$f(x_a \mid x_{\mathrm{pa}_D(a)}; \mathrm{do}(X_I = U_I)) = f(x_a \mid x_{\mathrm{pa}_D(a)}) = f(x_a \mid x_{\mathrm{pa}_D(a)}; \mathrm{do}(X_J = U_J))$$

by the truncated factorization of Equation (1).

(ii) Let $a \in [p]$. Since $\mathcal{I}$ is conservative, there is some $I \in \mathcal{I}$ such that $a \notin I$. Define $h_a(x_a, x_{\mathrm{pa}_D(a)}) := f^{(I)}(x_a | x_{\mathrm{pa}_D(a)})$. Note that, due to Definition 7, the function h_a does *not* depend on the choice of I.
Let $f(x) := \prod_{a=1}^{p} h_a(x_a, x_{\mathrm{pa}_D(a)})$; this is a positive density on $\mathcal{X}$ with $f(x_a | x_{\mathrm{pa}_D(a)}) = h_a(x_a, x_{\mathrm{pa}_D(a)})$, hence $f \in \mathcal{M}(D)$ and (D, f) is a causal model.
By defining level densities $\tilde{f}_I(x_I) := \prod_{i \in I} f^{(I)}(x_i)$, we can construct an intervention setting $\mathcal{S} := \{(I, \tilde{f}_I)\}_{I \in \mathcal{I}}$ with the requested properties. ■

The proof of the main result of Section 2, the graph theoretic criterion for two DAGs being interventionally Markov equivalent (Theorem 10), requires additional lemmas.

Lemma 45 *Let D be a DAG, $\mathcal{I}$ a family of targets and $I \in \mathcal{I}$ a target in this family. Define*

$$\mathcal{M}^{(I)}(D) := \{f^{(I)} \mid (f^{(J)})_{J \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(D)\} ,$$

the projection of $\mathcal{M}_{\mathcal{I}}(D)$ to the density component associated with the intervention target I. Then, $\mathcal{M}^{(I)}(D) = \mathcal{M}(D^{(I)})$.

Proof The inclusion “$\subset$” is immediately clear from Definition 7. It remains to show “$\supset$”.

Let $f \in \mathcal{M}(D^{(I)})$. Since $D^{(I)} \subset D$, f also obeys the Markov property of D; this means $f \in \mathcal{M}(D)$. Set $\tilde{f}_I(x_I) := f(x_I)$; since $f \in \mathcal{M}(D^{(I)})$, the components of $\tilde{f}_I$ are independent. For $J \in \mathcal{I}$, $J \neq I$, let $\tilde{f}_J$ be an arbitrary level density on $\mathcal{X}_J$. By Lemma 8(i), we know that, for intervention variables $U_J \sim \tilde{f}_J$ ($J \in \mathcal{I}$),

$$\big(f(\cdot \mid \mathrm{do}_D(X_J = U_J))\big)_{J \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(D) \ ,$$

hence $f(\cdot \mid \mathrm{do}_D(X_I = U_I)) \in \mathcal{M}^{(I)}(D)$ by definition of $\mathcal{M}^{(I)}(D)$. Moreover, by construction of $\tilde{f}_I$, we have $f(x \mid \mathrm{do}_D(X_I = U_I)) = f(x)$ and hence $f \in \mathcal{M}^{(I)}(D)$. ■

Lemma 46 *Let D be a DAG, $f \in \mathcal{M}(D)$, and $A \subset [p]$. Then,*

$$\prod_{a \in A} f(x_a \mid x_{\mathrm{pa}(a)}) = f(x_A \mid x_{\mathrm{pa}(A)}).$$

Proof Let $\sigma \in S_p$ be a topological ordering of D. Then, for $a \in A$,

$$\mathrm{pa}(a) \subset \mathrm{pa}(A) \cup \big[A \cap \sigma^{-1}(\{1, \ldots, a-1\})\big] \tag{8}$$

holds: every $b \in \mathrm{pa}(a)$ either lies in A^c and hence in $\mathrm{pa}(A)$ by the definition given in Appendix A.1, or in $A \cap \sigma^{-1}(\{1, \ldots, a-1\})$ by the definition of a topological ordering.

Hence we conclude

$$f(x_A \mid x_{\mathrm{pa}(A)}) = \prod_{a \in A} f(x_a \mid x_{A \cap \sigma^{-1}(\{1,\ldots,a-1\})}, x_{\mathrm{pa}(A)}) = \prod_{a \in A} f(x_a \mid x_{\mathrm{pa}(a)});$$

the first equality is the usual factorization of a density, the second equality follows from the Markov properties of f and Equation (8). ■

Lemma 47 *Let $\mathcal{I}$ be a family of targets. Assume D_1 and D_2 are DAGs with the same skeleton and the same v-structures such that $D_1^{(I)}$ and $D_2^{(I)}$ have the same skeleton for all $I \in \mathcal{I}$. Moreover, let $a \longrightarrow b \in D_1$. If there is some $I \in \mathcal{I}$ such that $|I \cap \{a, b\}| = 1$, then the arrow is also present in D_2: $a \longrightarrow b \in D_2$.*

Proof Since D_1 and D_2 have the same skeleton, we have at least $a \cdots\cdots b \in D_2$. Suppose $a \longleftarrow b \in D_2$. If $a \in I$, $b \notin I$, a and b are adjacent in $D_1^{(I)}$, but not in $D_2^{(I)}$, hence $D_1^{(I)}$ and $D_2^{(I)}$ have a different skeleton, a contradiction. On the other hand, if $a \notin I$ but $b \in I$, a and b are not adjacent in $D_1^{(I)}$, but in $D_2^{(I)}$, a contradiction, too. ■

Proof of Theorem 10 *(i) $\Rightarrow$ (ii):* Let $I \in \mathcal{I}$, and let $\mathcal{M}^{(I)}(D_1)$ and $\mathcal{M}^{(I)}(D_2)$ be defined as in Lemma 45. By Definition 9 of interventional Markov equivalence, it follows that $\mathcal{M}^{(I)}(D_1) = \mathcal{M}^{(I)}(D_2)$; hence $\mathcal{M}(D_1^{(I)}) = \mathcal{M}(D_2^{(I)})$ by Lemma 45.

(ii) $\Rightarrow$ (iii): this implication follows from Theorem 3.

(iii) ⇒ (iv): Let $a \longrightarrow b \in D_1$ be an arrow. Since $\mathcal{I}$ is conservative, there is some $I \in \mathcal{I}$ such that $b \notin I$. For this I, $a \longrightarrow b \in D_1^{(I)}$, so $a \cdots\cdots b \in D_2^{(I)}$ by assumption and hence $a \cdots\cdots b \in D_2$ because of $D_2^{(I)} \subset D_2$. Similarly, we can show the implication $a \longrightarrow b \in D_2 \;\Rightarrow\; a \cdots\cdots b \in D_1$, what proves that D_1 and D_2 have the same skeleton.

It remains to show that D_1 and D_2 also have the same v-structures. Let $a \longrightarrow b \longleftarrow c$ be a v-structure of D_1. There is some $I \in \mathcal{I}$ that does not contain b; $a \longrightarrow b \longleftarrow c$ is then an induced subgraph of $D_1^{(I)}$ and hence by assumption also of $D_2^{(I)}$. By consequence, $a \longrightarrow b \longleftarrow c$ is also an induced subgraph of D_2 since D_2 has the same skeleton as D_1. The argument is of course symmetric w.r.t. exchanging D_1 and D_2.

(iv) ⇒ (i): Let $(f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(D_1)$. By Lemma 8(ii), there is some density $f \in \mathcal{M}(D_1)$ and some intervention setting $\mathcal{S} = \{(I, \tilde{f}_I)\}_{I \in \mathcal{I}}$ such that $f^{(I)}(\cdot) = f(\cdot | \operatorname{do}_{D_1}(X_I = U_I))$ for random variables $U_I \sim \tilde{f}_I$, $I \in \mathcal{I}$.

The truncated factorization in Equation (1) tells us

$$\begin{aligned} f(x \mid \operatorname{do}_{D_1}(X_I = U_I)) &= \prod_{a \notin I} f(x_a \mid x_{\mathrm{pa}_{D_1}(a)}) \prod_{a \in I} \tilde{f}_I(x_a) = f(x) \prod_{a \in I} \frac{\tilde{f}_I(x_a)}{f(x_a \mid x_{\mathrm{pa}_{D_1}(a)})} \\ &= f(x) \frac{\tilde{f}_I(x_I)}{f(x_I \mid x_{\mathrm{pa}_{D_1}(I)})}. \end{aligned} \tag{9}$$

The last step uses Lemma 46.

We now claim that $\mathrm{pa}_{D_1}(I) = \mathrm{pa}_{D_2}(I)$. Indeed, if $b \in I$ and $a \in \mathrm{pa}_{D_1}(b) \setminus I$, $a \longrightarrow b$ is an arrow in D_1 with $|I \cap \{a, b\}| = 1$, hence $a \longrightarrow b \in D_2$ by Lemma 47 and therefore $a \in \mathrm{pa}_{D_2}(I)$; the argument is symmetric w.r.t. exchanging D_1 and D_2. It follows that $f(x_I | x_{\mathrm{pa}_{D_1}(I)}) = f(x_I | x_{\mathrm{pa}_{D_2}(I)})$, and by repeating the calculation in (9) for D_2 instead of D_1, we find $f(x | \operatorname{do}_{D_1}(X_I = U_I)) = f(x | \operatorname{do}_{D_2}(X_I = U_I))$.

Since this equality is true for all $I \in \mathcal{I}$, we have $f^{(I)}(\cdot) = f(\cdot | \operatorname{do}_{D_2}(X_I = U_I))$ for all $I \in \mathcal{I}$, so $(f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(D_2)$ by Lemma 8(i), which proves $\mathcal{M}_{\mathcal{I}}(D_1) \subset \mathcal{M}_{\mathcal{I}}(D_2)$. The other direction is completely analogous. ■

Points (i) to (iii) are even equivalent under non-conservative families of targets. The proof is more difficult in this case though.

B.2 Proofs for Section 3

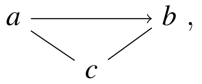
All statements of Section 3.2 are similar to analogous statements for the observational case developed by Andersson et al. (1997). Some of the proofs given there are even literally valid also for our interventional setting; in such cases, we will not repeat them here, but just refer to the original ones. However, in most cases, the generalization from the observational to the interventional case is not obvious and requires adapted techniques presented in this section. Here, $\mathcal{I}$ always stands for a conservative family of targets.

First, we show that for some DAG D, $\mathcal{E}_{\mathcal{I}}(D)$ is a chain graph (Proposition 15). For that purpose, we define $\mathcal{E}_{\mathcal{I}}(D)^*$ as the smallest chain graph containing $\mathcal{E}_{\mathcal{I}}(D)$. $\mathcal{E}_{\mathcal{I}}(D)^*$ is obtained from $\mathcal{E}_{\mathcal{I}}(D)$ by converting all arrows that are part of a directed cycle in $\mathcal{E}_{\mathcal{I}}(D)$ into lines (Andersson et al., 1997). We first state a couple of properties of $\mathcal{E}_{\mathcal{I}}(D)$ and $\mathcal{E}_{\mathcal{I}}(D)^*$ (Lemma 48), and then show that $\mathcal{E}_{\mathcal{I}}(D)^* = \mathcal{E}_{\mathcal{I}}(D)$ (Proposition 15).

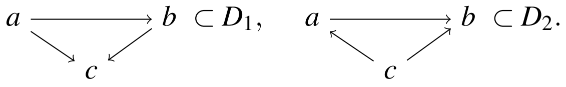
Lemma 48 (adapted from Andersson et al., 1997) *Let D be a DAG. Then:*

(i) *$\mathcal{E}_{\mathcal{I}}(D)$ has no induced subgraph of the form $a \longrightarrow b —— c$.*

(ii) *If $\mathcal{E}_{\mathcal{I}}(D)$ has an induced subgraph of the form*



then there exist $D_1, D_2 \in [D]_{\mathcal{I}}$ such that



(iii) *$\mathcal{E}_{\mathcal{I}}(D)^*$ has the same v-structures as D (and hence as $\mathcal{E}_{\mathcal{I}}(D)$).*

(iv) *$\mathcal{E}_{\mathcal{I}}(D)$ and $\mathcal{E}_{\mathcal{I}}(D)^*$ do not have any undirected chordless k-cycle of length $k \geq 4$.*

(v) *$\mathcal{E}_{\mathcal{I}}(D)^*$ has no induced subgraph of the form $a \longrightarrow b —— c$.*

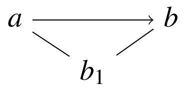
(vi) *If two vertices a and b are adjacent in $\mathcal{E}_{\mathcal{I}}(D)^*$ and there is some $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$, then the edge between a and b is directed in $\mathcal{E}_{\mathcal{I}}(D)$ and $\mathcal{E}_{\mathcal{I}}(D)^*$.*

Proof Points (i) to (v) correspond to Facts 1 to 5 of Andersson et al. (1997) where these properties were proven for observational essential graphs. A thorough inspection of the proofs given there reveals that they only make use of the fact that two Markov equivalent DAGs have the same skeleton and the same v-structures, which is also true in the interventional case by Theorem 10. Thanks to this, the proofs of Andersson et al. (1997) can be literally used here. (Note that the inverse implication also holds in the observational case, but not in the interventional one; see the discussion after Theorem 10.)

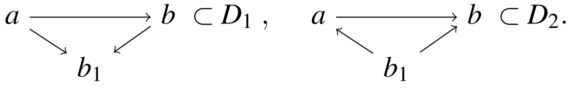
It remains to prove point (vi). The edge between a and b in $\mathcal{E}_{\mathcal{I}}(D)$ is directed since the arrow between a and b is $\mathcal{I}$-essential in D by Corollary 13. It remains to show that the edge is also directed in $\mathcal{E}_{\mathcal{I}}(D)^*$, that is, to show that it is *not* part of a directed cycle in $\mathcal{E}_{\mathcal{I}}(D)$.

Let's suppose, for the sake of contradiction, that the edge between a and b is part of a directed cycle $\gamma = (a, b \equiv b_0, b_1, \ldots, b_k \equiv a)$ in $\mathcal{E}_{\mathcal{I}}(D)$. W.l.o.g., we can assume that $a \longrightarrow b \in \mathcal{E}_{\mathcal{I}}(D)$, and that γ is the *shortest* such cycle containing a directed edge with one end point in I and the other one outside I.

Case 1: $k = 2$. Then γ is of the form



since two or three directed edges would imply the existence of a digraph with a cycle in the equivalence class of D. By point (ii), there are DAGs D_1 and D_2 in $[D]_{\mathcal{I}}$ such that



The condition $|I \cap \{a,b\}| = 1$ leaves four possibilities:

a) $a \in I; b, b_1 \notin I$: then,

$$\begin{tikzcd} a \arrow[rr] \arrow[dr] & & b \arrow[dl] & \subset D_1^{(I)}, & a \arrow[rr] & & b & \subset D_2^{(I)}; \\ & b_1 & & & & b_1 \arrow[ur] & & \end{tikzcd}$$

b) $a, b_1 \in I; b \notin I$: then,

$$\begin{tikzcd} a \arrow[rr] & & b & \subset D_1^{(I)}, & a \arrow[rr] & & b & \subset D_2^{(I)}; \\ & b_1 & & & & b_1 \arrow[ur] & & \end{tikzcd}$$

c) $b \in I; a, b_1 \notin I$: then,

$$\begin{tikzcd} a \arrow[dr] & & b \arrow[dl] & \subset D_1^{(I)}, & a & & b & \subset D_2^{(I)}; \\ & b_1 & & & & b_1 \arrow[ul] & & \end{tikzcd}$$

d) $b, b_1 \in I; a \notin I$: then,

$$\begin{tikzcd} a & & b & \subset D_1^{(I)}, & a & & b & \subset D_2^{(I)}. \\ & b_1 & & & & b_1 \arrow[ul] & & \end{tikzcd}$$

In all four cases, $(D_1^{(I)})^u \neq (D_2^{(I)})^u$, hence $D_1 \not\sim_{\mathcal{I}} D_2$, a contradiction.

Case 2: $k \geq 3$. Let i be the smallest index such that $b_i \text{—} b_{i+1} \in \mathcal{E}_{\mathcal{I}}(D)$ (there must be such an index, otherwise γ would be a directed cycle in D).

Case 2.1: $i = 0$. Since $a \longrightarrow b \text{—} b_1$ cannot be an induced subgraph of $\mathcal{E}_{\mathcal{I}}(D)$ by point (i), we must have $a \cdots b_1 \in \mathcal{E}_{\mathcal{I}}(D)$. More precisely, we must have $a \longrightarrow b_1 \in \mathcal{E}_{\mathcal{I}}(D)$, otherwise (a, b, b_1, a) would form a shorter directed cycle than γ, in contradiction to the assumption. This means that there exist DAGs $D_1, D_2 \in [D]_{\mathcal{I}}$ such that

$$\begin{tikzcd} a \arrow[rr] \arrow[dr] & & b \arrow[dl] & \subset D_1, & a \arrow[rr] \arrow[dr] & & b & \subset D_2. \\ & b_1 & & & & b_1 \arrow[ur] & & \end{tikzcd}$$

Again, the condition $|I \cap \{a, b\}| = 1$ leaves four possibilities:

a) $a \in I; b, b_1 \notin I$: then,

$$\begin{tikzcd} a \arrow[rr] \arrow[dr] & & b \arrow[dl] & \subset D_1^{(I)}, & a \arrow[rr] \arrow[dr] & & b & \subset D_2^{(I)}; \\ & b_1 & & & & b_1 \arrow[ur] & & \end{tikzcd}$$

b) $a, b_1 \in I; b \notin I$: then,

$$\begin{tikzcd} a \arrow[rr] & & b & \subset D_1^{(I)}, & a \arrow[rr] & & b & \subset D_2^{(I)}; \\ & b_1 & & & & b_1 \arrow[ur] & & \end{tikzcd}$$

c) $b \in I; a, b_1 \notin I$: then,

$$\begin{tikzcd} a \arrow[dr] & & b \arrow[dl] & \subset D_1^{(I)}, & a \arrow[dr] & & b & \subset D_2^{(I)}; \\ & b_1 & & & & b_1 & & \end{tikzcd}$$

d) $b, b_1 \in I; a \notin I$: then,

$$\begin{tikzcd} a & & b & \subset D_1^{(I)}, & a & & b & \subset D_2^{(I)}. \\ & b_1 & & & & b_1 & & \end{tikzcd}$$

Cases *b)* and *c)* are not compatible with the condition $(D_1^{(I)})^u = (D_2^{(I)})^u$. In cases *a)* and *d)*, the arrow $a \longrightarrow b_1$ is part of a directed cycle $(a, b_1, b_2, \ldots, b_k \equiv a)$, furthermore $|I \cap \{a, b_1\}| = 1$; this contradicts the assumption of minimality of the larger cycle γ.

Case 2.2: $i \geq 1$. Since $b_{i-1} \longrightarrow b_i \;—\; b_{i+1}$ cannot be an induced subgraph of $\mathcal{E}_{\mathcal{I}}(D)$, we must have $b_{i-1} \cdots\cdots b_{i+1} \in \mathcal{E}_{\mathcal{I}}(D)$. Either $b_{i-1} \longleftarrow b_{i+1} \in \mathcal{E}_{\mathcal{I}}(D)$, that is

$$\begin{tikzcd} b_{i-1} \arrow[rr] & & b_i \arrow[dl, dash] \\ & b_{i+1} \arrow[ul] & \end{tikzcd} \subset \mathcal{E}_{\mathcal{I}}(D) \ ,$$

which would imply the existence of a digraph with a directed 3-cycle in the equivalence class of D, a contradiction. The other cases are $b_{i-1} \longrightarrow b_{i+1} \in \mathcal{E}_{\mathcal{I}}(D)$ or $b_{i-1} \;—\; b_{i+1} \in \mathcal{E}_{\mathcal{I}}(D)$ which would mean that $a \longrightarrow b$ would be part of a shorter directed cycle $(a, b \equiv b_0, \ldots, b_{i-1}, b_{i+1}, \ldots, b_k \equiv a)$, contradicting the assumption of minimality of the cycle γ. ■

Proof of Proposition 15 We only prove the first point; the second one is an immediate consequence of Lemma 48(iv). We have to show that $\mathcal{E}_{\mathcal{I}}(D) = \mathcal{E}_{\mathcal{I}}(D)^*$, that means that

$$a \;—\; b \in \mathcal{E}_{\mathcal{I}}(D)^* \ \Rightarrow \ a \;—\; b \in \mathcal{E}_{\mathcal{I}}(D) \ .$$

By Lemma 48(iv), all chain components of $\mathcal{E}_{\mathcal{I}}(D)^*$ are chordal. Let D_1 and D_2 be two DAGs that are obtained by orienting all chain components of $\mathcal{E}_{\mathcal{I}}(D)^*$ according to some perfect elimination ordering, such that $a \longrightarrow b \in D_1$ and $a \longleftarrow b \in D_2$; such orientations exist by Proposition 44 and Corollary 41.

We now claim that $D_1 \sim_{\mathcal{I}} D_2$ by verifying the criteria of Theorem 10(iv); it then follows that $a \;—\; b \in \mathcal{E}_{\mathcal{I}}(D)$ because of $D_1 \cup D_2 \subset \mathcal{E}_{\mathcal{I}}(D)$:

- By Proposition 44, D_1 and D_2 have the same skeleton and the same v-structures.
- $D_1^{(I)}$ and $D_2^{(I)}$ have the same skeleton for all $I \in \mathcal{I}$: suppose, for the sake of contradiction, that $(D_1^{(I)})^u$ has some edge $c \;—\; d$ that $(D_2^{(I)})^u$ has not. W.l.o.g., we then have $c \longrightarrow d \in D_1$, $c \longleftarrow d \in D_2$, $c \in I$, $d \notin I$. But then c and d are adjacent in $\mathcal{E}_{\mathcal{I}}(D)^*$ with $|I \cap \{c, d\}| = 1$, hence the edge between c and d must be oriented in $\mathcal{E}_{\mathcal{I}}(D)^*$ by point (vi) of Lemma 48, and hence it is not possible that this edge has two different orientations in D_1 and D_2 by their constructio■

Proof of Proposition 16 *"⇐":* By the construction of $\mathcal{E}_{\mathcal{I}}(D)$, we know that $D \subset \mathcal{E}_{\mathcal{I}}(D)$ and $D^u = \mathcal{E}_{\mathcal{I}}(D)^u$. Furthermore, D has the same v-structures as $\mathcal{E}_{\mathcal{I}}(D)$. Let D' be another digraph that is obtained by orienting all chain components of $\mathcal{E}_{\mathcal{I}}(D)$ according to a perfect elimination ordering; by Proposition 44, D' is acyclic and has the same v-structures as $\mathcal{E}_{\mathcal{I}}(D)$ and hence as D. It remains to show that $D^{(I)}$ and $D'^{(I)}$ have the same skeleton for all $I \in \mathcal{I}$; this can be done similarly to the proof of Proposition 15.

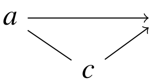
"⇒": let D' be a DAG with $D' \sim_{\mathcal{I}} D$. In particular, D' and D have the same skeleton and the same v-structures, so D' also has the same skeleton and the same v-structures as $\mathcal{E}_{\mathcal{I}}(D)$. It follows, with Proposition 44, that D' is oriented according to a perfect elimination ordering on all chain components of $\mathcal{E}_{\mathcal{I}}(D)$. ■

Lemma 49 *Let D be a DAG and $a \longrightarrow b$ an $\mathcal{I}$-essential arrow in D. Then $a \longrightarrow b$ is strongly $\mathcal{I}$-protected in $\mathcal{E}_{\mathcal{I}}(D)$.*

This lemma is an auxiliary result needed to prove Theorem 18. In its proof, we first show the weaker statement that every $\mathcal{I}$-essential arrow of D is $\mathcal{I}$-protected in $\mathcal{E}_{\mathcal{I}}(D)$.

Definition 50 (Protection) *Let G be a graph. An arrow $a \longrightarrow b \in G$ is **$\mathcal{I}$-protected** in G if there is some intervention target $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$, or $\mathrm{pa}_G(a) \neq \mathrm{pa}_G(b) \setminus \{a\}$.*

This definition is again a generalization of the notion of protection of Andersson et al. (1997); for $\mathcal{I} = \{\emptyset\}$, we gain back their definition. A *strongly* $\mathcal{I}$-protected arrow (Definition 14) is also $\mathcal{I}$-protected. In a chain graph G, an arrow $a \longrightarrow b$ is $\mathcal{I}$-protected if and only if there is some $I \in \mathcal{I}$ such that $|I \cap \{a,b\}| = 1$, or the arrow $a \longrightarrow b$ occurs in at least one subgraph of the form (a), (b), (c) in the notation of Definition 14, or in a subgraph of the form (d') (Andersson et al., 1997), where

(d'):  .

Proof of Lemma 49 As foreshadowed, we prove this lemma in two steps, corresponding to Facts 6 and 7 of Andersson et al. (1997): in a first step, we show that $a \longrightarrow b$ must be $\mathcal{I}$-protected, in a second step, we strengthen the result by showing that it must even be *strongly* $\mathcal{I}$-protected. For notational convenience, we abbreviate $G := \mathcal{E}_{\mathcal{I}}(D)$. We skip some steps of the proof that can be literally copied from proofs in Andersson et al. (1997).

Suppose, for the sake of contradiction, that $a \longrightarrow b$ is not $\mathcal{I}$-protected. Let D_1 be a digraph that is gained by orienting all chain components of G according to a perfect elimination ordering, where the edges of $T_G(a)$ and $T_G(b)$ are oriented such that all edges point away from a or b, respectively. Then D_1 is acyclic and $\mathcal{I}$-equivalent to D by Proposition 16.

Let D_2 be another digraph, differing from D_1 only by the orientation of the edge between a and b. It can be shown that D_2 is acyclic too (Andersson et al., 1997, proof of Fact 6). We now claim that $D_1 \sim_{\mathcal{I}} D_2$:

- D_1 and D_2 clearly have the same skeleton.
- D_1 and D_2 have the same v-structures. Otherwise, there would be some v-structure $c \longrightarrow a \longleftarrow b$ in D_2, or some v-structure $a \longrightarrow b \longleftarrow c$ in D_1. In both cases, this would imply $\mathrm{pa}_G(a) \neq \mathrm{pa}_G(b) \setminus \{a\}$, contradicting the assumption: in the first case, $c \notin T_G(a)$ by construction (all edges of $T_G(a)$ point away from a in D_2), so $c \in \mathrm{pa}_G(a)$, but $c \notin \mathrm{pa}_G(b)$; in the second case, $c \in \mathrm{pa}_G(b)$, but $c \notin \mathrm{pa}_G(a)$ by analogous arguments.
- $(D_1^{(I)})^u = (D_2^{(I)})^u$ for all $I \in \mathcal{I}$. Otherwise, there would be some $I \in \mathcal{I}$ such that the skeletons of $D_1^{(I)}$ and $D_2^{(I)}$ differ in the edge between a and b. This could only happen if $|I \cap \{a,b\}| = 1$, in contradiction with the assumption.

Hence, since $D_1, D_2 \in [D]_{\mathcal{I}}$, we have $D_1 \cup D_2 \subset G$ and thus $a - b \in G$, a contradiction. This proves that $a \longrightarrow b$ is $\mathcal{I}$-protected in G.

In the second step, we show that $a \longrightarrow b$ is even *strongly* $\mathcal{I}$-protected. If this was not the case, $a \longrightarrow b$ would occur in configuration (d') in G, but *not* in configuration (a), (b), (c) or (d) (see the comment following Definition 50). Define $P_a := \{d \in T_G(a) \mid d \longrightarrow b \in G\}$. It can be shown that P_a is a clique $G[T_G(a)]$ (Andersson et al., 1997, proof of Fact 7).

Let D_1 be the DAG that we get by orienting all chain components of G according to a perfect elimination ordering, such that, additionally,

- all edges of $D_1[T_G(b)]$ point away from b,
- all edges of $D_1[P_a]$ point towards a,
- and all other edges of $D_1[T_G(a)]$ point away from a.

Such an orientation exists by Corollary 42. Let D_2 be the digraph that we get by changing the orientation of the edge $a \longrightarrow b$ in D_1; as in the first part, it can be shown that D_2 is acyclic (Andersson et al., 1997, proof of Fact 7). Again, we claim that $D_1 \sim_{\mathcal{I}} D_2$:

- D_1 and D_2 clearly have the same skeleton.
- D_1 and D_2 have the same v-structures. Otherwise, there would be some v-structure $d \longrightarrow a \longleftarrow b$ in D_2, or a v-structure $a \longrightarrow b \longleftarrow d$ in D_1. In the first case, $d \notin P_a$ (otherwise, $d \longrightarrow b \in G$ by definition of P_a, and hence $d \longrightarrow b \in D_2$ since $D_2 \subset G$), and $d \notin T_G(a) \setminus P_a$ by construction (edges in $T_G(a) \setminus P_a$ point away from a in D_2), hence $d \longrightarrow a \in G$ and $a \longrightarrow b$ is in configuration (a) in G; in the second case, $d \notin T_G(b)$ by construction (all edges of $T_G(b)$ point away from b in D_1), so $a \longrightarrow b$ is in configuration (b) (notation of Definition 14) in G. Both cases contradict the assumption.
- Exactly as in the first part, $(D_1^{(I)})^u = (D_2^{(I)})^u$ for all $I \in \mathcal{I}$.

We can conclude that, since $D_1, D_2 \in [D]_{\mathcal{I}}$, $D_1 \cup D_2 \subset G$, so a — $b \in G$, a contradiction. ■

Proof of Theorem 18 *"⇒":* (i) and (ii) follow from Proposition 15, (iii) from Lemma 48(v), (iv) from Corollary 13 and (v) from Lemma 49.

"⇐": Consider the set $\mathbf{D}(G)$ of all DAGs that can be obtained by orienting the chain components of G according to a perfect elimination ordering; we have $\bigcup \mathbf{D}(G) \subset G$. On the other hand, for each undirected edge a — $b \in G$, there are DAGs D_1 and D_2 in $\mathbf{D}(G)$ such that $a \longrightarrow b \in D_1$, $a \longleftarrow b \in D_2$ (Corollary 41), hence $G \subset \bigcup \mathbf{D}(G)$. Together, we find $G = \bigcup \mathbf{D}(G)$.

We claim that $D_1 \sim_{\mathcal{I}} D_2$ for any two DAGs $D_1, D_2 \in \mathbf{D}(G)$:

- D_1 and D_2 have the same skeleton and the same v-structures by Proposition 44.
- $(D_1^{(I)})^u = (D_2^{(I)})^u$ for all $I \in \mathcal{I}$. Otherwise, there would be arrows $a \longrightarrow b \in D_1$, $a \longleftarrow b \in D_2$, and some $I \in \mathcal{I}$ such that $|I \cap \{a, b\}| = 1$; this would mean that a — $b \in G$ although $|I \cap \{a, b\}| = 1$, contradicting property (iv).

Let $D \in \mathbf{D}(G)$. We have shown that $\mathbf{D}(G) \subset [D]_{\mathcal{I}}$, hence $G = \bigcup \mathbf{D}(G) \subset \mathcal{E}_{\mathcal{I}}(D)$. It remains to show that $G \supset \mathcal{E}_{\mathcal{I}}(D)$.

Assume, for the sake of contradiction, that G has some arrow $a \longrightarrow b$ where $\mathcal{E}_{\mathcal{I}}(D)$ has an undirected edge a — b. According to property (v), $a \longrightarrow b$ is strongly $\mathcal{I}$-protected in G. If there was some $I \in \mathcal{I}$ such that $|I \cap \{a, b\}| = 1$, the edge between a and b was also directed in $\mathcal{E}_{\mathcal{I}}(D)$ by Corollary 13, a contradiction. Hence $a \longrightarrow b$ occurs in G in one of the configurations depicted in Definition 14. Exactly as in the proof of Theorem 4.1 of Andersson et al. (1997), we can construct a contradiction for each of the four configurations. Although the proof given there can be used literally, we reproduce it here since since we will use the following steps again in the proof of Lemma 22.

We assume w.l.o.g. that $T_G(a)$ is minimal in

$$A := \{T \in \mathbf{T}(G) \mid \exists\, a \in T, b \in V(G) : a \longrightarrow b \in G, a \frac{\quad}{} b \in \mathcal{E}_{\mathcal{I}}(D)\}$$

w.r.t. $\preceq_G$, and that $T_G(b)$ is minimal in

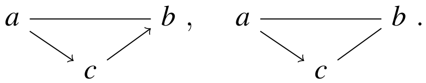
$$B := \{T \in \mathbf{T}(G) \mid \exists\, a \in T(a), b \in T : a \longrightarrow b \in G, a \frac{\quad}{} b \in \mathcal{E}_{\mathcal{I}}(D)\}.$$

Each configuration (a) to (d) of Definition 14 leads to a contradiction (c, c_1 and c_2 denote the vertices involved in the respective configuration):

(a) Because of the minimality of $T_G(a)$, $c \longrightarrow a$ must be oriented in $\mathcal{E}_{\mathcal{I}}(D)$, hence $c \longrightarrow a \frac{\quad}{} b$ is an induced subgraph of $\mathcal{E}_{\mathcal{I}}(D)$, contradicting Lemma 48(i).

(b) $a \longrightarrow b \longleftarrow c$ is then a v-structure in D, hence it is also a v-structure in $\mathcal{E}_{\mathcal{I}}(D)$, that means $a \longrightarrow b \in \mathcal{E}_{\mathcal{I}}(D)$, a contradiction.

(c) Because of the minimality of $T_G(b)$, the edge between a and c must be oriented in $\mathcal{E}_{\mathcal{I}}(D)$, so the vertices a, b and c are in one of the following configurations in $\mathcal{E}_{\mathcal{I}}(D)$:



Both possibilities violate Proposition 15(i).

(d) The v-structure $c_1 \longrightarrow b \longleftarrow c_2$ of D is also a v-structure of $\mathcal{E}_{\mathcal{I}}(D)$, hence $\mathcal{E}_{\mathcal{I}}(D)$ has two directed 3-cycles (c_1, b, a, c_1) and (c_2, b, a, c_2), a contradiction. ■

Proof of Lemma 20

(i) This immediately follows from Theorem 18(iii).

(ii) Let $a \longrightarrow b$ be an arrow in $\mathcal{E}_{\mathcal{I}}(D)$; by Theorem 18(v), it is strongly $\mathcal{I}$-protected in $\mathcal{E}_{\mathcal{I}}(D)$. If there is some $I \in \mathcal{I}$ such that $|I \cap \{a, b\}| = 1$, the arrow is by definition also strongly $\mathcal{I}$-protected in G. Otherwise, $a \longrightarrow b$ occurs in one of the configurations (a) to (d) of Definition 14 in $\mathcal{E}_{\mathcal{I}}(D)$. In configurations (a) to (c), the other arrows involved ($a \longleftarrow c$; $c \longrightarrow b$; or $a \longrightarrow c$ and $c \longrightarrow b$, resp.) are also present in G, hence $a \longrightarrow b$ is strongly $\mathcal{I}$-protected in G by the same configuration as in $\mathcal{E}_{\mathcal{I}}(D)$.

It remains to show that if $a \longrightarrow b$ is in configuration (d) in $\mathcal{E}_{\mathcal{I}}(D)$, it is also strongly $\mathcal{I}$-protected in G. In D, the vertices $\{a, c_1, c_2\}$ as defined in Definition 14 can occur in one of the following configurations:

$$c_1 \longleftarrow a \longleftarrow c_2, \quad c_1 \longleftarrow a \longrightarrow c_2, \quad c_1 \longrightarrow a \longrightarrow c_2.$$

The first and the third case are symmetric w.r.t. exchanging c_1 and c_2, hence we only consider the first two. Table 2 lists all possible configurations for the vertices $\{a, c_1, c_2\}$ in the graph G according to the condition $D \subset G \subset \mathcal{E}_{\mathcal{I}}(D)$. There is only one possibility for the arrow $a \longrightarrow b$ not to occur in one of the configurations (a) to (d) of Definition 14, and hence not being strongly $\mathcal{I}$-protected in G; however, the corresponding subgraph of $\{a, c_1, c_2\}$, $c_1 \frac{\quad}{} a \longleftarrow c_2$, is forbidden by Definition 19.

(iii) According to Theorem 10, we have to check the following properties:

- D_1 and D_2 have the same skeleton, namely $D_1^u = D_2^u = G^u$.
- D_1 and D_2 have the same v-structures: let $a \longrightarrow b \longleftarrow c$ be a v-structure in D_1. This v-structure is then also present in $\mathcal{E}_{\mathcal{I}}(D_1)$. Because of $D_2 \subset G \subset \mathcal{E}_{\mathcal{I}}(D_1)$, we find it also in G and in D_2. The argument is completely symmetric w.r.t. exchanging D_1 and D_2.
- For all $I \in \mathcal{I}$, $D_1^{(I)}$ and $D_2^{(I)}$ have the same skeleton: assume, for the sake of contradiction, that there is some $I \in \mathcal{I}$ and an edge $a \;\text{---}\; b$ that is present in $(D_1^{(I)})^u$, but not in $(D_2^{(I)})^u$. W.l.o.g., we can assume that $a \longrightarrow b \in D_1$, $a \longleftarrow b \in D_2$, $a \in I$, $b \notin I$. Because of Theorem 18(iv), we then have $a \longrightarrow b \in \mathcal{E}_{\mathcal{I}}(D_1)$ and $a \longleftarrow b \in \mathcal{E}_{\mathcal{I}}(D_2)$; however, this is not compatible with the requirements $G \subset \mathcal{E}_{\mathcal{I}}(D_1)$ and $G \subset \mathcal{E}_{\mathcal{I}}(D_2)$. ■

Proof of Lemma 21 If $a \longrightarrow b \in \mathcal{E}_{\mathcal{I}}(D)$, it would be strongly $\mathcal{I}$-protected by Theorem 18(v), and hence also strongly $\mathcal{I}$-protected in G by Lemma 20(ii), contradicting the assumption. Therefore, $a \;\text{---}\; b \in \mathcal{E}_{\mathcal{I}}(D)$ and hence $D \subset G' \subset \mathcal{E}_{\mathcal{I}}(D)$.

Suppose that G' contains an induced subgraph of the form $c \longrightarrow d \;\text{---}\; e$. Since G does not contain such an induced subgraph, it must be of the form $c \longrightarrow a \;\text{---}\; b$ or $c \longrightarrow b \;\text{---}\; a$ in G'. In both cases, $a \longrightarrow b$ is then strongly $\mathcal{I}$-protected in G, either by configuration (a) or (b), a contradiction. ■

Proof of Lemma 22 Let $D \subset G \subset \mathcal{E}_{\mathcal{I}}(D)$ be a partial $\mathcal{I}$-essential graph that only has strongly $\mathcal{I}$-protected arrows. We can literally use the second part of the proof of Theorem 18 to show $G \supset \mathcal{E}_{\mathcal{I}}(D)$; there, we only used the fact that every arrow in G is strongly $\mathcal{I}$-protected. ■

B.3 Proofs for Section 4

Proof of Proposition 25 *"⇒":*

(i) This claim follows from Corollary 42.

Induced subgraph of $\{a, c_1, c_2\}\ldots$		Configuration
$\ldots$in D	$\ldots$in G	of $a \longrightarrow b$ in G
$c_1 \longleftarrow a \longleftarrow c_2$	$c_1 \longleftarrow a \longleftarrow c_2$	(c)
	$c_1 \;\text{---}\; a \longleftarrow c_2$	—
	$c_1 \longleftarrow a \;\text{---}\; c_2$	(c)
	$c_1 \;\text{---}\; a \;\text{---}\; c_2$	(d)
$c_1 \longleftarrow a \longrightarrow c_2$	$c_1 \longleftarrow a \longrightarrow c_2$	(c)
	$c_1 \;\text{---}\; a \longrightarrow c_2$	(c)
	$c_1 \longleftarrow a \;\text{---}\; c_2$	(c)
	$c_1 \;\text{---}\; a \;\text{---}\; c_2$	(d)

Table 2: Possible configurations for the vertices $\{a, c_1, c_2\}$ in the proof of Lemma 20(ii). The labels in the last column refer to the configurations of Definition 14.

(ii) Suppose that there is some vertex $a \in N \setminus C$, that is a vertex $a \in N$ with $a \longleftarrow v \in D$. D' would have a directed cycle if $u \longleftarrow a \in D$, so $u \longrightarrow a \in D$. But then, $u \longrightarrow a \longleftarrow v$ is a v-structure in D, hence also in G, and consequently $a \notin \mathrm{ne}_G(v)$, a contradiction.

(iii) Assume that $\gamma = (v \equiv a_0, a_1, \ldots, a_k \equiv u)$ is a shortest path from v to u in G that does not intersect with C. We claim that γ is a directed path in D, which means that D' has a directed cycle, a contradiction.

Suppose that the claim is wrong, and let $a_i \longleftarrow a_{i+1} \in D$ be the first edge of (the chain) γ that points away from u in D; $i \geq 1$ holds by the assumption that, in particular, $a_1 \notin C$. $a_{i-1} \longrightarrow a_i \longleftarrow a_{i+1}$ cannot be an induced subgraph of D, otherwise it would also be present in G and hence γ would not be a *path* in G. Hence $a_{i-1} \cdots\cdots a_{i+1} \in G$; more precisely, $a_{i-1} \longleftarrow a_{i+1} \in G$ (and hence also in D), otherwise there would be a shorter path from v to u in G than γ that does not intersect with C. Because γ is a path in G, a_{i-1}, a_i and a_{i+1} can occur in G only in one of the following configurations:

$$\begin{tikzcd}
a_{i-1} \arrow[dr] & & a_{i+1} \arrow[ll] \arrow[dl, dash] & , & a_{i-1} \arrow[dr, dash] & & a_{i+1} \arrow[ll] \arrow[dl, dash] & . \\
& a_i & & & & a_i & &
\end{tikzcd}$$

However, both graphs cannot be an induced subgraph of the chain graph G.

"$\Leftarrow$": Since C is a clique in $G[T_G(v)]$, there is a DAG $D \in \mathbf{D}(G)$ with $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D\} = C$ by Proposition 44 and Corollary 42. It remains to show that D' is a DAG.

Assume, for the sake of contradiction, that D' has a directed cycle going through $u \longrightarrow v$. The return path from v to u, $\gamma = (v \equiv a_0, a_1, \ldots, a_k \equiv u)$, must come from a path in G and must therefore, by assumption, contain a vertex $a_i \in C$ ($i \geq 2$). Since $a_i \longrightarrow v \in D$ by construction, this means that D has a directed cycle $(a_0, a_1, \ldots, a_i, a_0)$, a contradiction.

Uniqueness of $\mathcal{E}_{\mathcal{I}}(D')$: Let $D_1, D_2 \in \mathbf{D}(G)$ with $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D_1\} = \{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D_2\} = C$, and set $D'_i := D_i + (u, v)$, $i = 1, 2$; we assume that $D'_1, D'_2 \in \mathbf{D}^+(G)$. To prove $D'_1 \sim_{\mathcal{I}} D'_2$, we have to check the following three points according to Theorem 10(iv):

- D'_1 and D'_2 obviously have the same skeleton.
- D'_1 and D'_2 have the same v-structures. We already know that D_1 and D_2 have the same v-structures. Let's assume, for the sake of contradiction, that (w.l.o.g.) D'_1 has a v-structure $u \longrightarrow v \longleftarrow a$ that D'_2 has not. In G, we must then have a line $a \mathrel{-\!\!-} v$, hence $a \in \mathrm{ne}_G(v)$. However, the arrow between a and v would then have the same orientation in D_1 and D_2 by construction, a contradiction.
- For all $I \in \mathcal{I}$, $D_1'^{(I)}$ and $D_2'^{(I)}$ have the same skeleton. If this was not the case, there would be some vertices $a, b \in [p]$ and some $I \in \mathcal{I}$ such that $a \longrightarrow b \in D'_1$, $a \longleftarrow b \in D'_2$ and $|I \cap \{a, b\}| = 1$. The arrow $u \longrightarrow v$ is part of D'_1 and D'_2 by construction, so the arrows between a and b must be present in D_1 and D_2; however, $D_1^{(I)}$ and $D_2^{(I)}$ would then not have the same skeleton, a contradiction. ■

Corollary 26 is an immediate consequence of Proposition 25 and the fact that we assume the score function to be decomposable, so we skip the proof here.

Proof of Lemma 27 Obviously, we have $D' \subset H$. To show $H \subset \mathcal{E}_{\mathcal{I}}(D')$, we look at some edge $a - b \in G$ with $a, b \notin T_G(v)$ and show that $a - b \in \mathcal{E}_{\mathcal{I}}(D')$. W.l.o.g., we can assume that $a \longrightarrow b \in D$. By Corollary 41, there exists a $D_2 \in \mathbf{D}(G)$ that has the same orientation of edges in $T_G(v)$, but an orientation of edges in $T_G(a)$ such that $a \longleftarrow b \in D_2$. By Proposition 25, we know that $D'_2 := D_2 + (u,v)$ is $\mathcal{I}$-equivalent to D', so in particular $a - b \in D' \cup D'_2 \subset \mathcal{E}_{\mathcal{I}}(D')$.

It remains to show that $a \longrightarrow b - c$ does not occur as an induced subgraph of H. The inserted arrow $u \longrightarrow v$ cannot be part of such a subgraph, since all other edges incident to v are oriented in H by construction. Since G has no such subgraph either (Theorem 18), it could only appear in H through one of the newly oriented edges of $T_G(v)$. This means that if H had an induced subgraph of the form $a \longrightarrow b - c$, the corresponding vertices would be in configuration $a - b - c$ in G; however, $c \in T_G(v)$ then, and so the edge between b and c would be oriented in H, a contradiction. ■

Proof of Proposition 28 *"⇒":*

(i) By Corollary 42, $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D\}$ is a clique, hence every subset—in particular, C—is a clique, too.

(ii) Assume that there is some $a \in C \setminus \mathrm{ad}_G(u)$; then $u \in \mathrm{ne}_G(v)$, otherwise $u \longrightarrow v - a$ would be an induced subgraph of G. Nevertheless, $a \in C$ means that $u \longrightarrow v \longleftarrow a$ is a v-structure in D, which should hence also be present in G.

"⇐": We only must prove the existence of the claimed $D \in \mathbf{D}(G)$, see the comment in the beginning of Section 4.2. We distinguish two cases:

- $u \longrightarrow v \in G$. The existence of the DAG $D \in \mathbf{D}(G)$ with the requested properties follows from Corollary 42.
- $u - v \in G$, hence $u - a \in G$ for all $a \in N$ because G is a chain graph. Therefore, $C \cup \{u\}$ is a clique in $G[\mathrm{ne}_G(v)]$, and the existence of the claimed D again follows from Corollary 42.

Uniqueness of $\mathcal{E}_{\mathcal{I}}(D')$: Let $D_1, D_2 \in \mathbf{D}(G)$ with $u \longrightarrow v \in D_1, D_2$ and $\{a \in \mathrm{ne}_G(v) \setminus \{u\} \mid a \longrightarrow v \in D_1\} = \{a \in \mathrm{ne}_G(v) \setminus \{u\} \mid a \longrightarrow v \in D_2\} = C$, and set $D'_i := D_i - (u,v)$, $i = 1,2$. To prove $D'_1 \sim_{\mathcal{I}} D'_2$, we have to check the following three points according to Theorem 10(iv):

- D'_1 and D'_2 have the same skeleton, namely $G^u - (u,v) - (v,u)$.
- D'_1 and D'_2 have the same v-structures. Otherwise, w.l.o.g., D'_1 would have a v-structure $a \longrightarrow b \longleftarrow c$ that D'_2 has not. D_1 and D_2 have the same v-structures, so $a \longrightarrow b \longleftarrow c$ is no induced subgraph of D_1; this implies $a = u$, $c = v$. Since D'_2 does not have the v-structure $u \longrightarrow b \longleftarrow v$, the vertices u, b and v must occur in configuration $u \longrightarrow b \longrightarrow v$ or $u \longleftarrow b \longrightarrow v$ in D'_2 (the configuration $u \longleftarrow b \longleftarrow v$ is not consistent with the acyclicity of D_2). However, all edges incident to v must have the same orientation in D'_1 and D'_2 by construction, a contradiction.
- Let $I \in \mathcal{I}$. Because of $(D_1^{(I)})^u = (D_2^{(I)})^u$ and $(D_i'^{(I)})^u = (D_i^{(I)})^u - (u,v) - (v,u)$ for $i = 1,2$, we have $(D_1'^{(I)})^u = (D_2'^{(I)})^u$. ■

Corollary 29 follows quickly from Proposition 28, and the proof of Lemma 30 is very similar to that of Lemma 27. Therefore we skip both proofs here and proceed with the proofs of Section 4.3.

Proof of Proposition 31 Note that we can write $N = \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u) = \mathrm{ne}_G(v) \cap \mathrm{ne}_G(u)$ because $u \,—\, v \in G$ and G is a chain graph.

"$\Rightarrow$":

(i) This follows from Corollary 42.

(ii) D and D' have the same skeleton; the same is true for $D^{(I)}$ and $D'^{(I)}$ for all $I \in \mathcal{I}$. To see the latter, assume that for some $I \in \mathcal{I}$, the intervention graphs $D^{(I)}$ and $D'^{(I)}$ have a different skeleton. Since D and D' only differ in the orientation of the arrow between u and v, the skeletons of $D^{(I)}$ and $D'^{(I)}$ can only differ in that u and v are adjacent in one of them and not adjacent in the other one. However, this would imply that $|I \cap \{u, v\}| = 1$, and hence the edge between u and v would be directed in G by Theorem 18(iv), contradicting the assumption of the proposition. Finally, D' has at least all v-structures that D has by construction.
As a consequence $D' \not\sim_{\mathcal{I}} D$ if and only if D' has *more* v-structures than D (Theorem 10). An additional v-structure in D' must be of the form $u \longrightarrow v \longleftarrow a$. The edge between v and a cannot be directed in G, otherwise $u \,—\, v \longleftarrow a$ would be an induced subgraph of G, which is forbidden by Theorem 18(iii). Hence $a \in \mathrm{ne}_G(v)$, or, more precisely, $a \in C \setminus N$.

(iii) If $N \setminus C$ is empty, the statement is trivial. Otherwise, assume that there is some shortest path $\gamma = (a_0, a_1, \ldots, a_k)$ from $N \setminus C$ to $C \setminus N$ in $G[\mathrm{ne}_G(v)]$ that has no vertex in $C \cap N$.
By definition of C, $a_k \longrightarrow v \in D$; furthermore, $u \longrightarrow a_0 \in D$ must hold, otherwise (v, a_0, u, v) would be a directed cycle in D'. Therefore, γ must not be a path from a_0 to a_k in D. Let $a_i \longleftarrow a_{i+1}$ be the first arrow in γ that points away from a_k in D. If $i = 0$, $u \longrightarrow a_0 \longleftarrow a_1$ would be a v-structure in D since $a_1 \notin N$: by assumption, $a_1 \notin N \cap C$, and $a_1 \notin N \setminus C$ because of the minimality of γ. Hence $i > 0$ (and $k > 1$) must hold, and $a_{i-1} \cdots\cdots a_{i+1}$ in D and G, otherwise there would be a v-structure in D. However, γ is not the *shortest* path with the requested properties then, a contradiction.

"$\Leftarrow$": From Proposition 43, we see that there exists a DAG D that has the requested properties, and in which, in addition, $\{a \in \mathrm{ne}_G(u) \mid a \longrightarrow u \in D\} = (C \cap N) \cup \{v\}$ (point (iv) of Proposition 43). The fact that $D' := D - (v, u) + (u, v) \not\sim_{\mathcal{I}} D$ can be seen by an argument very similar to the proof of point (ii) above; it remains to show that D has no v-u-path except (v, u). Suppose that $\gamma = (a_0 \equiv v, a_1, \ldots, a_k \equiv u)$, $k \geq 2$, is such a path. In particular, γ is then also a v-u-path in G, hence γ lies completely in $T_G(v)$.

If $k = 2$, then $a_1 \in N$, and so the vertices u, v and a_1 occur in one of the following configurations in D by Proposition 43:

$$\begin{tikzcd} v \arrow[rr] \arrow[dr] & & u \arrow[dl] \\ & a_1 & \end{tikzcd} , \quad \begin{tikzcd} v \arrow[rr] & & u \\ & a_1 \arrow[ul] \arrow[ur] & \end{tikzcd} .$$

Both configurations contradict the assumption that $\gamma = (v, a_1, u)$ forms a path in D. Thus we conclude $k \geq 3$, and we notice $a_{k-1} \in \mathrm{ne}_G(u) \setminus \{v\}$. If $a_{k-1} \in C$, $a_{k-1} \longrightarrow v \in D$, hence $(a_0, a_1, \ldots, a_{k-1}, a_0)$ would be a cycle in D. On the other hand, if $a_{k-1} \notin C$, we would have $a_{k-1} \longleftarrow u \in D$, so γ would not be a path in D.

Uniqueness of $\mathcal{E}_{\mathcal{I}}(D')$: Let $D_1, D_2 \in \mathbf{D}(G)$ with $u \longleftarrow v \in D_1, D_2$ and $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D_1\} = \{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D_2\} = C$, and set $D'_i := D_i - (v,u) + (u,v)$, $i = 1,2$; we assume that $D'_1, D'_2 \in \mathbf{D}^{\circlearrowleft}(G)$. As in the proofs of Proposition 25, we can check that $D'_1 \sim_{\mathcal{I}} D'_2$:

- D'_1 and D'_2 obviously have the same skeleton.
- D_1 and D_2 have the same v-structures. If this does not hold for D'_1 and D'_2, (w.l.o.g.) D'_1 must have a v-structure $u \longrightarrow v \longleftarrow a$ that D'_2 has not. Since $u \frac{\quad}{} v \longleftarrow a$ cannot be an induced subgraph of G, $a \in \mathrm{ne}_G(v)$; however, the edges between v and its neighbors are oriented in the same way in D'_1 and D'_2 by construction, a contradiction.
- For all $I \in \mathcal{I}$, $D_1'^{(I)}$ and $D_2'^{(I)}$ have the same skeleton: this can be seen by an argument very similar to that in the proof of Proposition 25. ■

Proof of Corollary 32 We have to show $\mathrm{pa}_D(v) = \mathrm{pa}_G(v) \cup C$ and $\mathrm{pa}_D(u) = \mathrm{pa}_G(u) \cup (C \cap N) \cup \{v\}$. The first identity is immediately clear. For the second identity, note that for any vertex $a \in C \cap N$, the arrow between a and u must be oriented as $a \longrightarrow u \in D$ because the other orientation would induce a 3-cycle. On the other hand, we have $a \longleftarrow u \in D$ for $a \in N \setminus C$ because a different orientation would induce a 3-cycle in D'. Finally, we also have $a \longleftarrow u \in D$ for any $a \in \mathrm{ne}_G(u) \setminus (\mathrm{ne}_G(v) \cup \{v\})$ since the other orientation would induce a v-structure $v \longrightarrow u \longleftarrow a$ in D. ■

Lemma 33 can be proven very similarly as Lemma 27. Finally, we finish this proof section with the proof of Proposition 34 characterizing a step of the turning phase of GIES for the case that we turn an $\mathcal{I}$-essential arrow in some representative $D \in \mathbf{D}(G)$. We will omit the proof of Lemma 35 since it can be proven similarly to Lemma 27.

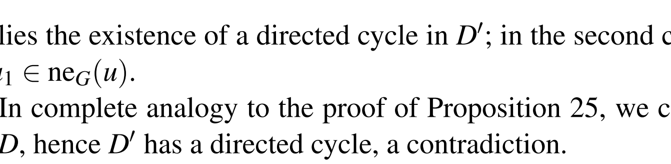
Proof of Proposition 34 When $v \longrightarrow u \in G$ (that is, u and v lie in different chain components), $N = \mathrm{ne}_G(v) \cap \mathrm{ad}_G(u) = \mathrm{ne}_G(v) \cap \mathrm{pa}_G(u)$ holds because G is a chain graph.

"⇒":

(i) This point follows from Corollary 42.

(ii) If this was not true, D' would have a cycle of the form (u,v,a,u) for some $a \in N$ since $N \subset \mathrm{pa}_G(u)$.

(iii) Suppose that the path $\gamma = (a_0 \equiv v, a_1, \ldots, a_k \equiv u)$ is a shortest counterexample of a path without vertex in $C \cup \mathrm{ne}_G(u)$.
Assume that $k = 2$. Since u and v lie in different chain components, the vertices u, v and a_1 can occur in one of the following configurations in G:



The first case implies the existence of a directed cycle in D'; in the second case, $a_1 \in N \subset C$, in the third case, $a_1 \in \mathrm{ne}_G(u)$.
Therefore $k \geq 3$. In complete analogy to the proof of Proposition 25, we can show that γ is also a v-u-path in D, hence D' has a directed cycle, a contradiction.

"$\Leftarrow$": Let $D \in \mathbf{D}(G)$ be a DAG with $\{a \in \mathrm{ne}_G(v) \mid a \longrightarrow v \in D\} = C$ and in which all edges of $D[T_G(u)]$ point away from u; such a DAG exists by Corollary 42 and meets the requirements of Proposition 34. It remains to show that D' is acyclic, that means that D has no v-u-path except (v,u).

Suppose, for the sake of contradiction, that D has such a path $\gamma = (a_0 \equiv v, a_1, \ldots, a_k \equiv u)$. γ is then also a v-u-path in G, hence there is, by assumption, some $a_i \in C \cup P$. If $a_i \in C$, $(a_0, a_1, \ldots, a_i, a_0)$ would be a cycle in D; on the other hand, if $a_i \in P$, $(a_i, a_{i+1}, \ldots, a_k, a_i)$ would be a cycle in D, a contradiction.

Uniqueness of $\mathcal{E}_{\mathcal{I}}(D')$: The proof given for Proposition 31 is also valid here. ■

Proof of Corollary 36 The fact that $\mathrm{pa}_D(v) = \mathrm{pa}_G(v) \cup C$ is clear from Proposition 34; it remains to show that $\mathrm{pa}_D(u) = \mathrm{pa}_G(u)$. Any neighbor a of u must also be a child of v, otherwise G would have a subgraph of the form $v \longrightarrow u -\!\!-\, a$, which is forbidden by Theorem 18(iii). Hence $a \longleftarrow u \in D$ for all $a \in \mathrm{ne}_G(u)$ since the other orientation would imply a directed cycle in D'. ■

References

S. A. Andersson, D. Madigan, and M. D. Perlman. A characterization of Markov equivalence classes for acyclic digraphs. *Annals of Statistics*, 25(2):505–541, 1997.

L. E. Brown, I. Tsamardinos, and C. F. Aliferis. A comparison of novel and state-of-the-art polynomial bayesian network learning algorithms. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI)*, volume 20, page 739, 2005.

R. Castelo and T. Kočka. On inclusion-driven learning of Bayesian networks. *Journal of Machine Learning Research*, 4:527–574, 2003.

D. M. Chickering. Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research*, 2(3):445–498, 2002a.

D. M. Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3(3):507–554, 2002b.

G. F. Cooper and C. Yoo. Causal discovery from a mixture of experimental and observational data. In *Uncertainty in Artificial Intelligence*, pages 116–125, 1999.

D. G. Corneil. Lexicographic breadth first search—a survey. In *Graph-Theoretic Concepts in Computer Science*, volume 3353 of *Lecture Notes in Computational Science*, pages 1–19. Springer, Berlin, 2004.

D. Eaton and K. Murphy. Exact Bayesian structure learning from uncertain interventions. In *Artificial Intelligence and Statistics*, volume 2, pages 107–114, 2007.

F. Eberhardt. Almost optimal intervention sets for causal discovery. In *Uncertainty in Artificial Intelligence*, pages 161–168, 2008.

F. Eberhardt, C. Glymour, and R. Scheines. On the number of experiments sufficient and in the worst case necessary to identify all causal relations among N variables. In *Uncertainty in Artificial Intelligence*, pages 178–184, 2005.

F. Eberhardt, P. O. Hoyer, and R. Scheines. Combining experiments to discover linear cyclic models with latent variables. In *Artificial Intelligence and Statistics*, pages 185–192, 2010.

A. Hauser and P. Bühlmann. Jointly interventional and observational data: estimation of corresponding Markov equivalence classes of directed acyclic graphs. Work in progress, 2012.

Y. He and Z. Geng. Active Learning of Causal Networks with Intervention Experiments and Optimal Designs. *Journal of Machine Learning Research*, 9:2523–2547, 2008.

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

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

K. B. Korb, L. R. Hope, A. E. Nicholson, and K. Axnick. Varieties of causal intervention. In *Pacific Rim International Conference on Artificial Intelligence*, pages 322–331, New York, 2004. Springer.

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

C. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky. Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the National Academy of Sciences*, 107(14):6286–6291, 2010.

D. Marbach, T. Schaffter, C. Mattiussi, and D. Floreano. Generating realistic in silico gene networks for performance assessment of reverse engineering methods. *Journal of Computational Biology*, 16(2):229–239, 2009.

C. Meek. *Graphical models: selecting causal and statistical models*. PhD thesis, Philosophy Dept. Carnegie Mellon University, Pittsburgh, PA, 1997.

K. Murphy. The Bayes net toolbox for Matlab. *Computing science and statistics*, 33(2):1024–1034, 2001.

J. Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

R. J. Prill, D. Marbach, J. Saez-Rodriguez, P. K. Sorger, L. G. Alexopoulos, X. Xue, N. D. Clarke, G. Altan-Bonnet, and G. Stolovitzky. Towards a rigorous assessment of systems biology models: the DREAM3 challenges. *PLoS ONE*, 5(2):e9202, 02 2010.

R. W. Robinson. Counting labeled acyclic digraphs. In F. Harary, editor, *New Directions in the Theory of Graphs*, pages 239–273, New York, 1973. Academic Press.

D. J. Rose. Triangulated graphs and the elimination process. *Journal of Mathematical Analysis and Applications*, 32(3):597–609, 1970.

D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5(2):266–283, 1976.

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

T. Silander and P. Myllymäki. A simple approach for finding the globally optimal Bayesian network structure. In *Uncertainty in Artificial Intelligence*, San Francisco, 2006.

P. Spirtes, C. N. Glymour, and R. Scheines. *Causation, Prediction, and Search.* MIT Press, 2000.

E. Szpilrajn. Sur l'extension de l'ordre partiel. *Fundamenta Mathematicae*, 16:386–389, 1930.

J. Tian and J. Pearl. Causal discovery from changes. In *Uncertainty in Artificial Intelligence*, pages 512–521, 2001.

S. Tong and D. Koller. Active learning for structure in Bayesian networks. In *International Joint Conference on Artificial Intelligence*, volume 17, pages 863–869, 2001.

I. Tsamardinos, L. E. Brown, and C. F. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, 2006.

T.S. Verma and J. Pearl. Equivalence and synthesis of causal models. In *Uncertainty in Artificial Intelligence*, page 270, 1990.

S. Wright. Correlation and causation. *Journal of Agricultural Research*, 20(7):557–585, 1921.