Causal Structure Learning in High Dimensions

Wenyu Chen

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2022

Reading Committee:
Mathias Drton, Chair
Ali Shojaie, Chair
Emilija Perković

Program Authorized to Offer Degree:
Statistics

University of Washington

Abstract

Causal Structure Learning in High Dimensions

Wenyu Chen

Co-Chairs of the Supervisory Committee:
Professor Mathias Drton
Department of Mathematics, Technische Universität München

Professor Ali Shojaie
Department of Biostatistics

Directed graphical models are commonly used to model causal relations between random variables and to understand conditional independencies in their joint distributions. We focus on the crucial task of structure learning, which aims to recover graphical structures using observational data sampled from distributions that obey certain underlying graphical model. A common challenge in structure learning is the computational and statistical cost of learning large graphs or using high dimensional data. In this dissertation, we study four cases where the efficiency of structure learning could be improved over existing methods. We propose new algorithms and provide theoretical consistency guarantees.

First, we study a simple setting of linear structural equation model (SEM) with equal error variances. It is known that in this setting the DAG can be uniquely identified from observational data (Peters and Bühlmann, 2014). We proposed in Chapter 2 a simple yet state-of-the-art procedure that sequentially estimates the causal ordering of the random variables. This procedure is consistent and readily extendable to high-dimensional setting. We provided theoretical guarantees as well as simulation results to demonstrate the efficiency.

In Chapter 3 we consider the problem of structure learning in sparse high-dimensional settings that may be subject to the presence of unmeasured confounders, as well as selection

bias. Based on the structure found in common families of large random networks and examining the representation of local structures in linear SEM, we propose a new local notion of sparsity for consistent structure learning in the presence of latent and selection variables, and develop a new version of the Fast Causal Inference (FCI) algorithm with reduced computational and sample complexity, which we refer to as local FCI (lFCI). The new notion of sparsity allows the presence of highly connected hub nodes, which are common in real-world networks, but problematic for existing methods. Our numerical experiments indicate that the lFCI algorithm achieves state-of-the-art performance across many classes of large random networks containing hub nodes.

In DAGs, directed paths represent causal pathways between the corresponding variables. The variable at the beginning of such a path is referred to as an ancestor of the variable at the end of the path. In Chapter 4, we investigate the graphical characterization of ancestral relations via CPDAGs and d-separation relations. We propose a framework that can learn definite non-ancestral relations without first learning the skeleton. We demonstrated that this framework yields structural information that can be used in both score- and constraint-based algorithms to learn causal DAGs more efficiently.

In Chapter 5, we consider an intermediate problem in DAG learning, where a partial causal ordering of variables is available. We discuss a general estimation procedure for discovering DAGs with arbitrary structure from partial orderings. We also present efficient estimation algorithms for two popular classes of high-dimensional sparse directed acyclic graphs, namely linear and additive structural equation models.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First of all, I would like to express my deepest appreciation to my PhD advisors, Mathias Drton and Ali Shojaie. Your invaluable guidance has made my research career and this dissertation possible, as well as enjoyable. You deeply influence the way I think about statistics and helped me develop useful skills that benefit me in the long term.

I also could not have undertaken this journey without my advisory committee, who generously provided knowledge and expertise. Many thanks go to Emilija Perković for your feedback and support. Discussing problems with you has always been cheerful and helpful. I also thank Thomas Richardson for your insightful suggestions. You introduced me to the intersection of causal inference and graphical models.

I am also grateful to the people who introduced me to academia. Many thanks to Eugenia Cheng, who showed me mathematics can be so much fun. Thanks should also go to Kathryn Lindsey, who ignited my passion for research. I'm also extremely grateful to my undergraduate advisor Rina Foygel Barber. Your continuous support helped me develop my statistical intuitions, communication skills, and so much more.

I also thank the faculty, department staff, and my fellow PhD students in the Department of Statistics and Biostatistics. I would like to recognize June Morita and Tamre Cardoso for supervising my role as lead TA and inspiring me to spend time on teaching and consulting.

Last but not the least, I would thank my family for nurturing, supporting, and motivating me. My parents always believe in me and stand behind the decisions I have made. Words cannot express my gratitude to my partner Shuyang for all the love, care, and joy you shared with me. Many thanks also to my cats DiscoBall and Myuu for emotional support. My family and friends, thank you all for always being there with me, in-person or online.

DEDICATION

to my family

Chapter 1

INTRODUCTION AND BACKGROUND

1.1 Directed Graphical Models and Structure Learning

Directed graphical models are commonly used to model causal relations between random variables in complex systems (Spirtes et al., 2001; Pearl, 2009; Maathuis et al., 2018). In this framework, each random variable is a function of other variables (its causes) and stochastic noise. Estimating such causal graphs is important in exploratory data analysis, to generate causal hypotheses, and facilitate design of experiments. Concretely, causal relations can be represented by a directed acyclic graph (DAG), with vertices representing random variables, and directed edges representing direct causal effects. The DAG aids, in particular, in understanding conditional independences that the model imposes on the joint distribution of the random variables. These conditional independences provide an alternative characterization of the joint distribution and can be read off the graph using the concept of d-separation. If a joint distribution satisfies all these imposed conditional independences, it is said to be Markov with respect to the DAG. Structure learning from observational data is then the problem of learning a DAG from data sampled independently from a distribution that is Markov with respect to the DAG.

A crucial aspect of structure learning stems from the fact that the data-generating DAG may be non-identifiable: Many different DAGs may yield the same statistical model for the observational data at hand. These DAGs form a Markov equivalence class. Members of a Markov equivalence class share the same adjacencies and unshielded colliders (Andersson et al., 1997). Assuming *faithfulness*, that is, the conditional independence relationships among the variables correspond exactly to d-separations implied by the DAG, the

Markov equivalence class can be uniquely recovered from conditional independence relationships among the corresponding random variables, and it can be represented via a completed partially directed acyclic graph (CPDAG) (Spirtes et al., 2001).

When all relevant variables are observed, a variety of techniques exist for learning the CPDAG from observational data if we assume some form of faithfulness. Existing structure learning methods can be broadly categorized into constraint-based, score-based, and hybrid approaches (for comprehensive discussions, see Colombo et al., 2012; Loh and Bühlmann, 2014; Raskutti and Uhler, 2018). In constraint-based approaches, a DAG is learned by discovering and imposing graphical constraints via tests of conditional independence. Examples of constraint based methods are SGS (Glymour et al., 1987) and PC (Spirtes et al., 2001). In score-based approaches, a score, for example BIC, is assigned to each DAG and then an algorithm searches for the DAG that optimizes the score. Since searching through the space of all possible DAGs is NP-hard (Chickering, 1996), the optimization is usually performed by greedy search, such as in Greedy Equivalence Search (GES) (Chickering, 2002). Hybrid approaches use schemes in which the two approaches inform each other, for example in Max-Min-Hill-Climbing (MMHC) (Tsamardinos et al., 2006) and Sparsest Permutation (Raskutti and Uhler, 2018).

One of the most commonly used constraint-based method is the PC algorithm (Spirtes et al., 2001), which is popularized in Kalisch and Bühlmann (2007) for high-dimensional settings, and is the building block for many other constraint-based and hybrid algorithms (see, e.g., Tsamardinos et al., 2006; Ogarrio et al., 2016). The PC algorithm hierarchically performs tests of conditional independence with conditioning sets of increasing size. Under a *faithfulness* assumption, the population version of PC algorithm outputs the correct CPDAG Spirtes et al. (2001), and the finite sample version is consistent in sparse high-dimensional settings (Kalisch and Bühlmann, 2007). The results has been extended to Gaussian copula models in Harris and Drton (2013) using rank correlations.

1.2 Structural Equation Models and identifiability

A structural equation model for a random vector $X = (X_1, \ldots, X_p)$ postulates causal relations in which each variable X_j is a function of a subset of the other variables and a stochastic error ε_j. In this framework, causal discovery and structure learning is the problem of inferring which of other variables each variable X_j depends on. Throughout this paper, we consider this problem where only observational data, that is, a sample from the joint distribution of X, is available.

Suppose, without loss of generality, that the observed random vector $X = (X_1, \ldots, X_p)$ is centered. In a structural equation model, X then solves an equation system

$$X_j = f_j(X_{\mathrm{pa}_j}, \varepsilon_j), \qquad j = 1, \ldots, p, \tag{1.1}$$

where pa_j are the parents of X_j, ε_j are independent random variables with mean zero, and the coefficients f_j are unknown functionals. A special family is the linear SEMs, which can be written as

$$X_j = \sum_{k \in \mathrm{pa}_j} \beta_{kj} X_k + \varepsilon_j, \qquad j = 1, \ldots, p, \tag{1.2}$$

where coefficients β_{kj} are unknown parameters.

The SEMs represent a variety of causal mechanisms and are well-studied. In particular, a SEM is identifiable if each variable is determined by some linear function of its parents and an independent error which belongs to some non-Gaussian distribution (Shimizu et al., 2006; Zhang and Hyvärinen, 2009b; Loh and Bühlmann, 2014; Zhang and Hyvärinen, 2009a; Wang and Drton, 2020). Relaxing the linearity assumption, it is also known that a SEM is identifiable if each variable is determined by some non-linear function of its parents and an independent error (Zhang and Hyvärinen, 2009a; Hoyer et al., 2008; Mooij et al., 2009; Peters et al., 2011), and consequently the setting of nonlinear functions with Gaussian errors is identifiable. The popular and simple setting of linear SEM with Gaussian noise, unfortunately, is non-identifiable in general. A special case when all error variances are equal

(or monotonicaly sorted along the causal ordering), is shown to be identifiable (Peters and Bühlmann, 2014; Chen et al., 2019; Ghoshal and Honorio, 2018; Park, 2020).

1.3 Notations

We will invoke the following graphical concepts. If the considered graph G contains the edge $k \to j$, then k is a parent of its child j. We write $\text{pa}(j)$ for the set of all parents of a node j. Similarly, $\text{ch}(j)$ is the set of children of j. If there exists a directed path $k \to \ldots \to j$, then k is an ancestor of its descendant j. The sets of ancestors and descendants of j are $\text{an}(j)$ and $\text{de}(j)$, respectively. Here, $j \in \text{an}(j)$ and $j \in \text{de}(j)$. A set of nodes C is ancestral if $\text{an}(j) \subseteq C$ for all $j \in C$. Similarly, C contains all its descendants if $\text{de}(j) \subseteq C$ for all $j \in C$.

Chapter 2

ON CAUSAL DISCOVERY WITH EQUAL VARIANCE ASSUMPTION

2.1 Introduction

In this work we consider the problem of learning the data-generating DAG in settings where all relevant variables are observed. While in general only an equivalence class of structures can then be inferred (Spirtes et al., 2001; Pearl, 2009), recent work stresses that unique identification is possible under assumptions such as non-linearity with additive errors, linearity with non-Gaussian errors, and linearity with errors of equal variance; see the reviews of Drton and Maathuis (2017) and Heinze-Deml et al. (2018) or the book of Peters et al. (2017).

This work is concerned with the equal variance case treated by Peters and Bühlmann (2014) and Loh and Bühlmann (2014) who prove identifiability of the causal structure and propose greedy search methods for its estimation. Our key observation is that the identifiability is implied by an ordering among certain conditional variances. Ordering estimates of these variances yields a fast method for estimation of the causal ordering of the variables. The precise causal structure can then be inferred using variable selection techniques for regression (Shojaie and Michailidis, 2010). Specifically, we develop a top-down approach that infers the ordering by successively identifying sources. The method is developed for low- as well as high-dimensional problems. Simulations show significant gains in computational efficiency when compared with greedy search and increased accuracy when the number of variables p is large.

In this chapter we also include a bottom-up method which identified the causal ordering by successively finding sinks via minimal precisions. We are aware that the same bottom-up

approach is concurrently studied in Ghoshal and Honorio (2018). We included the bottom-up approach in this document for completeness of presentation. We emphasize that our top-down approach only requires control of the maximum in-degree as opposed to the bottom-up approach which requires control of the maximum Markov blanket. This is discussed further in Section 2.3.2 and a direct numerical comparison is given in Section 2.4.2.

2.2 Identifiability by Ordering Variances

Suppose, without loss of generality, that the observed random vector $X = (X_1, \ldots, X_p)$ is centered. In a linear structural equation model, X then solves an equation system displayed in (1.2)

$$X_j = \sum_{k \neq j} \beta_{kj} X_k + \varepsilon_j, \qquad j = 1, \ldots, p, \tag{2.1}$$

where the ε_j are independent random variables with mean zero, and the coefficients β_{kj} are unknown parameters. Following Peters and Bühlmann (2014), we assume that all ε_j have a common unknown variance $\sigma^2 > 0$. We will write $X \sim (B, \sigma^2)$ to express the assumption that there indeed exist independent errors $\varepsilon_1, \ldots, \varepsilon_p$ of equal variance σ^2 such that X solves (1.2) for coefficients given by a real $p \times p$ matrix $B = (\beta_{jk})$ with zeros along the diagonal.

The causal structure inherent to the equations in (1.2) is encoded in a directed graph $G(B)$ with vertex set $V = \{1, \ldots, p\}$ and edge set $E(B)$ equal to the support of B. So, $E(B) = \{(k, j) : \beta_{kj} \neq 0\}$. Inference of $G(B)$ is the goal of causal discovery as considered in this paper. As in related work, we assume $G(B)$ to be a directed acyclic graph (DAG) so that B is permutation similar to a triangular matrix. Then (1.2) admits the unique solution $X = (I - B^\top)^{-1}\varepsilon$ where $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_p)$. Hence, the covariance matrix of $X \sim (B, \sigma^2)$ is

$$\Sigma := \mathbb{E}\left[XX^T\right] = \sigma^2 (I - B^\top)^{-1}(I - B^\top)^{-T}. \tag{2.2}$$

The main result of Peters and Bühlmann (2014) shows that the graph $G(B)$ and the parameters B and σ^2 are identifiable from the covariance in (2.2). No faithfulness assumptions are needed.

Theorem 1. *Let* $X \sim (B_X, \sigma_X^2)$ *and* $Y \sim (B_Y, \sigma_Y^2)$ *with both* $G(B_X)$ *and* $G(B_Y)$ *directed and acyclic. If* $\mathrm{Var}(X) = \mathrm{Var}(Y)$*, then* $G(B_X) = G(B_Y)$*,* $B_X = B_Y$*, and* $\sigma_X^2 = \sigma_Y^2$*.*

Let G be a DAG, then it admits a topological ordering of its vertices. In other words, there exists a numbering σ such that $\sigma(j) < \sigma(k)$ only if $k \notin \mathrm{an}(j)$. Every DAG contains at least one source, that is, a node j with $\mathrm{pa}(j) = \emptyset$. Similarly, every DAG contains at least one sink, which is a node j with $\mathrm{ch}(j) = \emptyset$. In this section we first give an inductive proof of Theorem 1 that proceeds by recursively identifying source nodes for $G(B)$ and subgraphs. We then clarify that alternatively one could identify sink nodes. Our first lemma clarifies that the sources in $G(B)$ are characterized by minimal variances. We define

$$\zeta \equiv \zeta(B) \;=\; \min_{(k,j)\in E(B)} \beta_{kj}^2. \tag{2.3}$$

Lemma 1. *Let* $X \sim (B, \sigma^2)$ *with* $G(B)$ *directed and acyclic. If* $\mathrm{pa}(j) = \emptyset$*, then* $\mathrm{Var}\left(X_j\right) = \sigma^2$*. If* $\mathrm{pa}(j) \neq \emptyset$*, then* $\mathrm{Var}\left(X_j\right) \geq \sigma^2(1+\zeta) > \sigma^2$*.*

Proof. For any directed path $\ell = (j_1, \ldots, j_z)$ in $G(B)$, define the path weight $w(\ell) = \prod_{i=1}^{z} \beta_{j_i, j_{i+1}}$. Let $\mathcal{L}_{k,j}$ be the set of all directed paths from k to j. The *total effect* of k on j is $\pi_{kj} = \sum_{\ell \in \mathcal{L}_{k,j}} w(\ell)$. Let $\Pi = (I - B^\top)^{-1}$. Then it holds that $\pi_{kj} = [\Pi]_{k,j}$. Note that $\pi_{jj} = 1$. From (2.2), $\mathrm{Var}\left(X_j\right) = \sigma^2 \sum_{k=1}^{p} \pi_{kj}^2$. Hence, if $\mathrm{pa}(j) = \emptyset$, then $\mathrm{Var}\left(X_j\right) = \sigma^2$ because $\pi_{kj}^2 = 0$ for all $k \neq j$. If $\mathrm{pa}(j) \neq \emptyset$ then by acyclicity of $G(B)$ there exists a node $\ell \in \mathrm{pa}(j)$ such that $\mathrm{de}(\ell) \cap \mathrm{pa}(j) = \{\ell\}$. Then $\pi_{\ell j}^2 = \beta_{\ell j}^2 \geq \zeta$ and

$$\mathrm{Var}\left(X_j\right) = \sigma^2\left(1 + \sum_{k \neq j} \pi_{kj}^2\right) \geq \sigma^2\left(1 + \pi_{\ell j}^2\right) \geq \sigma^2\left(1 + \zeta\right).$$

□

The next lemma shows that by conditioning on a source, or more generally an ancestral set, one recovers a structural equation model with equal error variance whose graph has the source node or the entire ancestral set removed. For a variable X_j and a vector $X_C = (X_k : k \in C)$, we define $X_{j.C} = X_j - \mathbb{E}\left[X_j \;\middle|\; X_C\right]$.

Lemma 2. *Let $X \sim (B, \sigma^2)$ with $G(B)$ directed and acyclic. Let C be an ancestral set in $G(B)$. Then $(X_{j.C} : j \notin C) \sim (B[-C], \sigma^2)$ for submatrix $B[-C] = (\beta_{kj})_{j,k\notin C}$.*

Proof. Let $j \notin C$. Since C is ancestral, X_C is a function of ε_C only and thus independent of ε_j. Hence, $\mathbb{E}\left[\varepsilon_j \,\middle|\, X_C\right] = \mathbb{E}\left[\varepsilon_j\right] = 0$. Because it also holds that $X_{k.C} = 0$ for $k \in C$, we have from (1.2) that

$$X_{j.C} = \sum_{k\in \mathrm{pa}(j)\setminus C} \beta_{kj} X_{k.C} + \varepsilon_j.$$

□

The lemmas can be combined to identify a topological ordering of $G(B)$ and prove Theorem 1.

Proof of Theorem 1. The claim is trivial for $p = 1$ variables, which gives the base for an induction on p. If $p > 1$, then Lemma 1 identifies a source c by variance minimization. Conditioning on c as in Lemma 2 reduces the problem to size $p-1$. By the induction assumption, σ^2 and $B[-\{c\}]$ can be identified. The regression coefficients in the conditional expectations $\mathbb{E}\left[X_j \,\middle|\, X_c\right]$ for $j \neq c$ identify the missing first row and column of B; see e.g. Drton (2018, §7). □

Next, we show that alternatively one may minimize precisions to identify a sink node. We state analogues of Lemma 1 and 2 which can also be used to prove Theorem 1.

Lemma 3. *Let $X \sim (B, \sigma^2)$ with $G(B)$ directed and acyclic. Let Σ be the covariance matrix of X, and $\Phi = \Sigma^{-1}$ the precision matrix. If $\mathrm{ch}(j) = \emptyset$, then $\Phi_{jj} = 1/\sigma^2$. If $\mathrm{ch}(j) \neq \emptyset$, then $\Phi_{jj} \geq \{1 + \zeta|\mathrm{ch}(j)|\}/\sigma^2 > 1/\sigma^2$.*

Proof. The diagonal entries of $\Phi = \frac{1}{\sigma^2}(I - B^\top)(I - B^\top)^T$ are $\Phi_{jj} = \frac{1}{\sigma^2}(1 + \sum_{k\in\mathrm{ch}(j)} \beta_{jk}^2)$. So $\Phi_{jj} = 1/\sigma^2$ if $\mathrm{ch}(j) = \emptyset$, and $\Phi_{jj} \geq \{1 + |\mathrm{ch}(j)|\zeta\}/\sigma^2$ if $\mathrm{ch}(j) \neq \emptyset$. □

Marginalization of a sink is justified by the following well-known fact (e.g. Drton and Maathuis, 2017, §5).

Lemma 4. *Let $X \sim (B, \sigma^2)$ with $G(B)$ directed and acyclic. Let C be an ancestral set in $G(B)$. Then $X_C \sim (B[C], \sigma^2)$ for submatrix $B[C] = (\beta_{kj})_{j,k \in C}$.*

2.3 Estimation Algorithms

2.3.1 Low-dimensional Problems

The results from Section 2.2 naturally yield an iterative top-down algorithm for estimation of a topological ordering for $G(B)$. In each step of the procedure we select a source node by comparing variances conditional on the previously selected variables, so the criterion in the minimization in Algorithm 1 is the variance

$$f_1(\hat{\Sigma}, \Theta, j) \;=\; \hat{\Sigma}_{j,j} - \hat{\Sigma}_{j,\Theta}\hat{\Sigma}^{-1}_{\Theta,\Theta}\hat{\Sigma}_{\Theta,j} \;=\; \frac{1}{\{(\hat{\Sigma}_{\Theta\cup\{j\},\Theta\cup\{j\}})^{-1}\}_{j,j}}, \tag{2.4}$$

where $\hat{\Sigma}$ is the sample covariance matrix. Alternatively, and as also observed by Ghoshal and Honorio (2018), a bottom-up procedure could construct the reverse causal ordering by successively minimizing precisions (or in other words, full conditional variances).

Algorithm 1: Topological Ordering: General procedure with criterion f

Input : $\hat{\Sigma} \in \mathbb{R}^{p \times p}$ (estimated) covariance of X

Output: Θ

$\Theta^{(0)} \leftarrow \emptyset$;

for $z = 1, \ldots, p$ **do**

  $\theta \leftarrow \arg\min_{j \in V \setminus \Theta^{(z-1)}} f(\hat{\Sigma}, \Theta^{(z-1)}, j)$;

  Append θ to $\Theta^{(z-1)}$ to form $\Theta^{(z)}$

return the ordered set $\Theta^{(p)}$.

To facilitate theoretical statements about our top-down procedure, we assume that the errors ε_j in (1.2) are all sub-Gaussian with maximal sub-Gaussian parameter $\gamma > 0$. We indicate this by writing $X \sim (B, \sigma^2, \gamma)$. Our analysis is restricted to inference of a topological

ordering. Shojaie and Michailidis (2010) give results on lasso-based inference of the graph given an ordering.

Theorem 2. *Let* $X \sim (B, \sigma^2, \gamma)$ *with* $G(B)$ *directed and acyclic. Suppose the covariance matrix* $\Sigma = \mathbb{E}\left[XX^T\right]$ *has minimum eigenvalue* $\lambda_{\min} > 0$*. If*

$$n \;>\; p^2 \left\{\log(p^2+p) - \log(\epsilon/2)\right\} 128 \left(1 + 4\frac{\gamma^2}{\sigma^2}\right)^2 \left(\max_{j \in V} \Sigma_{j,j}\right)^2 \left(\frac{\zeta\lambda_{\min} + 2\sigma^2}{\zeta\lambda_{\min}^2}\right)^2,$$

then Algorithm 1 using criterion criterion (2.4) *recovers a topological ordering of* $G(B)$ *with probability at least* $1 - \epsilon$*.*

The result follows using concentration for sample covariances (Ravikumar et al., 2011, Lemma 1) and error propagation analysis as in Harris and Drton (2013, Lemma 5). We give details in Appendix A.1, which is found in the supplementary materials.

2.3.2 High-dimensional Problems

The consistency result in Theorem 2 requires the sample size n to exceed a multiple of $p^2 \log(p)$ and only applies to low-dimensional problems. If $p > n$, method will stop at the nth step when the conditional variance in (2.4) becomes zero for all $j \notin \Theta$.

However, in the high-dimensional setting if $G(B)$ has maximum in-degree bounded by a small integer q, we may modify the criterion from (2.4) to

$$f_2(\hat{\Sigma}, \Theta, j) \;=\; \min_{C \subseteq \Theta, |C| = q} f_1(\hat{\Sigma}, C, j) \;=\; \min_{C \subseteq \Theta, |C| = q} \hat{\Sigma}_{j,j} - \hat{\Sigma}_{j,C}(\hat{\Sigma}_{C,C})^{-1}\hat{\Sigma}_{C,j}. \tag{2.5}$$

The intuition is that in the population case, adjusting by a smaller set $C \subseteq \Theta^{(z)}$ with $\mathrm{pa}(j) \subseteq C$ yields the same results as adjusting by all of $\Theta^{(z)}$. The next lemma makes the idea rigorous.

Lemma 5. *Let* $X \sim (B, \sigma^2)$ *with* $G(B)$ *directed and acyclic with maximum in-degree at most* q*. Let* $\Sigma = \mathbb{E}\left[XX^T\right]$*, and suppose* $S \subseteq V \setminus \{j\}$ *is an ancestral set. If* $\mathrm{pa}(j) \subseteq S$*, then* $f_2(\Sigma, S, j) = \sigma^2$*. If* $\mathrm{pa}(j) \nsubseteq S$*, then* $f_2(\Sigma, S, j) \geq \sigma^2(1 + \zeta)$*.*

Proof. The conditional variance of X_j given X_S is the variance of the residual $X_{j.S}$. By Lemma 2, $X_{j.S}$ has the same distribution as X'_j when $X' \sim (B[-S], \sigma^2)$. Now, j is a source of $G(B[-S])$ if and only if $\text{pa}(j) \subseteq S$. Lemma 1 implies that $\text{Var}\left(X_j|X_C\right) = \sigma^2$ if $\text{pa}(j) \subseteq S$ and $\text{Var}\left(X_j|X_C\right) \geq \sigma^2(1+\zeta)$ otherwise. The claim about $f_2(\Sigma, S, j)$ now follows. □

Based on Lemma 5, we have the following result whose proof is analogous to that of Theorem 2. The key feature of the result is a drop from p^2 to $(q+1)^2$ in the sample size requirement.

Theorem 3. *Let $X \sim (B, \sigma^2, \gamma)$ with $G(B)$ directed and acyclic with of maximum in-degree at most q. Suppose all $(q+1)\times(q+1)$ principal submatrices of $\Sigma = \mathbb{E}\left[XX^T\right]$ have minimum eigenvalue at least $\lambda_{\min} > 0$. If*

$$n \;>\; (q+1)^2 \left\{\log(p^2+p) - \log(\epsilon/2)\right\} 128 \left(1 + 4\frac{\gamma^2}{\sigma^2}\right)^2 \left(\max_{j\in V} \Sigma_{j,j}\right)^2 \left(\frac{\zeta\lambda_{\min} + 2\sigma^2}{\zeta\lambda_{\min}^2}\right)^2,$$

then Algorithm 1 using criterion (2.5) *recovers a topological ordering of $G(B)$ with probability at least $1-\epsilon$.*

We contrast our guarantees with those for the bottom-up method of Ghoshal and Honorio (2018) which selects sinks by minimizing conditional precisions that are estimated using the CLIME estimator (Cai et al., 2011). Because CLIME requires small Markov blankets, the bottom-up procedure has sample complexity $O\left(d^8 \log(p)\right)$ where d is the maximum total degree. This implies that the procedure cannot consistently discover graphs with hubs, i.e., nodes with very large out-degree, in the high dimensional setting. This said, the computational complexity of the bottom-up procedure is polynomial in d, while our top-down procedure is exponential in the maximum in-degree. In practice, we use a branch-and-bound procedure (Miller, 2020) to efficiently select the set which minimizes the conditional variance; see Section 2.4.2.

Bottom-up Approach: It comes to our attention that a method based on precision matrix estimation has been proposed in Ghoshal and Honorio (2018). The method uses constrained

ℓ_1 inverse matrix estimation (CLIME) method by Cai et al. (2011). Performance of this method is evaluated in the simulation studies alongside our other proposed methods. For completeness, we describe the bottom-up approach based on conditional variance estimation.

At each step of the bottom-up algorithm, we require estimates of all conditional precisions, i.e., the inverses of all conditional variances. The bottom-up variant estimates a *reversed* topological ordering by minimization of precisions, so the criterion is

$$f_2(\hat{\Sigma}, \Theta, j) \;=\; \{(\hat{\Sigma}_{V\setminus\Theta, V\setminus\Theta})^{-1}\}_{j,j} \;=\; \frac{1}{f_1(\hat{\Sigma}, V \setminus \Theta, j)}. \tag{2.6}$$

These precisions can be obtained by either estimating the entire conditional precision matrix, or by directly estimating all the conditional variances. In low-dimensional settings, the validity of this approach follows the same argument for Theorem 2. See Appendix A.1. It is worth noting that in high-dimensional setting, consistent estimation of both targets requires the considered graph to have small Markov blanket, as opposed to small maximum in-degree as for the top-down approach. Despite the stricter assumption needed, we present the bottom-up methods in this note as they could be potentially cheaper to compute.

A variety of methods are available to estimate high-dimensional precision matrices (Drton and Maathuis, 2017). As noted above, Ghoshal and Honorio (2018) proposed a iterative estimation method based on a simple update rule of CLIME Cai et al. (2011). Here we note that we only need the diagonal entries of the precision matrices, i.e., the inverses of the full conditional variances $\mathrm{Var}\left(X_j \;\middle|\; X_{V\setminus\{j\}}\right)$, we may also simply use a lasso-approach for variance estimation in a high-dimensional linear model. Following Yu and Bien (2019), we estimate the conditional variance with the organic lasso estimator, which is the minimal value of the ℓ_1^2 penalized problem

$$\hat{\sigma}^2_\lambda = \min_{\beta\in\mathbb{R}^p}\left(\frac{1}{n}\|y - X\beta\|_2^2 + 2\lambda\|\beta\|_1^2\right). \tag{2.7}$$

with $\lambda = \left\{2Mn^{-1}\log(p)\right\}^{1/2}$ for some $M > 1$.

We modify the criterion from (2.6) to

$$f_3(\hat{\Sigma}, \Theta, j) \;=\; \widehat{\mathrm{Var}}(X_j | X_{V\setminus(\Theta\cup\{j\})}) \;:=\; \hat{\sigma}^2_{j,\Theta,\lambda}, \tag{2.8}$$

which is the solution of (2.7) when we regress X_j on the standardized covariates $X_{V\setminus(\Theta\cup\{j\})}$. We derive the Theorem 4 using Theorem 9 in Yu and Bien (2019). Although the statement of Theorem 4 does not explicitly enforce sparsity in the graph, in general, the value of A will depend on the maximum degree of the graph.

Theorem 4. *Let* $X \sim (B, \sigma^2, \gamma)$ *with* $G(B)$ *directed acyclic and covariance matrix* $\Sigma = \mathbb{E}\left[XX^T\right]$. *Let*

$$A = \max_{j\in[p]}\left(\frac{\| \left(\Sigma_{V\setminus\{j\},V\setminus\{j\}}\right)^{-1} \|_1^2 \|\Sigma_{V\setminus\{j\},j}\|_1^2}{\mathrm{Var}\left(X_j \mid X_{V\setminus\{j\}}\right)}, \frac{\| \left(\Sigma_{V\setminus\{j\},V\setminus\{j\}}\right)^{-1} \|_1 \|\Sigma_{V\setminus\{j\},j}\|_1}{\sqrt{\mathrm{Var}\left(X_j \mid X_{V\setminus\{j\}}\right)}} + \frac{1}{4}\right),$$

and fix some $M > 1$. *If*

$$n > \frac{1}{\zeta^2\epsilon}\left\{A\left(8M + \frac{p^{1-8M}}{\log p}\right)(\log p)^{1/2} + \sqrt{2}\right\}^2,$$

then the bottom-up algorithm that uses criterion (2.8) *with the organic lasso variance estimator for* $\lambda = (2Mn^{-1}\log p)^{1/2}$ *recovers a topological ordering of* $G(B)$ *with probability at least* $1-\epsilon$.

2.4 Numerical Experiments

2.4.1 Low-dimensional Setting

We first assess performance in the low-dimensional setting. Random DAGs with p nodes and a unique topological ordering are generated by: (1) always including edge $v \to v+1$ for $v < p$, and (2) including edge $v \to u$ with probability p_c for all $v < u-1$. We consider a sparse setting with $p_c = 3/(2p-2)$ and a dense setting with $p_c = 0.3$. All linear coefficients are drawn uniformly from $\pm[.3, 1]$. The error terms are standard normal. Performance is measured using Kendall's τ between rankings of variables according to the true and estimated topological orderings. Although the true graph admits a unique ordering by construction, the graph estimated by the greedy search may not admit a unique ordering. Nevertheless, the ranking of variables according to the estimated graph is unique if we allow ties, and

Kendall's τ remains a good measure for all the methods. We also compute the percentage of true edges discovered (Recall), the percentage of estimated edges that are flipped in the true graph (Flipped), and the proportion of estimated edges which are either flipped or not present in the true graph (false discovery rate; FDR).

Tables 2.1 and 2.2 show averages over 500 random realizations for our top-down procedure (TD), the bottom-up procedure (BU) of Ghoshal and Honorio (2018), greedy DAG search (GDS), PC algorithm, and NOTEARS of Zheng et al. (2018). For the bottom up procedure in the low-dimensional setting, we may in fact simply invert the sample covariance to estimate precisions. In TD and BU, edges are filled along the estimated topological ordering in using z-test of partial correlations with Šidák's correction (Drton and Perlman, 2007). For PC, since the output is a Markov equivalent class, we only report the performance with one arbitrary DAG from the estimated equivalent class (See pdag2dag in Kalisch et al. (2022)). For GDS, there are two different implementations. In the pcalg package (Kalisch et al., 2022), the procedure is deterministic and prone to be stuck in local optima of the score function and hence yield worse result than GES; in the GDS code provided in Peters and Bühlmann (2014), the procedure start from a random graph, and multiple random restarts are allowed to aid avoiding local optima. However, we note that both GDS implementations may return adjacency matrices that do not represent any valid DAG, and sometime not even any PDAG [1]. To enable comparison across different methods, we remove the least amount of arrows (in the case of bidirected edge) or edges (in the case of cycles) to make the GDS output into a DAG before computing the performance metrics. For NOTEARS, we use the Python implementation for Gaussian setting from the authors. We report the result of one optimizer (NT0) without sparsity regularization and one optimizer (NT) with default sparsity regularization $\lambda = 0.1$.

In both dense and sparse settings, when $p = 5$, greedy search performs best in all metrics. The space of DAGs with 5 nodes is small enough for greedy search to cover. However, for

[1] In the printed version of this work (Chen et al., 2019) the GDS output was mistakenly assumed to be DAGs, and the performance of GDS is overestimated.

$p = 20$ and 40, the top-down approach does best, with highest recovery rate of topological ordering, and consequently highest power and lowest false discovery rate. With small sample size ($n = 100$), the NOTEARS methods could do better then the bottom-up method; but with larger sample sizes, the performance of bottom-up method catches up with top-down, and they both substantially outperforms NOTEARS and greedy search. We note that when p is small, the performance of NOTEARS is close to the greedy search since the two approaches are optimizing a similar loss function (with ℓ_0 and ℓ_1 penalty); with larger p, the NOTEARS methods show advantage from its more efficient optimization. The PC method performs poorly as it cannot identify the DAG. The GDS method (with sufficient computing power) will likely to perform well in identifying small graphs, and the performance will be suboptimal for larger graphs as the greedy search often stuck in local optimum. The NOTEAR methods exhibits good performance but still subpar compared to the proposed TD and BU methods. The top-down and bottom-up method both have a substantially higher average Kendall's τ than NOTEAR and greedy search.

In our experiments, the proposed methods are roughly 1000 times faster than NOTEAR and 15000 times faster than GES as graph size and density increases. On our personal computer, the average run time in the dense setting with $p = 40$ and $n = 1000$ is 0.3 seconds for the top-down and bottom-up methods, but 300 seconds for NOTEAR and 4,500 seconds for the greedy search with multiple restarting.

2.4.2 High-dimensional Setting

We now test the proposed procedures in a high-dimensional setting with $p > n$ in two scenarios. Random DAGs with p nodes and a unique topological ordering are generated by: (1) always including edge $v \to v + 1$ for $v < p$, and either (2a) for each $v > 2$, including $u_1, u_2 \to v$, where $u_i < v$, and u_i has out-degree $d_{\text{out}}(u_i) < 4$, or (2b) for each $v > 2$, including $u_1, u_2 \to v$, where $u_i < \min(v, 10)$. In both scenarios, the maximum in-degree is fixed to be $q = 3$. In the first scenario, it is also guaranteed that the maximum Markov blanket size is small, bounded by $k \leq 15$. In the second scenario when there exists hubs in

the graph, the maximum Markov blanket size grows with p, with $k \geq 0.2p$. The errors are standard normal.

Algorithm 1 with (2.5) as HTD (high-dimensional top-down) and to the bottom-up method of Ghoshal and Honorio (2018) as HBU. The best subset search step in HTD is carried with subset size $q = 3$; increasing q beyond the true maximum in-degree does not change performance substantially. The HBU is tuned with $\lambda_n = 0.5\sqrt{\log(p)/n}$. Results for greedy search are not shown as computation becomes intractable when $p > 100$. Performance is measured by Kendall's τ to provide direct comparison.

Table 2.3 demonstrates that in the first scenario, both methods perform reasonably well when the considered graph has small Markov blanket. The HTD procedure performs the best in low-dimensional and moderately high-dimensional settings, and both methods have similar performance in very high-dimensional settings. However, when there exists nodes with very large Markov blanket, the top-down method substantially outperforms the bottom-up method.

On our personal computer, the average run time for problems of size $p = 200$ is 10 minutes for the HTD method with $q = 3$. The computational complexity of HBU is determined by the choice of tunning parameter in the precisions estimation step.

Additional simulation settings are presented in Appendix A.2-A.5 in the supplement including a setting with Rademacher errors as considered by Ghoshal and Honorio (2018).

2.5 Discussion

In this note, we proposed a simple method for causal discovery under a linear structural equation model with equal error variances. The procedure consistently estimates a topological ordering of the underlying graph and easily extends to the high-dimensional setting where $p > n$. Simulations demonstrate that the procedure is an attractive alternative to previously considered greedy search methods in terms of both accuracy and computational effort. The advantages of the proposed procedures become especially salient as the number of considered variables increases.

In comparison to the related work of Ghoshal and Honorio (2018), our approach is computationally more demanding for graphs with higher in-degree but requires only control over the maximum in-degree of the graph as opposed to the maximum degree. We also note that as shown in simulations in Appendix A.5 a hybrid method in which greedy search is initialized at estimates obtained from our variance ordering procedures can yield further improvements in performance.

Finally, we note that all discussed methods extend to structural equation models where the error variances are unequal, but known up to ratio. Indeed, if $\mathrm{Var}\left(\varepsilon_j\right) = a_j^2\sigma^2$ for some unknown σ^2 but known $a_1, \ldots, a_p$, we may consider $\tilde{X}_j = X_j/a_j$ instead of the original variables.

Table 2.1: Low-dimensional dense settings

		$p = 5$			$p = 20$			$p = 40$		
	n	100	500	1000	100	500	1000	100	500	1000
Kendall's τ	TD	0.85	0.97	0.99	**0.93**	**0.99**	**>0.99**	**0.96**	**0.99**	>0.99
	BU	0.80	0.96	0.97	0.86	0.97	0.99	0.92	0.98	0.99
	NT0	0.80	0.89	0.91	0.82	0.84	0.85	0.87	0.88	0.88
	NT	0.79	0.86	0.86	0.79	0.82	0.82	0.84	0.86	0.86
	PC	0.18	0.15	0.14	0.18	0.14	0.14	0.17	0.13	0.15
	GDS	**0.88**	**0.98**	**0.99**	0.61	0.75	0.82	0.53	0.59	0.64
Recall %	TD	85	98	99	60	**98**	**>99**	35	**92**	99
	BU	84	98	98	58	97	99	34	92	98
	NT0	84	90	92	**75**	81	82	**72**	78	79
	NT	72	81	79	60	65	67	51	56	58
	PC	46	52	53	17	20	20	7	8	9
	GDS	**91**	**99**	**99**	62	81	88	44	63	71
Flipped %	TD	6	2	1	**3**	**1**	**<0.5**	**1**	**<0.5**	**<0.5**
	BU	6	2	2	3	1	<0.5	1	1	<0.5
	NT0	8	5	5	8	7	7	6	6	5
	NT	8	5	3	6	6	6	4	4	4
	PC	42	43	44	17	20	20	7	8	8
	GDS	**6**	**1**	**1**	13	11	8	11	14	14
FDR %	TD	10	2	2	**5**	**2**	**1**	**4**	**1**	**<0.5**
	BU	10	3	3	7	3	1	6	2	1
	NT0	12	6	5	30	19	18	44	26	25
	NT	11	6	4	19	14	13	26	21	4
	PC	55	54	56	60	60	61	65	66	8
	GDS	**9**	**2**	**1**	43	35	28	58	57	57

Table 2.2: Low-dimensional sparse settings

		$p = 5$			$p = 20$			$p = 40$		
	n	100	500	1000	100	500	1000	100	500	1000
Kendall's τ	TD	0.86	**0.97**	**0.99**	**0.78**	**0.97**	**0.99**	**0.71**	**0.94**	**0.98**
	BU	0.78	0.94	0.98	0.56	0.87	0.94	0.47	0.79	0.91
	NT0	0.86	0.90	0.91	0.63	0.71	0.75	0.53	0.58	0.60
	NT	0.80	0.83	0.87	0.66	0.78	0.79	0.62	0.75	0.76
	PC	0.15	0.11	0.20	0.19	0.17	0.19	0.15	0.15	0.15
	GDS	**0.88**	0.98	0.99	0.60	0.77	0.81	0.47	0.58	0.61
Recall %	TD	87	98	**99**	70	**98**	**>99**	57	**97**	**99**
	BU	85	97	99	64	95	98	49	93	98
	NT0	86	90	92	74	82	85	67	76	78
	NT	74	76	77	73	81	82	**73**	83	84
	PC	45	48	53	42	44	45	40	45	44
	GDS	**90**	**99**	99	**77**	89	90	72	81	82
Flipped %	TD	**5**	2	**1**	**5**	**2**	**<0.5**	**5**	**2**	**1**
	BU	6	3	1	7	4	2	7	6	2
	NT0	7	6	4	15	11	10	20	16	14
	NT	6	6	3	9	6	5	9	5	5
	PC	41	46	41	38	41	41	39	43	44
	GDS	6	**1**	1	15	10	9	20	18	17
FDR %	TD	**7**	3	**1**	**10**	**4**	**1**	**11**	**5**	**2**
	BU	8	4	1	15	8	4	17	11	6
	NT0	11	7	5	39	18	14	59	27	24
	NT	9	7	5	14	8	7	14	7	7
	PC	54	57	52	57	57	57	61	59	59
	GDS	9	**2**	1	39	26	23	54	47	48

Table 2.3: High-dimensional setting with maximum in-degree $q = 3$

		Small k		Hub graph	
n	p	HTD	HBU	HTD	HBU
80	$0.5n$	0.99	0.89	1.00	0.70
	$0.75n$	0.98	0.89	0.99	0.52
	n	0.95	0.87	0.95	0.39
	$1.5n$	0.84	0.83	0.77	0.25
	$2n$	0.72	0.73	0.55	0.16
100	$0.5n$	1.00	0.93	1.00	0.70
	$0.75n$	0.99	0.92	1.00	0.50
	n	0.97	0.87	0.97	0.38
	$1.5n$	0.86	0.84	0.74	0.26
	$2n$	0.73	0.78	0.63	0.12
200	$0.5n$	1.00	0.95	1.00	0.77
	$0.75n$	1.00	0.90	1.00	0.61
	n	0.99	0.79	0.99	0.48
	$1.5n$	0.87	0.74	0.80	0.20
	$2n$	0.74	0.64	0.65	0.13

Chapter 3

CAUSAL STRUCTURAL LEARNING VIA LOCAL GRAPHS

3.1 Introduction

Observational studies often involve latent variables (i.e., variables that remain unmeasured) as well as selection variables conditional on which the observations are made. Ignoring latent and selection variables may invalidate causal conclusions (Spirtes et al., 2001; Richardson and Spirtes, 2002). To account for their presence, the ancestral relationships and conditional independences among the observed variables can be represented by a maximal ancestral graph (MAG)(Richardson and Spirtes, 2002). As multiple MAGs may represent the same conditional independences, the target of estimation is the Markov equivalence class of these MAGs, which can be represented by a partial ancestral graph (PAG) (Ali et al., 2009; Zhang, 2008).

PAGs can be learned from data on the observed variables using the FCI algorithm (Spirtes et al., 2001). The FCI algorithm uses the fact that two nodes i and j are non-adjacent in the PAG if and only if the corresponding variables are conditionally independent given their D-SEP set (d-separation set). In simplified terms, this D-SEP set is comprised of ancestors that are adjacent or connected via certain collider paths (Spirtes et al., 2001). Since the D-SEP sets cannot be inferred directly, the FCI algorithm does not directly estimate the skeleton (i.e., adjacencies) of the PAG. Instead, FCI first uses an initial phase of the PC algorithm to obtain a preliminary skeleton, which is a superset of the PAG skeleton. It then uses the PC output to compute supersets of the D-SEP sets, referred to as p-D-SEP sets (possible-D-SEP sets), and estimates the final skeleton using the p-D-SEP sets. To infer the skeleton, PC and the second step of FCI both adopt a hierarchical search strategy, wherein edges are removed recursively via tests of conditional independence given subsets of increasingly larger sizes in

some search pool (neighbors in PC, p-D-SEP sets in FCI); see Section 3.3.1 for more details.

The FCI algorithm is consistent and complete in high-dimensional settings (Zhang, 2008; Colombo et al., 2012), but it is computationally expensive. This is partly because the p-D-SEP sets can be very large, leaving the final skeleton estimation step too many subsets to search amongst. Colombo et al. (2012) introduced multiple approaches for narrowing down the p-D-SEP sets, for example, by intersecting the sets with a bi-connected component (FCI_{path}), or applying conservative ordering rules (CFCI). They also proposed a fast approximation to the FCI algorithm, called RFCI, which directly estimates the final skeleton along with modified orientations, hence avoiding the computation of the p-D-SEP sets. The RFCI output, called RFCI-PAGs, in which the presence of an edge between two nodes only implies conditional dependence given subsets of their neighborhood, is generally less informative than a PAG. To reduce the cost of estimating the initial skeleton, an anytime version of FCI was proposed in Spirtes et al. (2001), and can be combined with the above modifications, but the skeleton it learns is only guaranteed to be a superset of the skeleton learned using FCI, and is therefore less informative. Claassen et al. (2013) proposed FCI+, based on an alternative construction of p-D-SEP sets. For networks with bounded maximal node degree, FCI+ has polynomial complexity in the number of nodes.

The outlined existing versions of the FCI algorithm all follow a *neighborhood-based search strategy*, in the sense that they search for separating sets among neighbors (in the PC step) and extended neighbors (in the p-D-SEP step). The computational and sample complexity of such neighborhood-based methods (including also the PC algorithm) scales with the size of the largest separator, which often scales with the maximum node degree of the graph; this is problematic in the presence of highly connected *hub nodes*, i.e., nodes with large degrees. Hub nodes abound in many real-world systems, such as biological networks and the Web (Chen and Sharp, 2004; Kleinberg et al., 1999). These networks are well-approximated by the family of power-law graphs, which have unbounded maximum degree (Kleinberg et al., 1999).

Instead of relying on the common sparsity assumption via bounded maximum node de-

gree, in this paper we exploit a *local-separation property* that holds for many large random networks. While this property — which also holds for power-law graphs containing hub nodes (Malioutov et al., 2006) — does not restrict the total number of paths between every pair of nodes, it implies a small number of *short paths* between them. In this work, we shift the focus from (global) D-SEP sets to *local* D-SEP sets, that is, D-SEP sets in the local graph. Accordingly, we shift from the neighborhood-based search strategy to a *local-graph-based search strategy*. The success of this strategy relies on an additional assumption that ensures that effects of long, non-local paths can be ignored when estimating conditional dependencies. The rationale for this assumption is that these paths carry weak dependencies that play minor roles in causal mechanisms and are unlikely to change the results of independence tests. In other words, in many settings conditional dependence relations can be learned "locally" by focusing on paths (short or long) inside the local graph. The assumptions are discussed in Section 3.4 and are compared with those of FCI. As we will show, under the assumption that dependencies can be determined locally (discussed in Section 3.4) and assuming the true MAG satisfies a local-separation property with small separator size, this strategy enjoys reduced computational and sample complexity.

Concretely, in this work, we propose a new *local FCI* (lFCI) algorithm for structure learning in the presence of latent and selection variables. The lFCI algorithm learns the skeleton of a PAG by testing conditional independences between pairs of nodes (i, j) given sets of small cardinality. However, in contrast to other algorithms, the conditioning sets are selected only from the nodes that are within short distance (therefore, local) to $\{i, j\}$. By doing so, under a different set of assumptions than those considered for FCI, lFCI can learn networks with p nodes and $O(p^a)$ maximal node degree ($a > 1$) in polynomial computational and sample complexity — in such cases, the complexity of FCI is exponential in p.

A similar idea has recently been employed in the *reduced PC* (rPC) algorithm (Sondhi and Shojaie, 2019). In the setting without latent or selection variables, rPC may offer reduced computational and sample complexity compared to PC. However, latent and selection variables as considered here pose new challenges that cannot be addressed by simply replacing

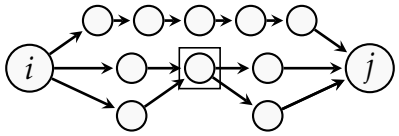


Figure 3.1: Illustration of a γ-local separator between i and j (shown in box) with $\gamma = 4$. There are 4 short paths between i and j, and the local separator is not a d-separator.

the PC steps in FCI with rPC steps. This is because rPC uses a notion of local separation in undirected graphs that does not naturally extend to mixed graphs. In addition, while rPC justifies focus on small conditioning sets through a local perspective, it does not follow the local-graph-based strategy adopted in our new lFCI. As a result, for any pair of nodes, rPC searches for separating sets among all other $p-2$ nodes, which can be computationally prohibitive. Beyond a reduction in computational cost, our local-graph-based strategy leads to high-dimensional consistency under less restrictive assumptions than those in Sondhi and Shojaie (2019).

The paper begins with graph-theoretic results on local separation in Section 3.2. The lFCI algorithm is presented in Section 3.3 and its consistency for linear SEM is established in Section 3.4. We illustrate the performance of lFCI through a simulation study in Section 3.5 and a real data application in Section 3.6.

Let $G = (V, E)$ be a graph with vertex set V and edge set E. We only consider graphs that are *simple* (i.e., there is at most one edge between any pair of nodes) and free of self-loops (i.e., each edge joins two distinct nodes). We allow three types of edge marks (head, tail and circle) and six types of edges: directed ($\rightarrow$), bi-directed ($\leftrightarrow$), undirected ($-$), nondirected ($\circ\!\!-\!\!\circ$), partially undirected ($\circ\!\!-$), and partially directed ($\circ\!\!\rightarrow$). A star $\star$ denotes an arbitrary mark on an edge; e.g., $\star\!\!\rightarrow$ represents an edge of type $\rightarrow$, $\leftrightarrow$, or $\circ\!\!\rightarrow$ in the graph.

Our terminology follows standard conventions as in Section 1.3, and we renew a few definition below in the context of mixed graphs. In particular, a graph G is directed (or undirected) if it contains only directed (or only undirected) edges. A mixed graph may

contain directed, bi-directed or undirected edges. The *skeleton* skel(G) is the undirected graph with the same adjacencies as G. A *path* is a sequence of distinct vertices, where each pair of consecutive vertices is *adjacent*, i.e., linked by an edge. A path of *length* n has $n+1$ vertices (i.e., n edges). A *directed path* is a path along directed edges following the arrowheads. Adding a directed edge back to the first node gives a *directed cycle*. A *directed acyclic graph (DAG)* is a directed graph without directed cycles. If a graph G contains the edge $k \to j$, then k is a *parent* of its *child* j. We write $\mathrm{pa}(G, j)$ for the set of all parents of a node j. Similarly, $\mathrm{ch}(G, j)$ is the set of children of j. If it contains a directed path $k \to \cdots \to j$, then k is an *ancestor* of its *descendant* j. If it contains a path $k \mathrel{\star\!\!\to} \cdots \mathrel{\star\!\!\to} j$, then k is *anterior* to j. The sets of parents, children, ancestors and descendants of j are denoted $\mathrm{pa}(G, j)$, $\mathrm{ch}(G, j)$, $\mathrm{an}(G, j)$ and $\mathrm{de}(G, j)$, respectively. We allow trivial paths, so that $j \in \mathrm{an}(G, j)$ and $j \in \mathrm{de}(G, j)$, but $j \notin \mathrm{pa}(G, j)$ and $j \notin \mathrm{ch}(G, j)$ as we exclude self-loops. If there exists a path from i to j, and the two endpoints are adjacent, then this forms a *cycle*. A triple of vertices (i, j, k) is *unshielded* if j is adjacent to both i and k, but i and k are not adjacent. A non-endpoint vertex j on a path π is a *collider* on the path if the edges preceding and succeeding it both have arrowheads at j. Otherwise, j is a *non-collider* on π. A *v-structure* is an unshielded triple (i, j, k) with j as collider. The *neighborhood* $\mathrm{adj}(G, i)$ is comprised of all nodes j adjacent to i in G. Its size $|\mathrm{adj}(G, i)|$ is the *degree* of i. The maximal degree of any vertex is denoted by $d_{\max}(G)$. When clear from the context, we will drop the indication of the graph G, writing, e.g., $d_{\max}$ or $\mathrm{an}(i)$ only.

Consider a DAG G whose vertex set V indexes a collection of random variables. Suppose V is partitioned as $V = X \cup L \cup Z$, where X indexes observed random variables, L indexes latent variables and Z indexes selection variables. As shown by Richardson and Spirtes (2002), G can be transformed into a unique maximal ancestral graph (MAG) G^* with vertex set X such that G^* retains the m-separation properties in G. The notion of m-separation, defined below, generalizes d-separation to MAGs, and the two are often used interchangeably when the graph is a DAG.

An ancestral graph is a mixed graph with directed and undirected edges, that contains

no directed cycles or almost directed cycles (i.e., cycles formed by a directed path and a bidirected edge), and no subgraph of the type $i - j \leftrightarrow\!\star\ k$. Let $\mathrm{un}(G)$ be the set of vertices in G that have no parents and are also not incident to a bidirected edge. Then in an ancestral graph, $\mathrm{un}(G)$ induces an undirected subgraph that contains all undirected edges of G. An ancestral graph is a MAG if two vertices are non-adjacent only if they can be m-separated, i.e., all paths between them can be blocked in the following sense.

Definition 1 (m-separation)**.** *A set Y blocks a path π in an ancestral graph if and only if:*

1. *π contains a triplet (i, j, k) such that j is a non-collider on this path and $j \in Y$; or*

2. *π contains a v-structure $i \star\!\rightarrow j \leftrightarrow\!\star\ k$ such that $j \notin Y$ and no descendant of j is in Y.*

If a path π from vertex i to vertex j is not blocked by Y, then π is also said to m-connect i and j given Y. If Y blocks every path between i and j, then i and j are m-separated given Y.

If i and j are m-separated by as set S, we say S is a m-separator of i, j. Moreover, we say S is *minimal* if it has the smallest cardinality among all m-separators of i, j. The MAGs that have the same set of m-separation relations form a Markov equivalence class. We denote the Markov equivalence class of G^* as $[G^*]$. We say an edge mark is *invariant* in $[G^*]$ if it is the same in all members of $[G^*]$. The Markov equivalence class can be represented by a *partial ancestral graph* (PAG), H, with three types of edge marks (head, tail and circle) that has the same adjacencies as G^*, and each non-circle edge mark in H is an invariant mark in $[G^*]$. For a given MAG, there may be more than one PAG that represents its Markov equivalence class. However, there is a unique PAG that is *maximally informative* in the sense that every non-circle edge mark is invariant, and every circle edge mark is variant. Alternatively, PAGs can be characterized as following:

Definition 2 (PAG)**.** *Let $G = (X \cup L \cup Z, E)$ be a DAG, and H be a simple graph with vertex set X and edges of the type $\rightarrow$, $\circ\!\!\rightarrow$, $\circ\!\!-\!\!\circ$, $\leftrightarrow$, $-$, or $\circ\!\!-$. Then H is a PAG representing G if and only if the following four conditions hold:*

1. *The absence of an edge between two vertices i and j in H implies that there exists a subset $Y \subseteq X \setminus \{i, j\}$ such that i and j are m-separated given $(Y \cup Z)$.*

2. *The presence of an edge between two vertices i and j in H implies that i and j are m-connected given $(Y \cup Z)$ for all subsets $Y \subseteq X \setminus \{i, j\}$.*

3. *If an edge between i and j in H has an arrowhead at j, then $j \notin \mathrm{an}(G, i \cup Z)$.*

4. *If an edge between i and j in H has a tail at j, then $j \in \mathrm{an}(G, i \cup Z)$.*

Given a vertex set V, a formal conditional independence statement is a triple denoted $A \perp\!\!\!\perp B|C$, where $A, B, C \subset V$ are non-empty and pairwise disjoint. The independence model defined by a MAG G, denoted $\mathcal{I}(G)$, is the set of formal conditional independence statements $A \perp\!\!\!\perp B|C$ for which A and B are m-separated by C in G. A probability distribution P obeys the model $\mathcal{I}(G)$ if all formal statements in $\mathcal{I}(G)$ are also probabilistic conditional independences in P. Such a distribution P is *faithful* to G if the conditional independence relations in P are exactly the same as $\mathcal{I}(G)$. If the distribution P over $X \cup L \cup Z$ is faithful to a DAG, and G is the MAG obtained by conditioning on Z and marginalizing L, then the absence of an edge between i and j in G implies that there exists some set $Y \subseteq X \setminus \{i, j\}$ such that $i \perp\!\!\!\perp j|Y \cup Z$, and the presence of an edge between i and j implies $i \not\perp\!\!\!\perp j|Y \cup Z$ for all $Y \subseteq X \setminus \{i, j\}$.

3.2 Local separation

3.2.1 Local separation and local paths

Our algorithm, presented in Section 3.3, is based on a *local separation property* that holds for many common networks. The property yields that short m-connecting paths between non-adjacent nodes can be blocked by small sets. A sufficient condition for the local separation property is the *local path property*, which involves a length parameter λ and a path count parameter η. These will be specified later for specific graphs.

Definition 3 ((η, γ)-local path property)**.** *Let $\eta, \gamma \geq 1$ be integers. An undirected graph G satisfies the (η, γ)-local path property if for any two non-adjacent nodes, there are at most η paths between them with length no longer than γ.*

While it does not restrict the total number of paths, the (η, γ)-local path property implies that the number of short paths between non-adjacent nodes is bounded. Many random graph processes generate sequences of undirected graphs with increasing vertex set size p that (for process-specific η and γ) satisfy the (η, γ)-local-path property with probability tending to 1 as $p \to \infty$. Examples include Erdős-Renyi graphs (Bollobás, 2001; Anandkumar et al., 2012b), power-law random graphs with strongly finite mean (Chung and Lu, 2006; Dembo and Montanari, 2010; Dommers et al., 2010), and Δ-regular random graphs (McKay et al., 2004). Surprisingly, in all these cases, the constants can be chosen as $\eta = 2$ and $\gamma = O(\log p)$.

Local separation in undirected graphs does not naturally extend to directed and mixed graphs, since m-separation is not implied by undirected graph separation. To overcome this issue, Sondhi and Shojaie (2019) consider a directed "local separator" between two nodes as the smallest set that blocks all short d-connecting paths between them. However, this "local separator" may unblock a large number of long d-connecting paths (see Figure 3.2). This is particularly problematic when edges amongst "local separator" nodes form dense structures. As a consequence, consistency of rPC requires strong assumptions on the data-generating distributions (under which the "local separators" act like true separators). In this work, we mitigate this limitation by focusing instead on subgraphs induced by nodes on short paths.

Definition 4 (Local graph)**.** *For a graph $G = (V, E)$ and two nodes $i, j \in V$, let $P(G, i, j)$ be the set of all paths between i and j, and let $P_\gamma(G, i, j)$ be the set of those that are not longer than γ. The γ-local graph of $\{i, j\}$, denoted $G_\gamma(i, j)$, is the vertex-induced subgraph of G induced by the set $V_\gamma(i, j) = \{v \in V : v \in \pi \text{ for some } \pi \in P_\gamma(G, i, j)\}$.*

The motivation for our definition is that subgraphs better capture causal relations than subsets of paths. Moreover, m-separators in local graphs are interpretable. To distinguish our concept from Sondhi and Shojaie (2019), we call our separator the *local-graph separator*.

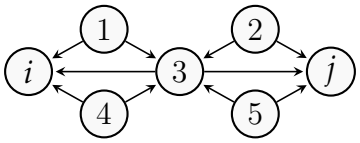


Figure 3.2: A graph G with $G = G_\gamma(i, j)$ for $\gamma = 3$; every node lies on a path of length at most three between i and j. The set $\{2, 3, 5\}$ is a γ-local-graph separator. In contrast, the definition of Sondhi and Shojaie (2019) makes the set $\{3\}$ a γ-"local separator" of (i, j), although there are 4 "local but not short" paths (e.g., $i - 1 - 3 - 2 - j$) that are d-connected given $\{3\}$.

Definition 5 (γ-local-graph separator)**.** *Let i, j be two nodes in a MAG $G = (V, E)$. A γ-local-graph separator of (i, j) is a subset $S \subset V_\gamma(i, j)$ that m-separates i and j in $G_\gamma(i, j)$.*

A local-graph separator is a genuine m-separator in the subgraph: It blocks not only short, but also long paths in the local graph; see Figure 3.2. However, a set that m-separates i, j in $G_\gamma(i, j)$ does not necessarily m-separate them in G (see Figure 3.1). Hence, local-graph separations are not necessarily reflected in the independence model $\mathcal{I}(G)$.

Remark 1. *The existence of a local-graph separator between two nodes is equivalent to the absence of an edge between them in the MAG $G = (V, E)$. To see this, fix $\gamma \geq 1$. For any set $U \subseteq V$ and $i, j \in U$, it holds that if i, j are d-separated by S_{ij} in G then they are d-separated by $S_{ij} \cap U$ in the subgraph of G induced by U. Therefore, two nodes are non-adjacent in G if and only if they are γ-local-graph separated by some set S_γ.*

Though local-graph separators might be less informative than m-separators in the full graph, they usually have bounded size even in graphs with unbounded node degrees. Local separators can be useful in such cases, as the minimal m-separators in the full graphs might be large.

3.2.2 Graphs with small local-graph separators

The computational and sample complexities of PC and FCI algorithms depend on the maximum size of d-(or m-) in the underlying graph. The applicability of the algorithms, both statistically and computationally, is thus limited to settings where the neighborhood-based D-SEP sets are all small. However, this does usually not hold in the presence of nodes with large degree (cf. Appendix B.5), which are common to many real-world networks. Nevertheless, such networks are often still sparse, in the sense of having small γ-local-graph separators. The algorithm proposed in this paper exploits this weaker notion of sparsity to offer sample and computational complexities that depend only on the maximum size of γ-local-graph separators, i.e., on

$$L(G, \gamma) = \max_{(i,j) \notin E} \min_{S \in \mathcal{S}_\gamma(i,j)} |S|,$$

where $\mathcal{S}_\gamma(i, j)$ is the set of all γ-local separators of nodes i, j in a mixed graph $G = (V, E)$.

If a DAG G has maximal in-degree at most $\Delta > 0$, then for arbitrary $\gamma \in \mathbb{N}$, it holds that $L(G, \gamma) \leq \Delta$. We next show that a similar upper bound holds more generally, as long as the DAG satisfies the local-path property, which allows for potentially unbounded node degrees.

Lemma 6. *If the skeleton of a DAG G satisfies the (η, γ)-local path property, then it holds that $L(G, \gamma) \leq \eta$.*

Proof. Fix any non-adjacent nodes i and j. Since G is acyclic, two nodes cannot be ancestors of each other in G or any subgraph. Without loss of generality, suppose $i \notin \text{an}(G, j)$, and let $S = \text{pa}(G_\gamma(i, j), i)$. ($S$ is defined analogously if $j \notin \text{an}(G, i)$.) Then paths between i and j in graph $G_\gamma(i, j)$ either have a non-collider included in $\text{pa}(G_\gamma(i, j), i)$, or have a collider that is not in $\text{pa}(G_\gamma(i, j), i)$. Thus, $\text{pa}(G_\gamma(i, j), i)$ is a d-separator. We complete the proof by the pigeonhole principle: Since $G_\gamma(i, j)$ is induced by $P_\gamma(G, i, j)$, for each $v \in \text{ne}(G_\gamma(i, j), i)$, the edge (i, v) (i.e., pigeonhole) must lie on at least one short path between i and j. (i.e., pigeons). Therefore, $\eta \geq |\text{ne}(G_\gamma(i, j), i)| \geq |S|$. □

Lemma 6 suggests that in many large random DAGs, the maximum node-degree of the local-path graph can be small while the maximum node-degree may be large. Since the proof employs a construction of neighborhood-based separators, as a corollary, for any non-adjacent pair (i, j) where $i \notin AN(G, j)$, the choice of $S = \text{pa}(G_\gamma(i, j), i)$ is a local separator and $|S| \leq \eta$. As a corollary to the construction we employed in the proof, the "approximation" of rPC in Sondhi and Shojaie (2019) is in fact an exact algorithm if the local path property holds (see Corollary 1).

The problem is more complicated for MAGs, which may have large minimal separators even with bounded degree (with bidirected edges m-separation of non-adjacent nodes generally requires consideration of non-neighboring nodes). Thus, compared with PC, the FCI theory requires an additional assumption on the size of the algorithm's possible-d-separation sets. The theory for RFCI imposes a similar limit on the size of separators in the initial step (Colombo et al., 2012). The FCI+ theory exploits an assumption of bounded node degree to avoid additional assumptions on the size of bidirected components (Claassen et al., 2013). However, these results either prohibit the existence of generic hub nodes with large degrees or have sub-par sample and computational complexities. In contrast, the size of minimal local-separation sets is determined by short paths in $G_\gamma(i, j)$. Thus, by utilizing local-graph separators, and under the additional assumptions discussed in Section 3.4, our framework achieves improved computational and sample complexity. Next we show that as long as the skeleton of a MAG has small number of short paths, the size of the local-separators is controlled:

Lemma 7. *If the skeleton of a MAG G satisfies the (η, γ)-local path property with $\eta \leq 3$, then $L(G, \gamma) \leq \eta$.*

Proof. This lemma is proved by enumeration of all possible graph configurations. Details are given in Appendix B.1. □

Lemma 7 covers most common random graphs. For instance, for Erdős-Renyi graphs, power-law graphs with strongly finite mean, and Δ-regular graphs, it holds with $\eta \leq 2$. As

we discuss in Section 3.3, for these graphs, we only need to search for separators of size up to 2.

Our next result shows that we can further reduce the size of separator by restricting focus on pairs of nodes in *local Markov blankets*. The *Markov blanket* of node i in a MAG G, denoted $\mathrm{mb}(G, i)$, is the minimal set of vertices that separates i from all other vertices. Concretely, it is the union of vertices connecting to i through either an edge, or a collider path (i.e., a path on which all non-endpoints are colliders). The γ-local Markov blanket $\mathrm{mb}_\gamma(G, i)$ is the union of vertices connecting to i through either an edge or a collider path of length at most γ.

Lemma 8. *Let $G = (V, E)$ be a MAG, and define*

$$L^{mb}(G, \gamma) = \max_{(i,j) \notin E, i \in \mathrm{mb}_\gamma(G,j)} \min_{S \in \mathcal{S}_\gamma(i,j)} |S|.$$

If the skeleton of G satisfies the (η, γ)-local path property with $\eta \leq 4$, then

$$L^{mb}(G, \gamma) \leq \max(0, \eta - 1).$$

Proof. Suppose $i \notin \mathrm{adj}(G, j)$ but $i \in \mathrm{mb}_\gamma(G, j)$. Then i and j must be connected via a collider path π. There must be a node on π, call it u, that is not ancestral to i and j, because π would otherwise prevent m-separation of i and j, which contradicts G being a MAG. Let $G_\gamma^{-u}(i, j)$ be the subgraph of $G_\gamma(i, j)$ induced by the complement of u. Every minimal separator of (i, j) in $G_\gamma^{-u}(i, j)$ also minimally separates i and j in $G_\gamma(i, j)$ (see, e.g., Zander and Liśkiewicz, 2020). It is easy to see that $G_\gamma^{-u}(i, j)$ has at most $\eta - 1$ many short paths between i and j. Hence, the result follows from Lemma 7. □

More generally, our framework accommodates hybrid graphs, consisting of a "global" graph with small maximal degree, and a "local" graph with bounded local-paths, paralleling the class of undirected hybrid graphs defined in Chung and Lu (2006). As a concrete example, the Watts-Strogatz (or small-world) graph consists of the union of a d-dimensional regular graph and an Erdős-Renyi random graph (Watts and Strogatz, 1998).

Theorem 5. *Let $G = (V, E)$ be a MAG. For any two non-adjacent nodes i and j, let M_{ij} be the set of nodes that do not lie on any path in $P(G, i, j)$ that uses a bidirected edge. Suppose for each pair of non-adjacent nodes i, j, there exists a set $M \subseteq M_{ij}$ such that the subgraph of G induced by $M \cup \{i, j\}$ has node-degree no larger than Δ, and the subgraph of G induced by $V \setminus M$ satisfies the local path property with some $\eta_0 \leq 3$ and some γ. Let $\eta = \eta_0 + \Delta$. The following statements hold,*

$$L(G, \gamma) \leq \eta \quad \textit{and} \quad L^{mb}(G, \gamma) \leq \max(0, \eta - 1).$$

Proof. Let i, j be non-adjacent nodes in G. Without loss of generality, let $i \notin \text{an}(G, j)$. Let G^1 and G^2 be the subgraphs induced by $M \cup \{i, j\}$ and $V \setminus M$, respectively. Let $S^1 = \text{pa}(G^1, i)$, and let S^2 be a γ-local-graph separator of (i, j) in G^2. We have $P(G, i, j) = P(G^1, i, j) \sqcup P(G^2, i, j)$ where $\sqcup$ stands for disjoint union. Thus, $S^1 \sqcup S^2$ is a γ-local-graph separator of (i, j). By Lemma 7, $|S^1| + |S^2| \leq \Delta + \eta_0$. Lemma 8 gives the Markov blanket result. □

3.3 A Local FCI Algorithm (lFCI)

In this section, we propose a novel algorithm that discovers absent edges by searching for local-graph separators, as defined in Section 3.2.

3.3.1 lFCI

To learn a MAG $G = (V, E)$, PC/FCI adopt the following strategy. Starting with a complete undirected graph C, they first search for separating sets of size $\ell = 0$: If two nodes are independent given a set of size 0 (i.e., marginally independent), the corresponding edge in C is removed. Iteratively increasing the value of ℓ by one, the algorithm visits all pairs (i, j) adjacent in C and searches amongst all sets $S \subseteq J(i, j, C)$ with $|S| = \ell$, where $J(i, j, C) \subseteq V \setminus \{i, j\}$ is a current *search pool*. If a conditional independence $i \perp\!\!\!\perp j | S$ is found, the edge $i - j$ is removed from C. The algorithm stops when ℓ exceeds the maximum size of the sets in the search pool. (In rPC, the iterations are stopped early at a specified level for ℓ.) The

value of ℓ at termination is the *reach level*, denoted m_{reach}. With a conditional independence oracle, PC terminates at $m_{\text{reach}}(\text{PC}) \leq d_{\max} - 1$, where $d_{\max}$ is the maximum node degree; the reach level of the second step of FCI is the maximum size of p-D-SEP sets.

Our lFCI algorithm follows a similar strategy but with two key differences. The first difference is the construction of the search pool, $J(i, j, C)$. Given a working skeleton C that is a supergraph of skel(G), a construction of $J(i, j, C)$ is valid if each pair of non-adjacent nodes (i, j) is separated by some subset of $J(i, j, C)$. PC/FCI adopt a neighborhood-based strategy: PC uses $J_{\text{PC}}(i, j, C) = (\text{adj}(i, C) \cup \text{adj}(j, C)) \setminus \{i, j\}$, and FCI uses J_{PC} in its first step and $J_{\text{FCI}}(i, j, C) = \text{p-D-SEP}(i, j)$ in its second step. In contrast, inspired by the local separation property, our lFCI algorithm adopts a local-graph-based strategy, in which we form an alternative search pool that is guaranteed to contain a local separator by including the nodes that are *close* to both i and j. Figure 3.3 exemplifies the difference between neighborhood-based and local-graph-based searches. More concretely, let $D_G(i, j) = \min_{\pi \in P(G,i,j)} |\pi|$ be the shortest-undirected-path distance between nodes i and j in G, with $D_G(i, j) = \infty$ if $P(G, u, v) = \emptyset$. Writing C_{-ij} for the working skeleton C with edge $i - j$ removed, we define

$$J_\gamma(i, j, C) = \left\{k \in V \setminus \{i, j\} : D_{C_{-ij}}(i, k) + D_{C_{-ij}}(j, k) \leq \gamma\right\}. \tag{3.1}$$

While otherwise distinct, the idea of searching among nodes that lies on connecting paths is related to the path modification of FCI in FCI_{path} (Colombo and Maathuis, 2014), which uses a different search pool, $J_{\text{FCI}_{\text{path}}}(i, j, C) = J_{\text{FCI}}(i, j, C) \cap J_p(i, j, C)$. The following lemma shows that $J_\gamma(i, j, C)$ is a superset of $V_\gamma(i, j)$ from Definition 4 and is hence a valid search pool.

Lemma 9. *Let G be a MAG, and let C be a super-graph of skel(G). Two nodes i, j are non-adjacent in G if and only if they are γ-local-graph separated by a subset of $J_\gamma(i, j, C)$.*

The second innovation in our approach lies in its termination criterion. The complexities of PC and FCI are determined by their reach levels m_{reach}, which scale with the maximum degree $d_{\max}$ of the graph. As a result, these values can be very large if the graph includes

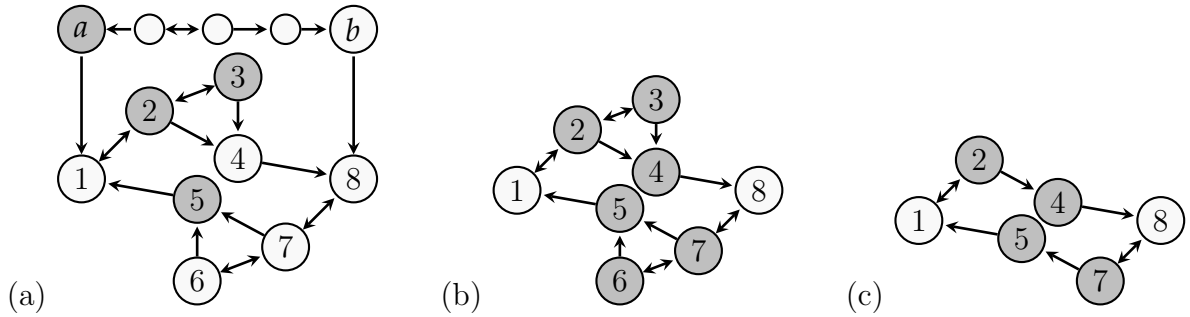


Figure 3.3: Search strategies of FCI and lFCI: (a) True G, (b) local-graph $G_4(1,8)$, (c) local-graph $G_3(1,8)$. Search pools $J_{\text{FCI}}(1,8)$, $J_4(1,8)$ and $J_3(1,8)$ are shaded. FCI discovers the separator $\{a,2,3,5\}$ in G (differs from minimal separator $\{a,4,5\}$). For both $\gamma = 3,4$, lFCI discovers the local-graph separator $\{4,5\}$ (small but only correct in the local graph). Note that FCI cannot be early-stopped at reach level 2 even if we ignore the path $(1,a,\ldots,b,8)$.

hub nodes. In particular, when considering sequences of structure learning problems where a few nodes are allowed to have $O(p)$ many neighbors, PC and FCI (and also FCI+) cannot terminate in polynomial time. Our approach offers a strategy to circumvent this problem by early termination when searching on local graphs. Indeed, sparse graphs may still satisfy the conditions in Theorem 5 for bounded η. The reach level of our local-graph-based approach is then at most η, so $O(1)$. Therefore, through its focus on local graphs, our lFCI algorithm may enjoy polynomial-time complexity even in graphs with hub nodes—settings that become problematic for PC and FCI. We emphasize that an ad hoc early stopping (or "anytime") version of PC and FCI avoids the computational issue (Spirtes et al., 2001) but will generally result in false discoveries. Indeed, even if the conditions in Theorem 5 hold, the smallest neighborhood-based D-SEP set is not necessarily small; compare Figure 3.3.

Our lFCI proposal is summarized in Algorithm 15. Starting with a complete graph C and level $\ell = 0$, lFCI traverses every edge (i,j) and searches for a conditional independence $i \perp\!\!\!\perp j | S$ given subsets $S \in J_\gamma(i,j,C)$ of size $|S| = \ell$. If a conditional independence is found the edge is removed from C. The level ℓ is increased after checking all edges, and the

algorithm terminates when ℓ hits the reach level $m_{\text{reach}} = \eta$, which is picked in advance with a view towards potential underlying graph structure (similar to how node degrees are bounded in other algorithms). Throughout the search, we keep track of the shortest-path-distances between nodes, but only update them after completing the ℓ-th level. This makes the algorithm *order-independent*, i.e., the output does not depend on the order in which edges are tested (see, e.g., Colombo and Maathuis, 2014).

3.3.2 Tuning parameters

Given Lemma 9, γ acts as a tuning parameter that controls the breadth of the search in our algorithm. Theoretically, γ should be small enough such that the underlying graph G has small γ-local-graph separators, yet large enough such that paths not contained in G_γ contribute little to total effects in the graphical model. Theorem 5 notes that many random graphs satisfy $L(G, \gamma) \leq \eta$ with $\gamma = O(\log p)$ with high probability as the number of nodes $p \to \infty$. Due to this fact, our later analysis allows (but does not require) γ to grow with p, in which case distributional conditions may be weakened for larger graphs. We note that in our later simulations lFCI terminates early, and its performance is rather insensitive to our choice of γ, see SectionB.4

The maximum size of separating sets, η, can also be seen as a tuning parameter that controls the depth of the search and allows lFCI to terminate at smaller levels than PC/FCI. Choosing η is akin to an a priori choice of the maximum node degree in other algorithms, recall Theorem 5.

3.3.3 Orientation rules

After inferring the skeleton using conditional independence tests, we orient as many edges as possible to obtain a PAG representation of the Markov equivalence class of MAGs. Given the undirected skeleton of the true MAG and a collection of minimal separators, the orientation procedure proposed in Zhang (2008) applies eleven deterministic rules to obtain the maximally informative PAG. In other words, the population version of FCI is sound (i.e.,

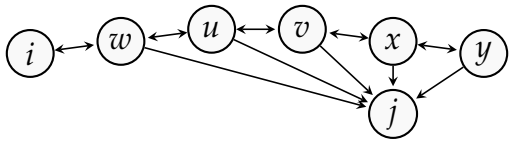


Figure 3.4: Nodes i and j are m-separated given $\{w, u, v, x, y\}$ and all other non-adjacent pairs are marginally m-separated. There is a discriminating path (i, w, u, v, x, y, j) for y. In FCI, the edge $y \to j$ is oriented correctly. For $\gamma = 5$, the γ-local separator of (i, j) is $\{w, u, v, x\}$, with which the discriminating path rule outputs $y \leftrightarrow j$, which is inconsistent with the truth.

never returns a wrong result) and complete (i.e., the output is maximally informative in the sense of discovering all causal relations common to the graphs in the equivalence class). However, these properties are *not* guaranteed if we apply the rules directly with local-separation, because the local separators are usually not m-separators.

Nonetheless, if the estimated skeleton C in Algorithm 15 is the true skeleton of G, correct orientation using local-separation sets can be achieved with only a single change to Zhang's Rule $\mathcal{R}_4$, which pertains to *discriminating paths*. A path between i and j, $\pi = (i, \ldots, x, y, j)$, is a discriminating path for y if it includes at least three edges; y is adjacent to j on π; i is not adjacent to j; and every vertex between i and y is a collider on π and a parent of j. Figure 3.4 illustrates the failure of the unmodified discrimination path rule (Rule $\mathcal{R}_4$). The original (Zhang, 2008) and the modified versions may be contrasted as follows:

$\mathcal{R}_4$: If π is a discriminating path between i and j for y, and $y \circ\!\!\rightarrow\!\!\star j$, then if $y \in S(i, j)$, orient $y \circ\!\!\rightarrow\!\!\star j$ as $y \to j$; otherwise orient the triple (x, y, j) as $x \leftrightarrow y \leftrightarrow j$.

$\mathcal{R}'_4$: If π is a discriminating path between i and j for y, and $y \circ\!\!\rightarrow\!\!\star j$, then if $v \in S(i, j)$, orient $y \circ\!\!\rightarrow\!\!\star j$ as $y \to j$; if $y \notin S(i, j)$ and all vertices in π are contained in the γ-local-graph of (i, j), then orient (x, y, j) as $x \leftrightarrow y \leftrightarrow j$; otherwise orient $y \circ\!\!\rightarrow\!\!\star j$ as $y \circ\!\!\rightarrow j$.

Rule $\mathcal{R}'_4$ avoids wrong decisions when local-graph separators do not provide enough informa-

tion. The original and modified rules give the same output under the following condition.

Assumption 1 (Local discriminating paths)**.** *Let G be a MAG and γ be an integer. Denote $\Pi^D(G, i, j, y)$ as the set of discriminating paths between i and j for y in G. If $\Pi^D(G, i, j, y) \neq \emptyset$ for the triple (i, j, y), then there exists $\pi \in \Pi^D(G, i, j, y)$ such that $\pi \subset G_\gamma(i, j)$.*

In the next lemma we show that, given correct skeleton and local-separation sets, the new set of rules is correct and maximally informative.

Lemma 10. *Let G be a MAG. Let η and γ be integers such that $\gamma > 2$ and $L(G, \gamma) \leq \eta$. Suppose that in Algorithm 15, the estimated skeleton C is equal to the skeleton of G, and all* SEP *sets are local-graph (or full-graph) separators. Then the output of Algorithm 15, $\widehat{G}$, is a PAG for $[G]$. If in addition Assumption 1 holds, then $\widehat{G}$ is maximally informative.*

Proof. In FCI, Zhang's orientation rules $\mathcal{R}_0$ (unshielded triple rule) and $\mathcal{R}_4$ (discriminating path rule) introduce arrowheads using the separation sets, whereas the rest of the rules only depend on the results of $\mathcal{R}_0$ and $\mathcal{R}_4$. Therefore, the orientation phase makes no mistake if all arrowheads introduced by $\mathcal{R}_0$ and $\mathcal{R}_4$ are correct, and maximally informative if $\mathcal{R}_0$ and $\mathcal{R}_4$ introduce as many arrowheads as possible. For details of the rules, see Zhang (2008).

First we consider $\mathcal{R}_0$, which orients an unshielded triple (i, j, k) into a v-structure if j is not in the separator for (i, k). In lFCI, given C is the true skeleton, if j is not in the γ-local separator, then (i, j, k) must be a marginally blocked path in both G and $G_\gamma(i, k)$. Thus, $\mathcal{R}_0$ produces the same output using local- and full-graph-separators. Next we show $\mathcal{R}_4'$ orients correctly. Indeed, if π is a discriminating path between i, j for y and $y \in S_\gamma(i, j)$, then the last edge on π must be oriented as $y \to j$ to avoid unblocking π. If $y \notin S_\gamma(i, j)$ and π is contained in the local-graph, then $\mathcal{R}_4'$ is the same as $\mathcal{R}_4$ in $G_\gamma(i, j)$; otherwise, $\mathcal{R}_4'$ simply avoids making wrong decisions. We conclude that by Theorem 1 of Zhang (2008), the lFCI output is correct.

Under Assumption 1, if (i, j, y) can be oriented using a full-graph separator by $\mathcal{R}_4$, then there is a discriminating path in $G_\gamma(i, j)$ such that the same orientation is made by $\mathcal{R}_4'$. Therefore, the output of lFCI is identical to that of FCI, which is maximally informative. □

Unlike FCI, our lFCI algorithm does not guarantee skeleton recovery under a conditional independence oracle, unless all conditional dependencies can also be determined locally. This is because conditional independence need not always correspond to local-separation. Therefore, our theoretical guarantees in Section 3.4 will be concerned with high-dimensional consistency. More specifically, we focus on the regime where the high-dimensional sparsity entails that, up to effects of lower order, conditional dependence corresponds to local structure in the graph.

3.3.4 Computational complexity

As discussed in Section 3.3.1, the computational advantages of lFCI over FCI, RFCI and FCI+ stem from two key differences: (i) the use of a local-graph-based strategy (J_γ) instead of the neighborhood-based strategy (J_{FCI}); and (ii) searching up to sets of size η. As a result, for graphs satisfying the local-separation property with small η, lFCI achieves computational complexity $O(p^{\eta+2})$, which is the same as that of rPC (Sondhi and Shojaie, 2019). In contrast, the worst case computational complexity of FCI is exponential and FCI+ has computational complexity $O(p^{d_{\max}+2})$. Though FCI+ offers polynomial complexity when $d_{\max}$ is bounded (Claassen et al., 2013), it becomes inefficient in the setting of power-law graphs with highly connected hub nodes, when $d_{\max} = O(p^a)$ for some $a > 0$ (Molloy and Reed, 1995). For these graphs, FCI+ offers exponential complexity $O(p^{p^a+2})$, compared with $O(p^4)$ for lFCI.

We note that lFCI's computational advantages depend on the local separation property: while lFCI improves computational efficiency when the true MAG has small local-separators, it can also lead to reduced efficiency if the true MAG does not satisfy satisfy local-separation with small η and γ. For example, for graphs with small graph-diameter or high local-connectivity, lFCI may take more searching and testing steps, compared with FCI and RFCI.

3.3.5 Initialization with Moral Graph

Following the observation from Lemma 8, in Algorithm 4 we propose a modified version of lFCI that utilizes local Markov blankets for improved computational and sample complexities. Concretely, instead of starting with a complete graph C, Algorithm 4 starts with an estimated *local moral graph*.

The moral graph of a MAG G is the undirected graph in which two nodes i, j are adjacent whenever one node is in the G-Markov blanket of the other, say $j \in \mathrm{mb}(G, i)$. Accordingly, the local moral graph is the undirected graph obtained by taking instead the γ-local Markov blankets, $\mathrm{mb}_\gamma(G, i)$. For large enough γ, these two notions coincide as the local moral graph differs from the moral graph only if the shortest path between two nodes is longer than γ and only includes bidirected edges. This is unlikely in large common random graphs if we allow γ to increase with p; see SectionB.7 of the Supplementary Material. Consequently, we simply employ the moral graph, which in the models we treat later can be estimated by the support of the inverse covariance matrix. In our simulation study, we used score matching to estimate the precision matrix in high dimensions (Lin et al., 2016; Yu et al., 2019), which here coincides with the SCIO algorithm (Liu and Luo, 2015). The solution to this special case of penalized score matching problem can also be formulated as the SCIO algorithm (Liu and Luo, 2015). For high dimensional problems, the loss could suffer from unbounded penalization. As suggested in Yu et al. (2019), we circumvent this problem by amplifying the diagonal entries of the covariance matrix.

Initializing C as an estimated moral graph may significantly recude the size of the search pools $J_\gamma(i, j, C)$. Moreover, this additional step allows Algorithm 4 to terminate at level $\eta - 1$ instead of η (Lemma 8). Moreover, the search pools $J_\gamma(i, j, C)$ are significantly reduced when C is initialized as an estimated moral graph. Consequently, the screening step considerably reduces the search space. For Erdős-Renyi graphs, power-law graphs with strong finite mean, and Δ-regular graphs, the algorithm can then stop after checking separating sets of size 0 and 1 only. However, these improvements require slightly more restrictive conditions in

theoretical analysis (see Appendix B.2). We will explore the performance of Algorithm 4 in Section 3.5.

3.4 Consistency of the lFCI Algorithm

In this section, we establish the consistency of our lFCI algorithm in high-dimensional settings, i.e., when the number of nodes in the graph is potentially larger than the available sample size. To this end, we first discuss in detail a set of assumptions under which lFCI is consistent. We focus on linear structural equation models with sub-Gaussian noise. Proofs are provided in Appendix B.1.

3.4.1 Linear Structural Equation Models

Let $G = (V, E)$ be a MAG with vertex set $V = \{1, \dots, p\}$. Let $W = (W_1, \dots, W_p)$ be an associated random vector. The linear structural equation model given by G assumes that W solves an equation system of the form

$$W = BW + \epsilon, \tag{3.2}$$

where $B = (\beta_{i,j}) \in \mathbb{R}^{p \times p}$ is a matrix of unknown parameters with $\beta_{ij} \neq 0$ only if $j \to i$ is an edge in G. The random vector $\epsilon = (\epsilon_1, \dots, \epsilon_p)$ is comprised of stochastic noise with positive definite covariance matrix $\mathrm{Var}(\epsilon) = \Omega = (\omega_{i,j})$. It can be partitioned into two independent subvectors $\epsilon_{\mathrm{un}(G)}$ and $\epsilon_{V \setminus \mathrm{un}(G)}$. Here, $\epsilon_{\mathrm{un}(G)}$ is assumed to satisfy the global Markov property for the undirected subgraph induced by the undirected part $\mathrm{un}(G) \subseteq V$, and $\epsilon_{V \setminus \mathrm{un}(G)}$ satisfies the global Markov property for the subgraph formed by the bidirected edges among nodes in $V \setminus \mathrm{un}(G)$; compare, e.g., Drton and Richardson (2008). In particular, ϵ_i and ϵ_j are marginally independent when $i, j \notin \mathrm{un}(G)$, and conditionally independent given all other errors when $i, j \in \mathrm{un}(G)$. Consequently, the error covariance matrix Ω can be permuted into block-diagonal form with two blocks. One block has an inverse whose support is given by the undirected edges of G, and the other block has its support given by the bidirected edges of G (Richardson and Spirtes, 2002, Section 8).

Let I be the identity matrix. Since G is a MAG, it does not contain any directed cycles. Thus $I - B$ is invertible, and (3.2) has a unique solution W with covariance matrix

$$\Sigma := \mathrm{Var}\,(W) = (I - B)^{-1}\Omega(I - B)^{-\top}. \tag{3.3}$$

Conditional independences in the linear SEM correspond exactly to zero *partial correlations*. For nodes i and j, and $S \subseteq V \setminus \{i, j\}$, the partial correlation of W_i and W_j given W_S is

$$\rho(i, j|S) = \Sigma(i, j|S)/\sqrt{\Sigma(i, i|S)\Sigma(j, j|S)},$$

where $\Sigma(i, j|S) = \Sigma(i, j) - \Sigma(i, S)\Sigma(S, S)^{-1}\Sigma(S, j)$. Given a sample of n independent observations generated from the distribution of W, the corresponding *sample partial correlations* $\widehat{\rho}(i, j|S)$ are obtained by replacing Σ by the sample covariance matrix $\widehat{\Sigma}_n$. In order to test $i \perp\!\!\!\perp j|S$ in a practical run of the lFCI algorithm, we test the vanishing of $\rho(i, j|S)$ and reject the conditional independence if $\sqrt{n - |S| - 3}\left|g\left(\widehat{\rho}(i, j|S)\right)\right| > \Phi^{-1}(1 - \alpha_n/2)$, where $g(\rho) = \frac{1}{2}\log\left(\frac{1+\rho}{1-\rho}\right)$ is Fisher's z-transform, Φ is the normal cdf, and $\alpha_n \in (0, 1)$ is a significance level.

3.4.2 Consistency

Consider a sequence of learning problems indexed by p, with a sequence of MAGs $\{G_p\}$ and i.i.d. samples of size $n = n_p$, each sample being generated from a distribution that is Markov to the respective MAG. When the context is unambiguous, we suppress the subscript p.

As discussed in Section 3.2, our algorithm requires an assumption on the size of γ-local-graph separators.

Assumption 2 (Local-separation Property)**.** *The MAG G satisfies $L(G, \gamma) \leq \eta$ where the lFCI parameter η is fixed but γ may grow with p.*

As shown in Section 3.2, many common random graphs satisfy Assumption 2 with $\eta = O(1)$ and $\gamma = O(\log p)$ with high probability as the number of nodes $p \to \infty$. If the graph satisfies the requirements of Theorem 5, then Assumption 2 holds with $\eta = \eta_0 + \Delta$.

We also require an assumption on the covariance matrix to establish large sample consistency of estimated conditional correlations in high dimensions.

Assumption 3 (Covariance/precision matrix). *The random vector $W = (W_1, \ldots, W_p)$ follows a linear SEM of the form* (1.2)*, with sub-Gaussian errors ϵ. Moreover, for the lFCI parameter η in Assumption 2 the spectral norms of all $(\eta + 2) \times (\eta + 2)$ submatrices of its covariance matrix Σ are bounded by a constant $0 < M < \infty$, so*

$$\max_{A \subseteq [p], |A| \leq \eta+2} \left(\|\Sigma_{A,A}\|, \|(\Sigma_{A,A})^{-1}\| \right) \leq M < \infty.$$

As in Sondhi and Shojaie (2019), we assume a faithfulness condition that is less restrictive than the λ-strong faithfulness assumption that appears, e.g., in Colombo et al. (2012).

Definition 6 ((η, λ)-strong-path-faithfulness). *Given $\eta > 0$ and $\lambda \in (0, 1)$, a distribution P is (η, λ)-strong-path-faithful to a MAG $G = (V, E)$ if both of the following conditions hold:*

(i) $\min\{|\rho(i, j|S)| : (i, j) \in E, S \subset V \setminus \{i, j\}, |S| \leq \eta\} > \lambda$, *and*

(ii) $\min\{|\rho(i, j|S)| : (i, j, S) \in N_G\} > \lambda$, *where N_G is the set of triples (i, j, S) such that i and j are not adjacent, but for some $k \in V$, (i, j, k) is an unshielded triple, and i and j are not m-separated given S.*

Assumption 4 (Path faithfulness and Markov property). *The joint distribution P of the random vector W is (η, λ)-strong-path-faithful to the MAG G, where the sequence $\lambda = \lambda_p$ is specified later.*

The next assumption captures the local point of view underlying our algorithm and posits small partial correlations given local separators.

Assumption 5 (Local partial correlation). *Suppose Assumption 2 holds with lFCI parameters η and γ. Let $\mathcal{S}_{\eta,\gamma}(i, j)$ denote the collection of γ-local-graph separators of (i, j) with size at most η. With λ from Assumption 4, it holds that*

$$\max_{(i,j) \notin E} \min_{S \in \mathcal{S}_{\eta,\gamma}(i,j)} |\rho(i, j|S)| \leq \frac{1}{2}\lambda.$$

As we now show, this assumption holds under a directed β-walk-summability condition. This condition mirrors the walk-summable condition for undirected graphs, which holds for a large class of networks (Malioutov et al., 2006).

Assumption 6 (Directed β-summability)**.** *The joint distribution P of the random vector W belongs to a linear SEM in the form* (1.2)*, in which the weighted adjacency matrix B satisfies $\|B\| \leq \beta < 1$, where β is a fixed constant and $\|\cdot\|$ denotes the spectral norm.*

If the norm of the error covariance matrix Ξ is bounded, then directed β-summability implies Assumption 5. We state this in the following lemma, which is a slightly modified version of Lemma 2 in Sondhi and Shojaie (2019).

Lemma 11. *If Assumptions 2, 3, 4, 6 are satisfied, $\|\Xi\|$ is bounded, and γ is larger than some constant $\gamma^*(\eta, M, \lambda, \|\Xi\|)$, then Assumption 5 is also satisfied.*

To further justify Assumption 5, we conducted simulation studies (see Appendix B.5 and Section SM5 of the Supplementary Material) which show that in a number of natural settings, long paths carry weak effects while short paths carry strong effects, and Assumption 5 is likely to hold. We note also that if long paths may carry strong effects, then Assumption 5 might only be satisfied with large values of γ (e.g., $\gamma_p = p$), and local graph search strategy will not enjoy any benefits with regard to computational efficiency.

We now establish our main consistency result.

Theorem 6. *Consider a sequence of MAGs $\{G_p\}$ and distributions such that Assumptions 2, 3, 4, 5 hold with $n = \Omega((\log p)^{1/(1-2c)})$ and $\lambda = \Omega(n^{-c})$ for some $c \in (0, 1/2)$. Then there exists a sequence of significance levels $\alpha_n \to 0$ such that Algorithm 15 consistently learns a PAG for $[G_p]$ from an i.i.d. sample of size n. Moreover, if Assumption 1 holds, then the consistently learned PAG is maximally informative.*

The sample complexity of lFCI established in Theorem 6 offers considerable improvement over the worst-case sample complexity of the FCI and RFCI algorithms for graphs with unbounded size of D-SEP sets and the FCI+ algorithm for graphs with large node degrees.

As a corollary, we also improve the theory of reduced PC (Sondhi and Shojaie, 2019) for DAG learning: we derive the correctness of reduced PC and its "approximate version" under the local path condition (Definition 3). The sample complexity is also improved by applying an alternative error propagation computation; see Appendix B.1 for details.

Corollary 1. *Consider a sequence of DAGs* $\{G_p\}$ *and distributions such that Assumptions 2, 3, 4, 5 hold with* $n = \Omega((\log p)^{1/(1-2c)})$ *and* $\lambda = \Omega(n^{-c})$ *for some* $c \in (0, 1/2)$*. Then, there exists a sequence of significance levels* $\alpha_n \to 0$ *such that rPC and the approximate rPC both consistently learn the CPDAG of* G_p *from an i.i.d. sample of size* n*.*

3.5 Numerical Experiments

We explore the performance of our algorithm on three types of graphs: Erdős-Renyi, power-law, and Watts-Strogatz graphs. Since generating general random MAGs is challenging, for each family, we generate DAGs with $p = |V| \in \{100, 200, 500\}$ nodes and average node degree 2 using the `igraph` library in R. Edge weights are drawn uniformly from $(-1, -0.1] \cup [0.1, 1)$, and $n = \{100, 200, 500\}$ observations are generated using the `rmvDAG` function in the `pcalg` library (Kalisch et al., 2022). We randomly choose $q = 0.2p$ nodes as latent variables, and the rest as observed. We include no selection variables.

We run FCI, RFCI, FCI+, lFCI, and the Markov blanket version of lFCI on the observed data, with thresholds $\alpha = \{10^{-20}, 10^{-10}, 10^{-5}, 10^{-4}, 0.5 \cdot 10^{-4}, 10^{-3}, 0.5 \cdot 10^{-3}, 10^{-2}\}$. The `pcalg` library is used for the existing methods. We run lFCI with $\eta = 2$ and $\gamma = \lceil \log p \rceil$. We repeat the experiment 200 times for each α. The maximum node degrees of Erdős-Renyi graphs and Watts-Strogatz graphs ranges from 7 to 9. The maximum node degrees of power-law graphs grow with p, with medians 41, 68 and 130.

The performance of the algorithms are evaluated using precision-recall curves with respect to skeletons in Figures 3.5–3.7. In all settings, Algorithm 15 and 4 offer improvement over existing methods in terms of larger area under curves. Though they sometimes have lower precision in the beginning of the curves, they offer better trade-offs for higher recall

values. The improvement is substantial in high-dimensional cases. For power-law graphs, Algorithm 15 and 4 are superior in both low- and high-dimensional settings.

We also compare the edge orientation performances by counting the average number of different edge marks between the outputs and the true maximally informative PAG. We only show comparison in the case of $n = 200$ and $\alpha = 10^{-4}$. Differences in edge marks are shown together with the Structural Hamming Distances (SHD) between output and truth in Figure 3.8. The performance of Algorithm 15 is on par with FCI/RFCI and superior to FCI+ in both low- and high-dimensional cases. Though Algorithm 4 performs better in the skeleton steps, the orientation steps are not as reliable in high-dimensional cases. A comparison of computational cost is presented in the Supplementary Material.

3.6 Application: Gene Regulatory Network Inference

We apply the lFCI algorithm to gene expression data from The Cancer Genome Atlas (TCGA) database [1] The dataset contains the gene expression levels measured by RNAseq for 20530 genes from $n = 551$ patients with prostate cancer. The aim of this application is to infer the regulatory network among the genes.

We construct a network of known gene regulatory relations among genes measured in TCGA with at least one known interaction in BIOGRID (Stark et al., 2006); this leaves us with a graph G^{B} with $p = 2478$ nodes. The graph G^{B} includes many disjoint subgraphs, of which only 10 subgraphs (denoted $G_i^{\mathrm{B}} = (V_i^{\mathrm{B}}, E_i^{\mathrm{B}})$, $i = 1, \ldots, 10$) have more than 2 nodes; see Figure 3.9. Since BIOGRID represents gene regulatory relations in normal cells, G^{B} might not accurately capture the interactions in cancerous cells (Ideker and Krogan, 2012; Shojaie, 2021). We denote the true unobserved network in cancerous cells as G^{T} to underscore this difference.

In practice, the true graphs, $G_i^{\mathrm{T}} = (V_i, E_i^{\mathrm{T}})$, are unknown. However, the clustering of nodes in G^T are expected to be similar to those in G^B. Thus, assuming that induced

[1]The results published here are in whole or part based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

subgraphs of G^{T} over $V_1^{\mathrm{B}}, \ldots, V_{10}^{\mathrm{B}}$ are also disjoint, we use the PC algorithm to estimate the edge sets of G_i^{T}, $i = 1, \ldots, 10$. Specifically, for each V_i, we run PC with different significance levels and choose the one that maximizes the eBIC score (Foygel and Drton, 2010). We denote the resulting CPDAGs as $\widetilde{G}_i = (V_i, \widetilde{E}_i)$. The skeleton of these graphs are shown in Figure 3.10. Figures 3.9 and 3.10 demonstrates the similarities and discrepancies between G^{B} and G^{T}.

To capture the situation in which researchers only have access to data from a subset of genes, we run the following experiment on each subgraph: We randomly sample a subset of genes and infer their causal relations using PC, FCI, and lFCI. In the ℓ-th experiment, we randomly sample half of the genes from V_i as observed nodes, denoted V_i^{ℓ}, and treat the rest as unobserved. To make the problem scientifically interesting, we assign higher probability of being observed to genes with degree more than 8 in $\widetilde{G}$. The ground truth is the MAG deduced from G_i^{T} over V_i^{ℓ}. In practice, we use the MAG deduced from $\widetilde{G}_i$, and call this MAG $\widetilde{G}_i^{\ell}$. We run PC, FCI and lFCI (setting $\eta = 2$ and $\gamma = \lceil \log |V_i| \rceil$) with different threshold levels (α); for each output (PAG), we find a MAG compatible with it, compute the likelihood of the corresponding Gaussian graphical model, and compute the eBIC score with tuning parameter set to 0.5; we allow different α levels for PC, FCI and lFCI that each maximizes the eBIC score. For comparison, we also run a baseline method: we first infer an undirected graph using graphical lasso (glasso) (Friedman et al., 2008) with penalty tuned by eBIC. For this baseline, we assign edge marks (arrowhead, tail, and circle) randomly to obtain a mixed graph.

We compare the estimated graphs with $\widetilde{G}_i^{l}$. More specifically, we compare the graphs estimated using only observed variables V_i^{l}, to the "truth" deducted from the network over V_i. We use the directed Structural Hamming Distance (dSHD) to measure the difference between the estimated graph and $\widetilde{G}_i^{l}$. For directed graphs, dSHD is defined as the number of edge additions, deletions or flips to transform one graph into another. In mixed graphs, we count edges with two mismatching marks as a "flip" and edges with one mismatching mark as half a "flip". We repeat this process 40 times for each V_i, $i = 1, \ldots, 10$.

The results in Table 3.1 demonstrate that, as expected, FCI and lFCI outperforms PC. They also suggest that lFCI slightly outperforms or is comparable with FCI in 9 of the 10 subgraphs. The improvement is more pronounced in large graphs and graphs containing many "hubs" (e.g., component 1, 3 and 5), but is nonetheless persistent in all subgraphs.

3.7 Discussion

Causal structure learning from observational data is an important and challenging problem. The challenges are compounded in the presence of unmeasured confounding and selection bias. The gold-standard approach for this task, the FCI algorithm (Spirtes et al., 2001), and its relatives, RFCI (Colombo et al., 2012) and FCI+ (Claassen et al., 2013), are based on neighborhood-based search strategies that become inefficient in graphs with unbounded maximum degrees. However, such graphs are abundant in biological and physical systems. To facilitate causal structure discovery in such settings, our local FCI (lFCI) algorithm utilizes the *local separation property* of large (random) networks (Anandkumar et al., 2012b) by considering an alternative local-graph-based search strategy focused on short paths between pairs of observed nodes. This idea applies naturally to linear Gaussian structural equation models (SEMs), in which conditional independence is equivalent to zero partial correlation. However, the proposed algorithm only relies on conditional independence tests, and can be, in principle, applied to a wider collection of models in which causal relations are well-characterized by local structures. Extending this idea to more general distributions, using, e.g., Gaussian copulas (Harris and Drton, 2013), or using conditional mutual information (Anandkumar et al., 2012a) can be fruitful directions of future research.

For linear Gaussian SEMs, Assumption 5 gives a condition under which lFCI consistently learns a correct PAG. The conditional covariances involved in this assumption can be expressed as summations over products along *treks*, which are particular paths in the graph (Sullivant et al., 2010; Draisma et al., 2013); see also the discussion in SectionB.3 of the Supplementary Material. From this point of view, conditioning on the local separators proposed in Section 3.2 can be regarded as eliminating the contribution of short treks. It is then intu-

itive that the remaining long treks, each of which gives a product of many correlations, only make lower-order contributions. However, we found it difficult to formalize this argument into an assumption that weakens Assumption 6 as this would require explicitly controlling the number of long treks and their overall contribution to conditional covariances.

Algorithm 2: lFCI

Input : Tests of conditional independences $i \perp\!\!\!\perp j | S$,
maximum separating set size η, locality parameter γ.

Output: A mixed graph $\widehat{G}$

$C, C_{\text{old}} \leftarrow$ complete undirected graph over $[p]$;
Initialize SEP $\leftarrow \varnothing$ and $\ell \leftarrow -1$;
repeat
  $\ell \leftarrow \ell + 1$;
  repeat
    Select a (new) ordered pair of nodes (i, j) that are adjacent in C;
    repeat
      Choose a (new) $S \subseteq J_\gamma(i, j, C_{\text{old}})$ with $|S| = \ell$;
      if $i \perp\!\!\!\perp j | S$ **then** Delete edge (i, j) from C, and record SEP$(i, j) \leftarrow S$;
    until (i, j) is deleted or all subsets of $J_\gamma(i, j, C_{\text{old}})$ with size ℓ are checked;
  until all ordered pairs of (i, j) has been checked;
  $C_{\text{old}} \leftarrow C$;
until $\ell > \eta$;
`// At this stage the algorithm has determined the estimated skeleton` C
Convert C into a PAG $\widehat{G}$ by orienting edges using the SEP sets under application of the modified rules in Section 3.3.3;
return The mixed graph $\widehat{G}$.

Algorithm 3: lFCI$_{\text{mb}}$: lFCI with moral graph step

Input : Tests of conditional independences $i \perp\!\!\!\perp j|S$,
maximum separating set size η, locality parameter γ.

Output: A partial ancestral graph C

1 $C \leftarrow$ estimated local moral graph;

2 Run the edge removal loop in Algorithm 15 for search set size $l = 1, \ldots, \eta - 1$;

3 Orient edges in C using the SEP sets, by the modified rules in Section 3.3.3;

4 **return** A PAG C.

	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}
$\|V\|$	623	493	418	387	169	108	105	87	74	14
PC	99%	99%	101%	95%	98%	110%	96%	114%	84%	85%
FCI	96%	95%	94%	92%	90%	89%	82%	96%	76%	**80%**
lFCI	**94%**	**94%**	**92%**	**91%**	**89%**	**87%**	**80%**	**91%**	**73%**	81%

Table 3.1: Average directed Structural Hamming Distances (dSHD) between outputs and truth, as percentage of the glasso baseline.

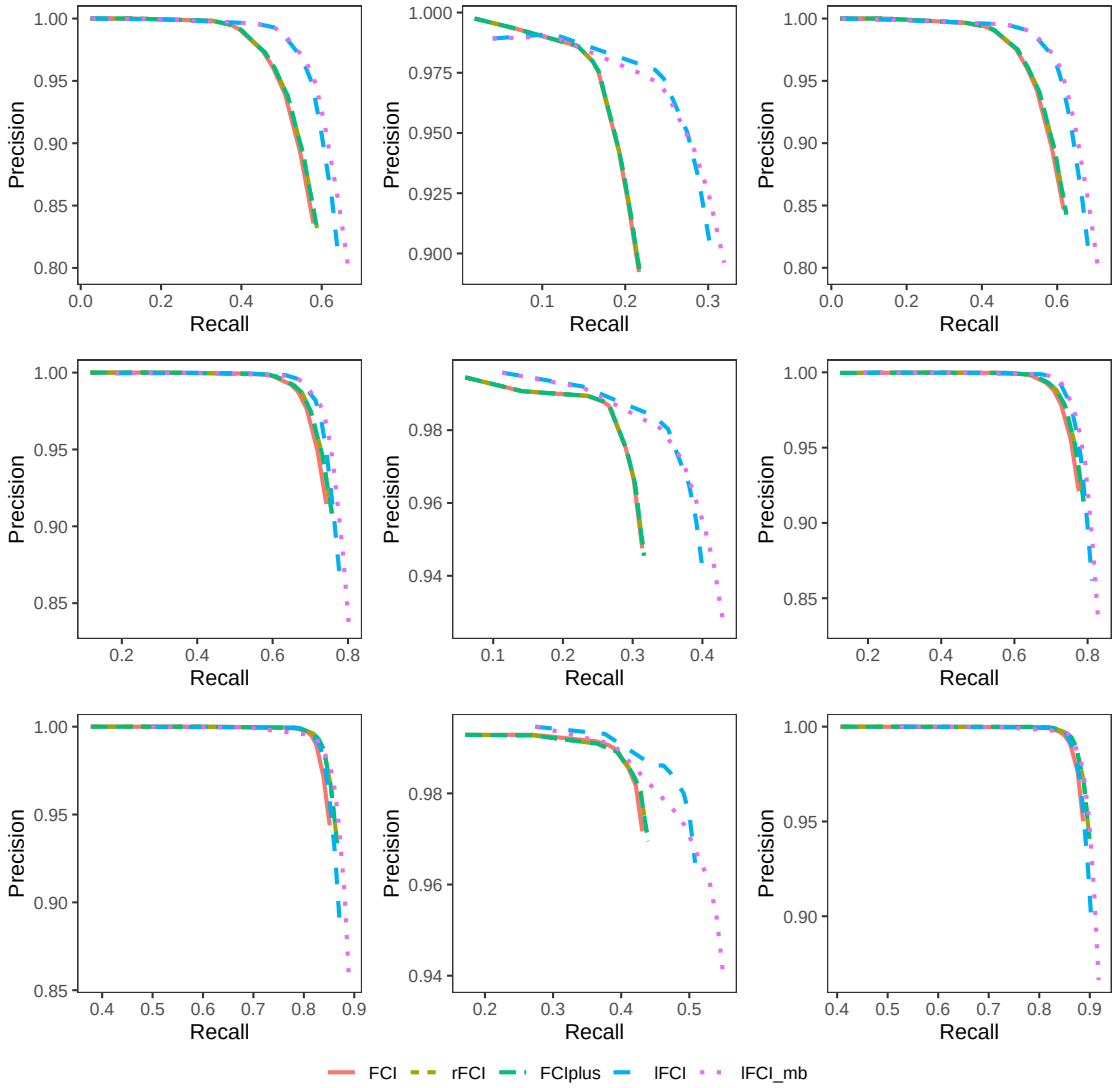


Figure 3.5: Precision-recall (PR) curves for graphs with $p = 100$ from Erdős-Renyi (left column), power-law (middle column), and Watts-Strogatz (right column) graphs based on $n = 100$ (top row), $n = 200$ (middle row), $n = 500$ (bottom row) samples.

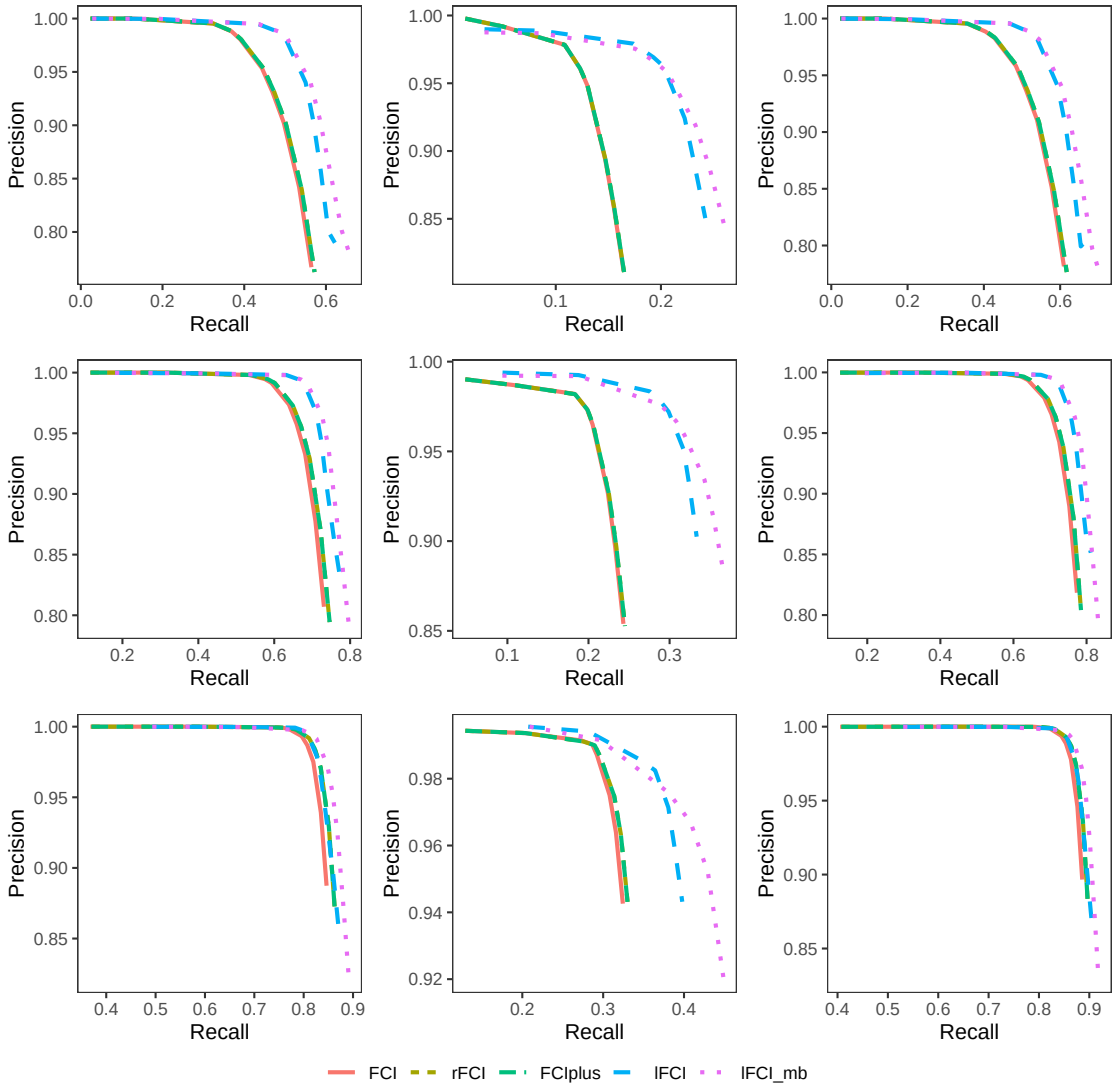


Figure 3.6: Precision-recall (PR) curves for graphs with $p = 200$ from Erdős-Renyi (left column), power-law (middle column), and Watts-Strogatz (right column) graphs based on $n = 100$ (top row), $n = 200$ (middle row), $n = 500$ (bottom row) samples.

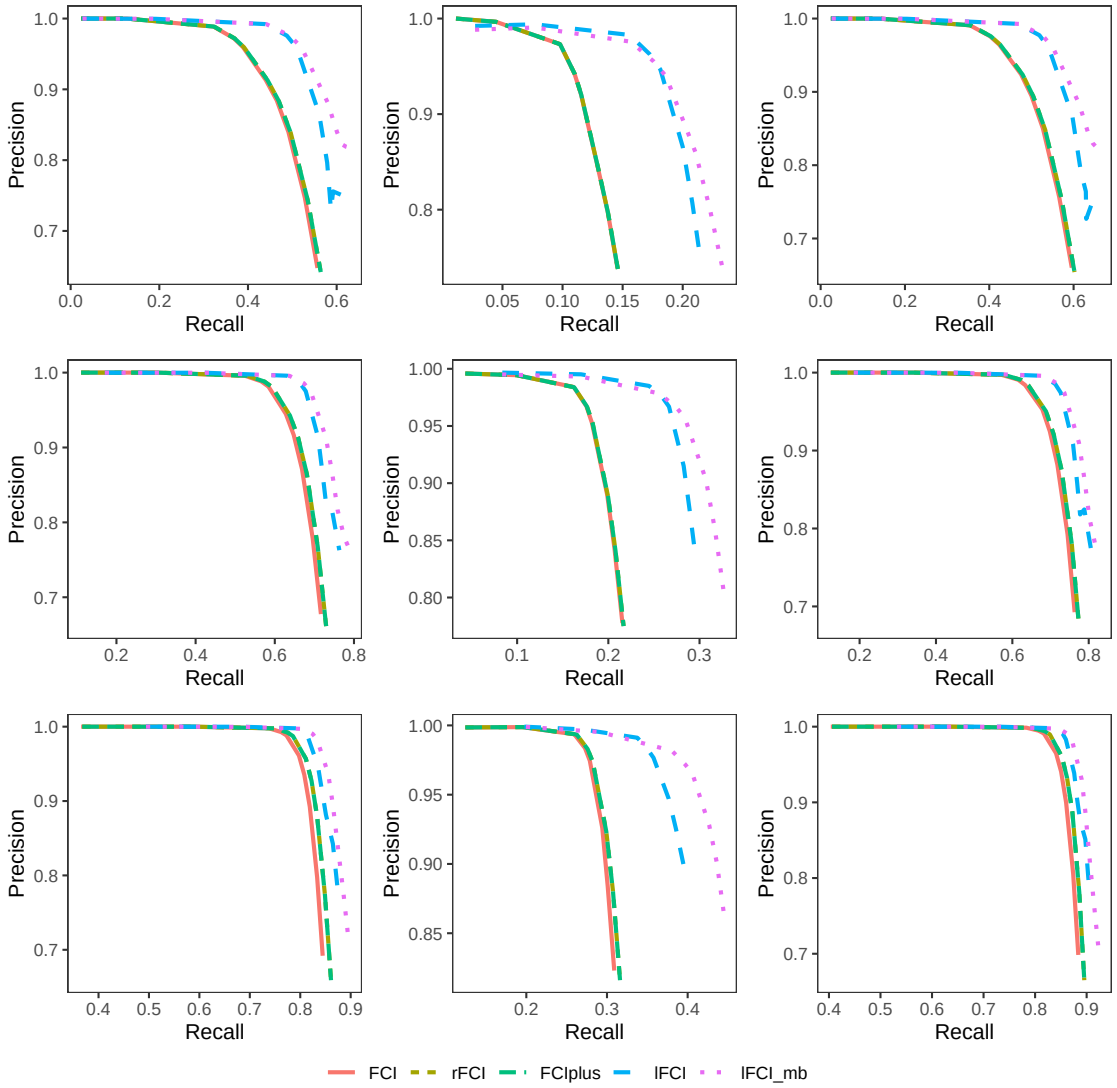


Figure 3.7: Precision-recall (PR) curves for graphs with $p = 500$ from Erdős-Renyi (left column), power-law (middle column), and Watts-Strogatz (right column) graphs based on $n = 100$ (top row), $n = 200$ (middle row), $n = 500$ (bottom row) samples.

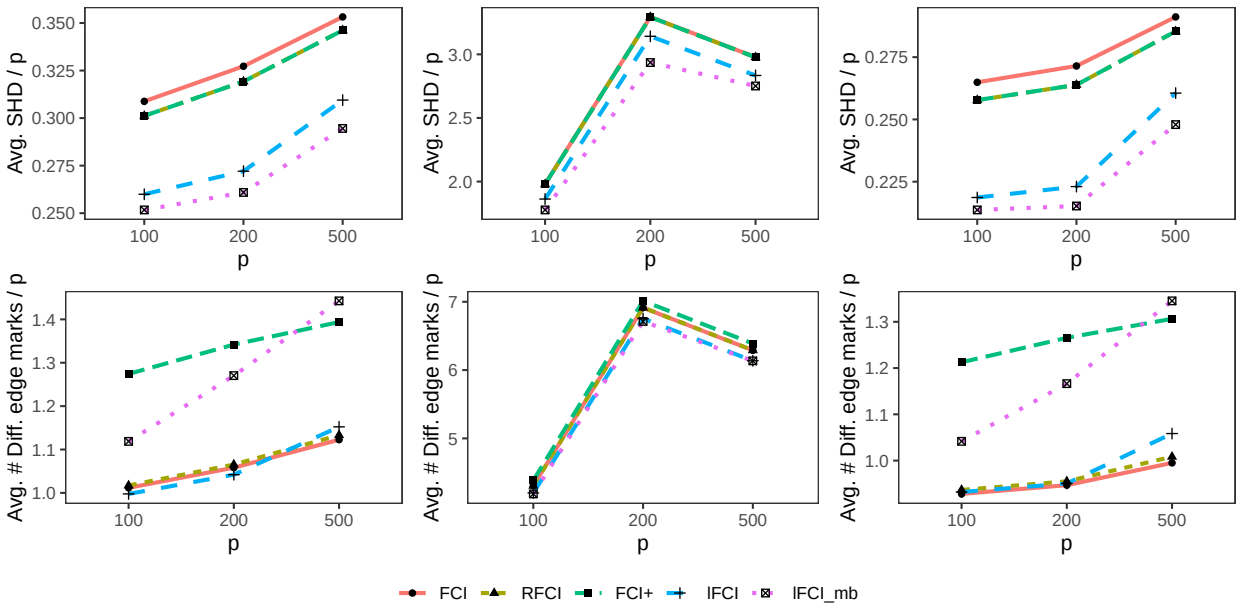


Figure 3.8: Average Structural Hamming Distances divided by p (top row) and difference in edge marks divided by p (bottom row) with $p = 100, 200, 500$ from Erdős-Renyi (left column), power-law (middle column), and Watts-Strogatz (right column) graphs based on $n = 200$.

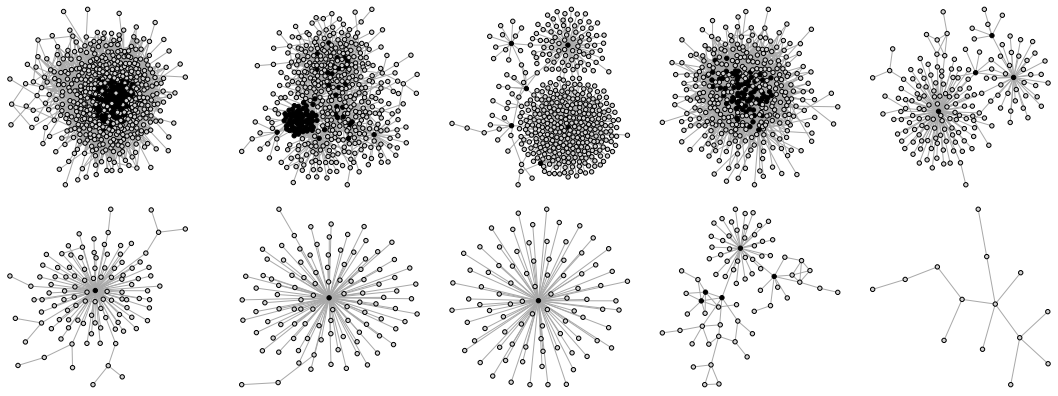


Figure 3.9: Visualization of the BIOGRID networks $G_1^{\mathrm{B}}, \ldots, G_{10}^{\mathrm{B}}$.

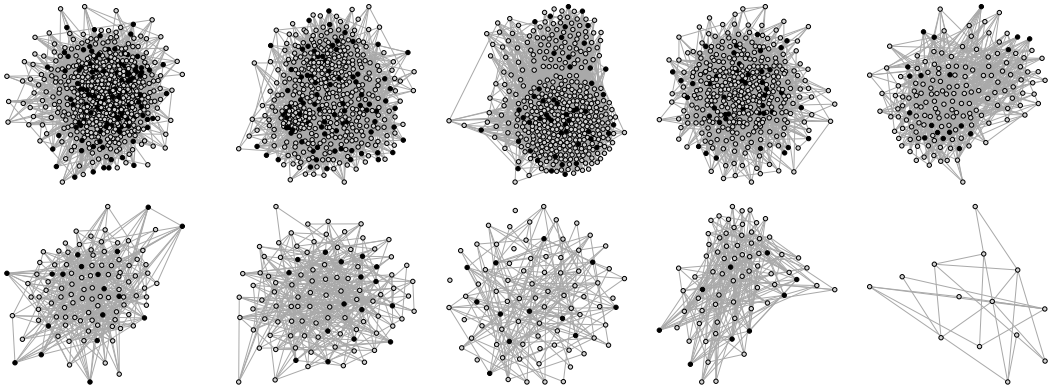


Figure 3.10: Visualization of the estimated TCGA networks $\widetilde{G}_1, \ldots, \widetilde{G}_{10}$.

Chapter 4

DEFINITE NON-ANCESTRAL RELATIONS AND STRUCTURE LEARNING

4.1 Introduction

In this work, we study *definite ancestral* and *definite non-ancestral* relations, which are ancestral and non-ancestral relations shared by all members of a Markov equivalence class. We provide graphical interpretations of these relations in the CPDAG, and we also provide a framework for reliably learning definite non-ancestral (DNA) relations without the need to recover the skeleton. These relations not only provide causal interpretations (in the form of "change in X definitely does not cause change in Y"), but also facilitate further structure learning. Indeed, we show that learning DNA relations directly can greatly improve the statistical and computational efficiency of existing structure learning algorithms.

In most existing structure learning methods, edge orientations and ancestral relations can only be deduced after learning the skeleton, and hence the correctness of direct and pathway causal relations depends on correct recovery of skeleton, which is often not reliable with small number of observations. In this work we propose a method that reliably learns the definite non-ancestral relations without the need to recover the skeleton. These relations not only provides causal interpretations (in the form of "change in A definitely does not cause change in B"), but also facilitates further structure learning. We also propose a structure learning framework that first discovers definite non-ancestral relations and then use them to facilitate learning the DAG.

Existing structure learning methods can be broadly categorized into constraint-based, score-based, and hybrid approaches. In this paper, we will focus on two prominent examples of learning algorithms: The PC algorithm Spirtes et al. (2001) and the Sparsest Permutation

algorithm (Raskutti and Uhler, 2018; Solus et al., 2021). The PC algorithm is the default constraint-based method and hierarchically performs tests of conditional independence with conditioning sets of increasing size. Under a *faithfulness assumption*, the population version of PC algorithm outputs the correct CPDAG Spirtes et al. (2001). At the population level (i.e., with "infinite data"), the faithfulness assumption merely requires that the conditional independences in the data-generating distributions coincide (to sufficient order) with d-separation relations in the DAG. However, good performance of the sample version is only guaranteed when the assumption is strengthened to bound signals of conditional dependence away from zero, which is far more restrictive (Kalisch and Bühlmann, 2007; Uhler et al., 2013). On the other hand, the Sparsest Permutation algorithm (Raskutti and Uhler, 2018; Solus et al., 2021) is a hybrid learning method that searches among all topological orderings. For each ordering, a DAG is inferred via conditional independence tests given the ordering, and the number of edges in the DAG is used as the score. The algorithm looks for the sparsest DAG under which the data-generating distribution is Markov. The SP algorithm relies on weaker distributional assumptions than PC, but comes at increased computational cost due to its score-based searching scheme.

As noted above, we propose here an algorithm to directly learn DNA relations. This algorithm is constraint-based and derives the relations from conditional independences/dependences. The learning algorithm is based on two well-known rules of conditional independence that has been studied extensively in Entner et al. (2013); Claassen and Heskes (2011); Claassen et al. (2013); Magliacane et al. (2016). We will then show that the learned DNA relations provide *order-constraining* information. In other words, they define a subset of all possible topological orderings (or DAGs) that is guaranteed to contain one correct ordering (or DAG). Therefore, DNA relations can be used to reduce the number of conditional independence tests needed for running the PC algorithm. Similarly, the ordering information provided by the DNA relations can also help significantly reduce the search space of the SP algorithm, which grows exponentially with graph size. Regarding this last point, we would like to highlight an independent preprint Squires et al. (2020) that uses information from

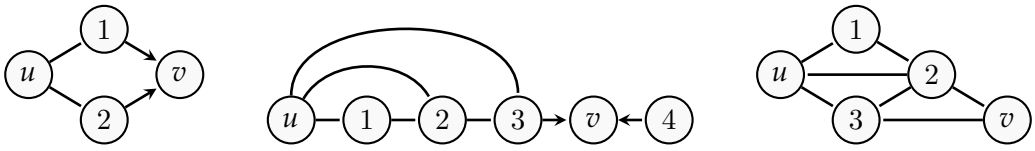


Figure 4.1: In the graph on the left, $A_{uv} = \{1, 2\}$ is not a clique, so u must have an arrowhead to one of them, and thus $u \rightsquigarrow v$ as suggested by Lemma 12. In the middle graph, $A_{uv} = \{3\}$, which is trivially a clique, u is not definitely ancestral to v. In the last graph, $A_{uv} = \{2, 3\}$, which forms a clique, and it is possible to orient both 2 and 3 into u and make $u \notin \text{an}(v)$, and therefore u is not definite ancestral to v according to Lemma 12.

the moral graph to reduce the search space of SP. Compared with that work, our framework of DNA relations is more general and can accommodate weaker assumptions.

4.2 Definite Non-Ancestral Relations

A DAG G is usually not uniquely identifiable from the distribution of observational data. Instead, there may be other graphs G' that entail the exact same d-separation relations. Together, these graphs form the *Markov equivalence class*, $[G]$. Importantly, the graphs in $[G]$ share their adjacencies and possibly also some of their edge marks. In other words, the equivalent graphs $G' \in [G]$ may differ only through reversals of directed edges and some of the directed edges may in fact be oriented the same way in all members of $[G]$; these latter set of edges can be unambiguously interpreted as direct causal effects. The characterization and discovery of such edges is well-studied (Andersson et al., 1997; Meek, 1995).

Throughout this paper, we will use the adjective 'definite' to highlight the structure common to all members of a Markov equivalence class, $[G]$. With this terminology, the edges that can be oriented the same way in all members of $[G]$ are definite (as opposed to incidental) arrows. We similarly define definite ancestral and definite non-ancestral relations with regard to causal pathways.

Definition 7 (Definite Ancestral and Definite Non-Ancestral Relations)**.** *Let u, v be two*

nodes in a DAG G. Then u is definite ancestral to v, denoted $u \rightsquigarrow v$, if $u \in \mathrm{an}(G', v)$ for all $G' \in [G]$. Similarly, u is definite non-ancestral to v, or $u \not\rightsquigarrow v$, if $u \notin \mathrm{an}(G', v)$ for all $G' \in [G]$. When writing $u \rightsquigarrow W$ for a set of nodes W, we mean $u \in \mathrm{an}(G', W)$ for all $G' \in [G]$; when writing $u \not\rightsquigarrow W$, we mean $u \not\rightsquigarrow w$ for all $w \in W$. Finally, we define $D^{\rightsquigarrow}(G) = \{(u, v) : u \rightsquigarrow v \text{ in } G\}$ and $D^{\not\rightsquigarrow}(G) = \{(u, v) : u \not\rightsquigarrow v \text{ in } G\}$.

We emphasize that the two notions are not complementary to each other. While the acyclicity of the graph entails that $u \rightsquigarrow v$ implies $v \not\rightsquigarrow u$, the converse need not be true.

In the rest of this section, we present two perspectives of definite ancestral and non-ancestral relations: the CPDAG perspective and the d-separation perspective. The former provides a set of rules to read the (non-)ancestral relations off a CPDAG, and the latter enables efficient learning without knowing the CPDAG. This second perspective provides a foundation for learning DNA directly from a probability distribution, or data drawn from it.

4.2.1 Ancestral and non-ancestral relations from CPDAGs

Let G be any DAG. The Markov equivalence class $[G]$ can be represented using the completed partially directed acyclic graph (CPDAG) G^*. The CPDAG is a mixed graph containing directed and undirected edges that has the same skeleton as G and whose edges are directed if and only if they have the same orientation in all members of the Markov equivalence class. The CPDAG representation is complete in the sense that all directed edges are definite arrows, and for each undirected edge $u - v$, there exists two DAGs in $[G]$ that contain $u \to v$ and $u \leftarrow v$, respectively.

From the CPDAG perspective, definite ancestral and non-ancestral relations can be identified via *possibly directed paths*, which are paths between two nodes with no arrow into the initial node. We say a possibly directed path is *unshielded* if no three consecutive nodes on the path form a triangle. Notably, it has been shown that if there exists a possibly directed path from u to v in the CPDAG, then some subsequence of this path forms an unshielded possibly directed path (Zhang, 2008; Perkovic et al., 2018), We now give a comprehensive

characterization of definite ancestral and definite non-ancestral relations in terms of the CPDAG; all proofs are given in the supplement.

Lemma 12. *Let u and v be two nodes in a CPDAG G^*.*

- *Let A_{uv} be the collection of all nodes that lie immediately after u on some unshielded possibly directed path to v. Then $u \rightsquigarrow v$ if and only if either u has a definite arrow into A_{uv} or A_{uv} is nor a clique nor a singleton.*

- *$u \not\rightsquigarrow v$ if and only if there is no possibly directed path from u to v in G^*.*

Figure 4.1 shows examples of definite ancestral relations.

4.2.2 Ancestral and non-ancestral relations from d-separations

Definite ancestral relations are easy to interpret, but somewhat delicate to read off the CPDAG. In contrast, definite non-ancestral relations have more subtle interpretations but are easier to read off the CPDAG; they are also simpler to learn from d-separation relations. Next we present well known rules about ancestral effects. (See, e.g., (Entner et al., 2013; Claassen and Heskes, 2011; Claassen et al., 2013; Magliacane et al., 2016).)

Lemma 13 (DA/DNA via d-separation)**.** *Let $G = (V, E)$ be a DAG. Let u, v, x be distinct vertices, and let $W \subseteq V \setminus \{u, v, x\}$. Then the following holds:*

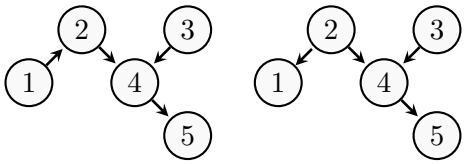
- *If u, v are d-connected given W but d-separated given $W \cup x$, then $x \rightsquigarrow W \cup \{u, v\}$.*

- *If u, v are d-separated given W but d-connected given $W \cup x$, then $x \not\rightsquigarrow u$, $x \not\rightsquigarrow v$ and $x \not\rightsquigarrow W$.*

- *If u, v are d-separated marginally (i.e., given $\emptyset$), then $u \not\rightsquigarrow v$ and $v \not\rightsquigarrow u$.*

The claims of Lemma 13 are illustrated in Figure 4.2. The following result is a special case of the second statement[1]:

[1]This special case is discussed in Squires et al. (2020).



definite ancestral	$2 \rightsquigarrow 4,\ 3 \rightsquigarrow 4,\ 4 \rightsquigarrow 5$
definite non-ancestral	$1 \not\rightsquigarrow 3,\ 3 \not\rightsquigarrow 1,\ 2 \not\rightsquigarrow 3,\ 3 \not\rightsquigarrow 2,\ 4 \not\rightsquigarrow \{1,2,3\},\ 5 \not\rightsquigarrow \{1,2,3\},\ 5 \not\rightsquigarrow 4$
layering	$(3, \{1,2\}, \{4,5\})$ or $(\{1,2\}, 3, \{4,5\})$

Figure 4.2: A Markov equivalence class comprising two DAGs and its definite ancestral/non-ancestral relations. All DNA except for $5 \not\rightsquigarrow 4$ can be discovered by Lemma 13.

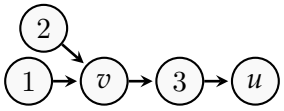


Figure 4.3: In this example, $u \not\rightsquigarrow v$ but cannot be read from the DAG using Lemma 13.

Corollary 2. *If u, v are d-separated given $V \setminus \{u, v, x\}$ and are d-connected given $V \setminus \{u, v\}$, then x is a sink, i.e., $x \not\rightsquigarrow V \setminus \{x\}$.*

Note also that the characterization of definite (non-)ancestral relations is not complete, meaning there exist such relations that do not feature the configurations stated in Lemma 13; see Figure 4.3.

In general, Lemma 13 provides a way to identify DNA between two nodes, i.e., in the form of $u \not\rightsquigarrow v$. However, we cannot identify definite ancestral relations between two nodes (see Figure 4.4). For this reason, we only discuss learning and applications of DNA in Section 4.3 and 4.4.

4.3 Learning DNA relations

In this section, we discuss how to learn DNA relations from observational data. Let $\Omega_0(\mathbb{P}) = \{(u, v, S) : u \perp\!\!\!\perp v | S \text{ in } \mathbb{P}\}$ be the collection of all conditional independences, in the form of

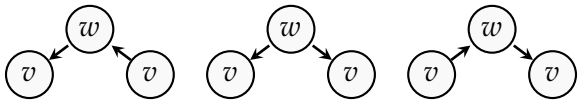


Figure 4.4: Three Markov equivalent DAGs, in all of which nodes u and v are marginally d-connected and d-separated given w; thus $w \rightsquigarrow \{u, v\}$. However, this does not imply $w \rightsquigarrow u$ or $w \rightsquigarrow v$.

tuples, that hold in distribution $\mathbb{P}$. We can express the Markov property as follows.

Definition 8 (Markov Property)**.** *A distribution $\mathbb{P}$ is Markov with respect to a DAG G if*

$$u, v \text{ are d-separated by } S \text{ in } G \implies (u, v, S) \in \Omega_0(\mathbb{P}).$$

The reverse implication is known as the faithfulness condition. If $\mathbb{P}$ is Markov and strong-faithful to G, then the Markov equivalence class, $[G]$, can be recovered exactly from $\mathbb{P}$, or from observations obtained from $\mathbb{P}$. Unfortunately, strong faithfulness is a very restrictive distributional assumption that is often violated by generic graphical models (Uhler et al., 2013). Constraint based structure learning algorithms often rely on weaker conditions to recover the equivalence class. See Section 4.4.2 and Section 4.4.3 for detailed discussion.

Lemma 13 states that DNA relations can be identified from d-separation and a d-connection relations: DNA relations can be correctly learned from the distribution if d-separation and d-connection relations correspond exactly to conditional independence and conditional dependence. We formalize this condition as DNA-faithfulness.

Definition 9 (DNA-faithfulness)**.** *Let Ω and $\bar{\Omega}$ be two collection of triples consisting of two vertices and one set of nodes from a DAG G. We say that a joint distribution $\mathbb{P}$ is DNA-faithful to G with respect to $(\Omega, \bar{\Omega})$ if $\Omega \subseteq \Omega_0(\mathbb{P})$, $\bar{\Omega} \cap \Omega_0(\mathbb{P}) = \emptyset$, and it holds for any three nodes u, v, z and set $S \subseteq V \setminus \{u, v, z\}$ that*

$$(u, v, S) \in \Omega \text{ and } (u, v, S \cup z) \in \bar{\Omega} \implies u, v \text{ are d-separated by } S \text{ in } G. \tag{4.1}$$

We can learn DNA relations by a two-step procedure that first runs an algorithm to learn d-separation relations, and then performs additional tests to identify d-connections and learn DNA by Lemma 13. Let $\mathcal{A}$ be an arbitrary constraint-based structure learning algorithm that tests and collects a set of conditional independence statements $\Omega_{\mathcal{A}}(\mathbb{P})$ that hold in a distribution $\mathbb{P}$. Then we perform additional tests to detect conditional dependences in $\mathbb{P}$, i.e., we form

$$\bar{\Omega}_{\mathcal{A}}(\mathbb{P}) = \{(u, v, S \cup z) : u \not\perp\!\!\!\perp v | S \cup z, (u, v, S) \in \Omega_{\mathcal{A}}(\mathbb{P})\}.$$

We summarize this framework to learn DNA in Algorithm 4.

Algorithm 4: General DNA-learning framework

Input : An arbitrary constraint based algorithm $\mathcal{A}$

Output: A set of DNA $D \subseteq D^{\not\rightsquigarrow}(G)$

```
D ← ∅;
Run A, record the conditional independences discovered by A as Ω_A;
for (x, y, S) ∈ Ω_A do
    for z ∈ V \ {x, y, S} do
        if x ⫫̸ y | S ∪ z then
            Record z ↛ x, z ↛ y and z ↛ S in D
return D.
```

Theorem 7 (Correctness of DNA Learning)**.** *Let $\mathbb{P}$ be a distribution Markov to a DAG G, and let $\mathcal{A}$ be a constraint-based learning algorithm such that $\mathbb{P}$ is DNA-faithful to G with respect to $(\Omega_{\mathcal{A}}(\mathbb{P}), \bar{\Omega}_{\mathcal{A}}(\mathbb{P}))$. Then, the output of Algorithm 4 is a set of true DNA relations in G.*

4.3.1 Learning DNA from small d-sep sets

In the previous section we discussed DNA learning as a two-step procedure: first look for d-separations and then look for d-connections. A classic approach for discovering d-separation relations systematically is to hierarchically test for conditional independences given sets of increasing size, $0, 1, 2, \ldots$. In particular, this is the idea behind the PC algorithm. Adopting this strategy here, we can use the first few rounds of PC to learn d-separations. In other words, we use the PC algorithm with early stopping as the constraint-based procedure $\mathcal{A}$ in Algorithm 4. The procedure is summarized in Algorithm 5.

Algorithm 5: DNA-learning via small conditioning sets

Input : A conditional independence test, a level K

Output: A set of DNA relations $D \subseteq D^{\not\leftrightarrow}(G)$

1 Repeat steps 2-5 of Algorithm 4 using PC with conditioning sets of size $0, \ldots, K$.

2 **return** D.

The rationale behind stopping the PC early when learning DNA is that the learning procedure $\mathcal{A}$ in Algorithm 4 only needs to find correct d-separations. However, false positive edges in the output of $\mathcal{A}$ are allowed. In fact, our empirical results in the supplement (Section 4.5.1) show that in most cases a large number of DNA relations can be learned from considering only the first two steps of the PC algorithm, i.e., from setting $K = 0$ or 1. We note that according to Definition 9 and Theorem 7, restricting the size of the conditioning sets also leads to less restrictive faithfulness assumption for correct DNA learning. But this also reduce the potential amount of DNA that can be learned.

With a view towards practical algorithms, we now focus on data from a multivariate Gaussian distribution $\mathbb{P}$ and adopt a threshold λ to specify signal strengths. Let $\rho(u, v|S)$ be the partial correlation obtained from the conditional distribution for the pair (u, v) given

S. Define

$$\Omega_\lambda^K(\mathbb{P}) := \{(u, v, S) : |S| = K \text{ and } |\rho(u, v|S)| \leq \lambda\},$$
$$\bar{\Omega}_\lambda^K(\mathbb{P}) := \{(u, v, S) : |S| = K \text{ and } |\rho(u, v|S)| > \lambda\}.$$

Furthermore, let

$$\Omega_\lambda^{\uparrow K}(\mathbb{P}) := \bigcup_{k=0}^{K} \Omega_\lambda^k(\mathbb{P})$$

be the triples obtained from conditioning sets of size at most K and strength threshold λ. The next result guarantees the correctness of Algorithm 5.

Theorem 8 (Correctness of DNA learning from small conditioning sets)**.** *Let G be a DAG, and let $\mathbb{P}$ be a distribution that is Markov to G and DNA-faithful to G with respect to $\left(\Omega_\lambda^{\uparrow K}(\mathbb{P}), \bar{\Omega}_\lambda^{K+1}(\mathbb{P})\right)$. Then Algorithm 5 with level K returns a correct set of DNA relations.*

For all choices of K, the DNA-faithfulness assumption with respect to $(\Omega_\lambda^{\uparrow K}(\mathbb{P}), \bar{\Omega}_\lambda^{K+1}(\mathbb{P}))$ is weaker than the λ-strong faithfulness assumption for the PC algorithm; see Uhler et al. (2013). However, it cannot be directly compared with the adjacency and orientation faithfulness, which are the condition for completeness of PC (Ramsey et al., 2006). Nevertheless, in the special case of $K = 0$, the DNA-faithfulness condition is mild: If $\mathbb{P}$ is Gaussian, then it is sufficient if the correlation of every pair of marginally d-connected nodes is bounded away from zero. Our numerical result in Section 4.5 also confirms that DNA-faithfulness is generally not stronger than other notions of faithfulness.

Given Theorem 8, we next provide sample guarantees for linear structural equation models (SEMs) with sub-Gaussian errors. In other words, we assume that the considered joint distribution $\mathbb{P}$ is that of a random vector X that satisfies

$$X = B^T X + \varepsilon,$$

where $B = (\beta_{uv})$ has entry $\beta_{uv} \neq 0$ only if $u \to v$ is an edge in the considered DAG. The vector ε is comprised of independent random variables. In the following theorem we

assume the distribution of the random vectors to be sub-Gaussian. Given data we may form sample partial correlation $\hat{\rho}(u, v|S)$ and implement Algorithm 5 by rejecting a hypothesis of conditional independence when $\hat{\rho}(u, v|S) > \lambda$. The following result covers both low- and high- dimensional problems.

Theorem 9 (Sample guarantee for DNA learning from small conditioning sets)**.** *Suppose data is generated as n independent draws from the joint distribution $\mathbb{P}$ of a random vector $X \in \mathbb{R}^p$ that follows a linear SEM. Let Σ be the covariance matrix and assume that each $X_i/\Sigma_{ii}^{1/2}$ is sub-Gaussian with parameter σ. Assume $\mathbb{P}$ is Markov with respect to a DAG G, and also DNA-faithful to G with respect to $(\Omega_\lambda^{\uparrow K}(\mathbb{P}), \bar{\Omega}_\lambda^{K+1}(\mathbb{P}))$. Assume the minimal eigenvalues of all $(K+3)\times(K+3)$ submatrices of Σ are bounded below by $M > 0$. For any $\zeta > 0$, if the sample size satisfies*

$$n \geq \left\{\log(p^2 + p) - \log(\zeta/2)\right\} 128(1 + 4\sigma^2)^2 \max_i(\Sigma_{ii})^2 (K+3)^2 \left(\frac{M + 1 + 2/\lambda}{M^2}\right)^2, \qquad (4.2)$$

then the output of Algorithm 5 with level K and tests based on sample partial correlations is correct with probability at least $1 - \zeta$.

4.3.2 Learning DNA from large d-sep sets

The d-separation relations for DNA can also be learned tractably by testing conditional independences given large conditioning sets of size $p-2, p-3, \ldots$, where p denotes again the number of considered variables. In fact, by Corollary 2, if x, y are d-separated given $V \setminus \{x, y, u\}$ and are d-connected given $V \setminus \{x, y\}$, then $u \not\rightsquigarrow V \setminus u$. This means that we can learn DNA relations from the moral graph, which encodes the d-separation relations that hold between each pair of nodes when conditioning on all other nodes. A special case of this approach was implemented as a recursive algorithm in (Squires et al., 2020). The theorem below establishes the correctness of the general strategy for learning DNA relations.

Given a joint distribution $\mathbb{P}$, define

$$\Omega_\lambda^{\downarrow K}(\mathbb{P}) = \bigcup_{k=p-K}^{p} \Omega_\lambda^k(\mathbb{P}).$$

Algorithm 6: DNA-learning via large conditioning sets

Input : A conditional independence test, a level K

Output: A set of DNA relations $D \subseteq D^{\not\rightsquigarrow}(G)$

Initialize $\widetilde{V} = V$;
repeat
  $\widetilde{M} \leftarrow$ moral graph over $\widetilde{V}$;
  for $u \in \widetilde{V}$ **do**
    $M_u \leftarrow$ moral graph over $\widetilde{V} \setminus u$;
    $\widetilde{M}_u \leftarrow$ subgraph of $\widetilde{M}$ over $\widetilde{V} \setminus u$;
    if $\widetilde{M}_u$ contains more edges than M_u **then**
      Record $u \not\rightsquigarrow \widetilde{V} \setminus u$;
      Update $\widetilde{V} \leftarrow \widetilde{V} \setminus \{u\}$; Break;
until $|\widetilde{V}| = p - K$;
return D.

Theorem 10 (Correctness of DNA learning from large conditioning sets)**.** *Let G be a DAG, and let $\mathbb{P}$ be a distribution that is Markov to G and DNA-faithful to G with respect to $\left(\Omega_\lambda^{\downarrow K-1}(\mathbb{P}), \bar{\Omega}_\lambda^K(\mathbb{P})\right)$. Then Algorithm 6 with level K returns a correct set of DNA relations.*

Next, we establish sample consistency of the algorithm for linear SEMs with sub-Gaussian errors.

Theorem 11 (Sample Guarantee of DNA Learning from large conditioning sets)**.** *Consider the setup of Theorem 9, but assuming DNA-faithfulness with respect to $\left(\Omega_\lambda^{\downarrow K-1}(\mathbb{P}), \bar{\Omega}_\lambda^K(\mathbb{P})\right)$. Let*

$$\lambda^* = \min_{A \subseteq [p], |A| \geq p-K, i,j \in A, [(\Sigma[A])^{-1}]_{ij} \neq 0} \left|[(\Sigma[A])^{-1}]_{ij}\right|,$$

where $\Sigma[A]$ denotes the $A \times A$ sub-matrix of Σ. Denote the sample covariance matrix as S_n.

(i) ***Low-dimensional case****. Assume the minimal eigenvalue of Σ is bounded below by*

$M > 0$. *If the sample size satisfies*

$$n \geq \{\log(p^2 + p) - \log(\zeta/2)\}128(1 + 4\sigma^2)^2 \max_i(\Sigma_{ii})^2 p^2 \left(\frac{M + 2/\lambda^*}{M^2}\right)^2, \tag{4.3}$$

then the output of Algorithm 6 at level K and with conditional independence tests that hard-threshold the inverses of submatrices of S_n with respect to $\lambda^*/2$ is correct with high probability.

(ii) ***High-dimensional case.*** *Suppose there exists a sequence of λ_n satisfying $\lambda_n \leq \lambda^* \|\Sigma^{-1}\|_1^{-1}/4$ such that $\lambda_n \geq \|\Sigma^{-1}\|_1 \|\Sigma - S_n\|_\infty$ with high probability as $n, p \to \infty$. Then the output of Algorithm 6 at level K using CLIME (Cai et al., 2011) with tuning parameter λ_n for conditional independence tests is correct with high probability.*

4.4 DNA Applications

4.4.1 Ordering constraints and Layering

A topological ordering of a DAG G, denoted π, is a total ordering of the vertices such that $\pi(u) < \pi(v)$ for each edge $u \to v$ in G. Due to acyclicity, there always exists at least one such ordering, though the ordering might not be unique. Note that a valid ordering for G need not be valid for other DAGs in $[G]$.

Without prior knowledge, learning DAGs from orderings requires checking all $p!$ possible orderings. However, DNA relations constrain the set of possible orderings. More specifically, we will show in Lemma 14 that if $u \not\rightsquigarrow v$, then there exists a valid topological ordering of G with $\pi(u) > \pi(v)$.

In general, we say an ordering π is compatible with a DNA set D if $\pi(u) > \pi(v)$ for each $u \not\rightsquigarrow v$ in D. We say D is an *order-constraining DNA set* if D contains no DNA statement cycles, i.e., we cannot follow a sequence of DNA statements in D, such as $u \not\rightsquigarrow v, v \not\rightsquigarrow w, \ldots$ and get back to u. Formally we define *ordering-constraints* as following:

Definition 10 (Ordering-constraint DNA set)**.** *A set $D \subseteq D^{\not\rightsquigarrow}(G)$ is a ordering-constraint DNA set for G if there does not exists a sequence $u, v, \ldots, w, u$ such that $u \not\rightsquigarrow v, \ldots, w \not\rightsquigarrow u$ in D.*

Given an arbitrary DNA set D, we can obtain an order-constraining subset by removing statements until there is no cycle. Specifically, if D is output of Algorithm 4, then we only need to remove one from each pair of $(u \not\rightsquigarrow v, v \not\rightsquigarrow u)$. If D is an order-constraining DNA set, then there must be some topological ordering that is compatible with both D and some member of the Markov equivalence class.

Lemma 14 (Ordering constraints)**.** *Let G be a DAG. If $u \not\rightsquigarrow v$ in G, then there exists a topological ordering π that is compatible with some $G' \in [G]$ and satisfies $\pi(u) > \pi(v)$. If D is an order-constraining DNA set for G, then there exists a topological ordering π that is compatible with some $G' \in [G]$ and satisfies $\pi(u) > \pi(v)$ for all $(u, v) \in D$.*

Order-constraining DNA sets reduce the set of possible topological orderings to be considered in structure learning to a subset that is guaranteed to contain at least one correct ordering. Therefore, this reduced set can be adopted in score-based structure learning methods such as Greedy Equivalence Search (GES) and Sparsest Permutation (SP) to trim down their search space of topological orderings and DAGs.

Moreover, order-constraining DNA sets yield *layerings* of DAGs. A *layering* of a DAG is an ordered partition of the vertex set into layers, which must be such that there is no arrow pointing from one layer to a preceding layer (Manzour et al., 2021). The finest layering we may hope to infer from conditional independence tests is the one given by the chain components of the CPDAG Andersson and Perlman (2006). Efficient algorithms have been developed to learn graph structures based on layering information, but they require knowing the layers a priori (Manzour et al., 2021). In addition to reducing the number of possible orderings, the layering of a DAG can also be used to develop efficient algorithms for learning DAGs. We present such an approach in the next section, but first show that layerings can be learned correctly from DNA.

Theorem 12 (DAG Layering via DNA)**.** *Let G be a DAG and $D \subseteq D^{\not\rightsquigarrow}(G)$. The output of Algorithm 7 is a valid layering of G.*

Algorithm 7: Learning layering from DNA

Input : DNA set $D \subseteq D^{\not\leftrightarrow}(G)$.

Output: DAG layering L.

Sources $\leftarrow$ Sinks $\leftarrow \emptyset$; $V' \leftarrow V$;

repeat

  Find the smallest subset $S \subseteq V'$ such that $(u, s) \in D$ for all $u \in V' \setminus S$, $s \in S$, or $(s, v) \in D$ for all $v \in V' \setminus S$,$s \in S$;

  In the former case, Sources $\leftarrow$ [Sources, S]; in the latter case, Sinks $\leftarrow$ [S, Sinks];

  $V' \leftarrow V' \setminus S$;

until $V' = \emptyset$;

return L = [Sources, Sinks].

4.4.2 Sparsest Permutation with DNA

Let G be a DAG and let π be an arbitrary ordering of its vertices. We define $G^\pi = (V, E^\pi)$ as the DAG deduced via the following rule:

$$(\pi(i), \pi(j)) \in E^\pi \text{ iff } i < j \text{ and}$$
$$\pi(i),\ \pi(j) \text{ not d-separated by} \{\pi(1), \ldots, \pi(j-1)\} \setminus \pi(i) \text{ in } G. \quad (4.4)$$

Clearly, $G^\pi = G$ when π is a valid topological ordering of G. We write $G^\pi(\Omega)$ if we replace the d-separations with conditional independences in Ω.

The Sparsest Permutation (SP) algorithm is a hybrid structure learning method: it searches through the space of all orderings to minimize the edge count (which plays the role of a score) of DAGs induced by the orderings via the constraint-based rule (4.4); that is, it finds

$$\pi^* = \arg\min_\pi |G^\pi(\Omega)|.$$

Under the *Sparsest Markov Representation condition* (SMR) — which requires that for any G' that is Markov to $\mathbb{P}$, either $|G'| > |G|$ or G' is Markov equivalent to G — SP recovers

the correct Markov equivalence class of G (Raskutti and Uhler, 2018; Solus et al., 2021). Since SMR is a necessary condition for restricted-faithfulness (Raskutti and Uhler, 2018), the correctness of the SP algorithm relies on weaker assumptions than that of PC.

SP can be implemented as a greedy algorithm because starting from any arbitrary ordering, there always exists a non-increasing (in number of edges) sequence of DAGs that ends up in the correct Markov equivalence class (Solus et al., 2021). One particularly efficient greedy approach is the Triangle SP (TSP), which moves from one ordering to the other by reverting *covered arrows*, that is, edges in the form of $u \to v$ satisfying $\text{pa}(v) = \text{pa}(v) \cup \{u\}$. By repeatedly looking for covered arrow reversals that induce sparser DAGs, TSP will recover a DAG in the target Markov equivalence class (Solus et al., 2021).

However, greedy search can be computationally burdensome. With an arbitrary initialization of the ordering, it may need to traverse a long non-increasing sequence of DAGs to reach the target. Moreover, since all members of a Markov equivalence class are connected via non-increasing sequences, greedy search may spend steps exploring the same equivalent class before moving on to a sparser one.[2]

From this perspective, incorporating DNA information can be very useful. The two problem discussed above both can be alleviated by incorporating DNA information. On the one hand, an order-constraining DNA set reduces the search space and provides better initial orderings; on the other hand, the layering information learned by Algorithm 7 breaks down the learning problem into smaller sized problems that are easier to tackle.

Given a valid layering, the true DAGs can be learned by applying SP to the first layer, then adjust all the lower layers by the first layer and repeat this process until the last layer. We call this recursive approach the Layered-SP.

The next shows that our modified approach is correct.

---

[2]To circumvent the problem of bad initialization and getting stuck in an equivalence class in practice, Solus et al. (2021) suggest implementing Greedy TSP with limited search depth d and restarting with different initialization for r times. These two tricks help circumvent the problem of bad initialization and getting stuck in equivalence class.

Algorithm 8: Layered-SP

Input : Constraint-based algorithm $\mathcal{A}$

Output: A topological ordering

$D \leftarrow$ DNA relations learned by Algorithm 4 with $\mathcal{A}$;

$D' \leftarrow$ an order-constraining subset of D obtained by dropping cycle-inducing DNA relations;

$L = (L_1, \ldots, L_m) \leftarrow$ layering deduced from D by Algorithm 7;

Run SP on L_1 with initialization compatible with D', and record the output ordering as π_1;

for $l = 2, \ldots, m$ **do**

    Run SP on L_l adjusted for $\cup_{i=1}^{l-1} L_i$ with initialization compatible with D', and record the output ordering as π_l

$\pi \leftarrow [\pi_1, \ldots, \pi_m]$;

return π.

Theorem 13 (Layered-SP)**.** *Let G be a DAG. Suppose observational data are drawn from a distribution $\mathbb{P}$ that is Markov to G, DNA-faithful with respect to $(\Omega_{\mathcal{A}}(\mathbb{P}), \bar{\Omega}_{\mathcal{A}}(\mathbb{P}))$, and SMR to G. Then the output of Algorithm 8, denoted π, satisfied $G^{\pi} \in [G]$.*

The proof above suggests that the sparest ordering can be learned by applying the sparsest permutation algorithm on each components of the incomplete ordering.

We note that in practice, Solus et al. (2021) suggest implementing Greedy TSP with limited search depth d and restart with different initialization for r times. These two implementation details help circumvent the problem of bad initialization and getting stuck in equivalence class. In our algorithm, if we have reasonably informative incomplete orderings, we can just leave $d = \infty$ and $r = 1$ for more consistent performance.

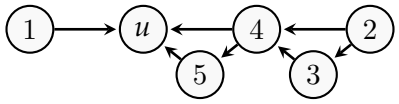


Figure 4.5: Algorithm 5 with level $K = 0$ discovers u being a sink. When accessing conditional independencies among others, we can exclude u from the search.

4.4.3 PC algorithm with DNA

As noted before, the PC algorithm learns a CPDAG from conditional independencies given sets of increasing sizes. In the PC algorithm, we may leverage DNA information by excluding non-ancestral neighbors in conditional independence tests. When accessing the independence relation between u and v from the perspective of u in a working skeleton C, instead of searching for d-separation sets among $\text{adj}(C, u)$, we use the following rule:

- If $u \not\rightsquigarrow v$, search $\text{adj}(C, u) \setminus \{w : w \not\rightsquigarrow u\}$; otherwise, search $\text{adj}(C, u) \setminus \{w : w \not\rightsquigarrow v \text{ and } w \not\rightsquigarrow u\}$.

In the next lemma we show the correctness of the modifications, and in Figure 4.5 we present a concrete example.

Lemma 15 (DNA and d-sep)**.** *The version of PC with neighborhood search replaced by the above rule is correct.*

It is worth noting that if $u \not\rightsquigarrow v$ and $v \not\rightsquigarrow u$, then u and v are non-adjacent in the true DAG. However, the orientation step of PC requires the d-separator of u, v, and hence we cannot simply remove the edge $u - v$ without searching and testing.

4.5 Numerical Experiments

We examine the performance of structure learning algorithms augmented by DNA. We generate 200 random Erdős-Renyi DAGs with $p = 10$ and expected neighborhood size s from 2

to 7. For each DAG, we build a linear SEM with coefficients drawn uniformly from $\pm[0.3, 1]$ and i.i.d. Gaussian error with variance drawn uniformly from $[1, 2]$.

We first compare the population versions of SP and PC with their DNA-modified versions. We plug in the conditional independence set $\Omega_\lambda(\mathbb{P}) = \{(i, j, S) : |\rho(i, j|S)| \leq \lambda\}$ with $\lambda = \{10^{-2}, 10^{-3}\}$ where ρ is the partial correlation. We report the number of conditional independence tests performed by each method, as well as their recovery rate, which is the proportion of times that the correct Markov equivalence class is recovered. A higher recovery rate under fixed λ means the method is less demanding with respect to faithfulness conditions, and is likely more statistically efficient. The results are shown in Figure 4.6. The DNA version of both SP and PC have higher recovery rate, showcasing the improvement on statistical efficiency provided by our proposed algorithms. The number of conditional independence tests also highlight the computational gains by augmenting SP and PC with DNA. Notably, even with low learning levels ($K = 0$), the DNA modifications significantly reduce the total number of tests performed, especially when the graph is moderately sparse. Higher learning level ($K = 1$) provides more improvement, though it can increase the number of tests in some settings.

We also compare the sample versions of SP, PC and their DNA-modified versions. To prevent false discoveries in the DNA Algorithm 5, we picked a large threshold for the d-connection step ($\lambda' = 0.2$). All other tests in SP and PC were performed at level $\lambda = 0.02$. We draw 10000 samples from each SEM and use them to infer the CPDAG. We report the recovery rate as well as the number of tests performed. As expected, DNA provides improvement over both SP and PC when the underlying truth is moderately sparse, and the improvement is more significant for SP. On the other hand, when the underlying true DAG is sparse, DNA could not provide much improvement, and false discoveries in DNA may hinder the performance.

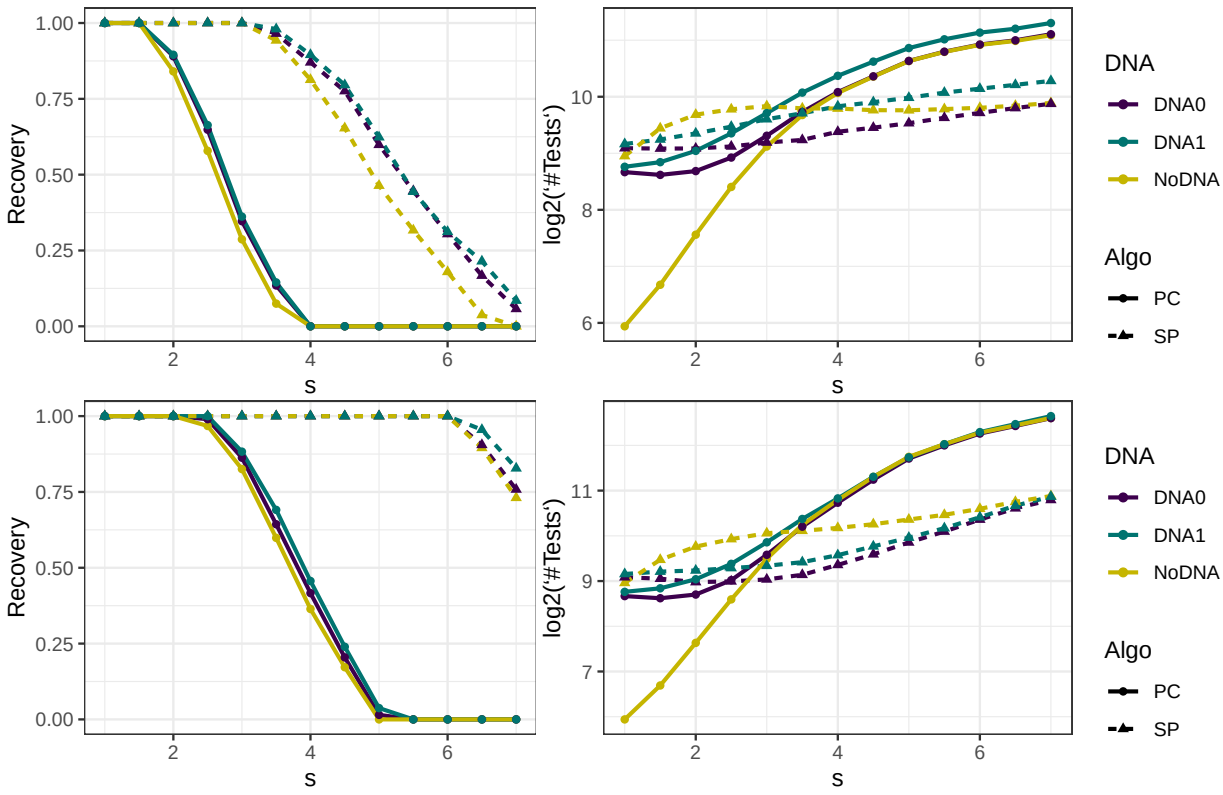


Figure 4.6: Recovery rate (Left) and Number of conditional independence tests (right) of the population version of SP, PC and their DNA modifications for random ER graphs with $p = 10$ nodes, expected neighborhood size s (x-axis), and $\lambda = 0.01$ (top row) and $\lambda = 0.001$ (bottom row).

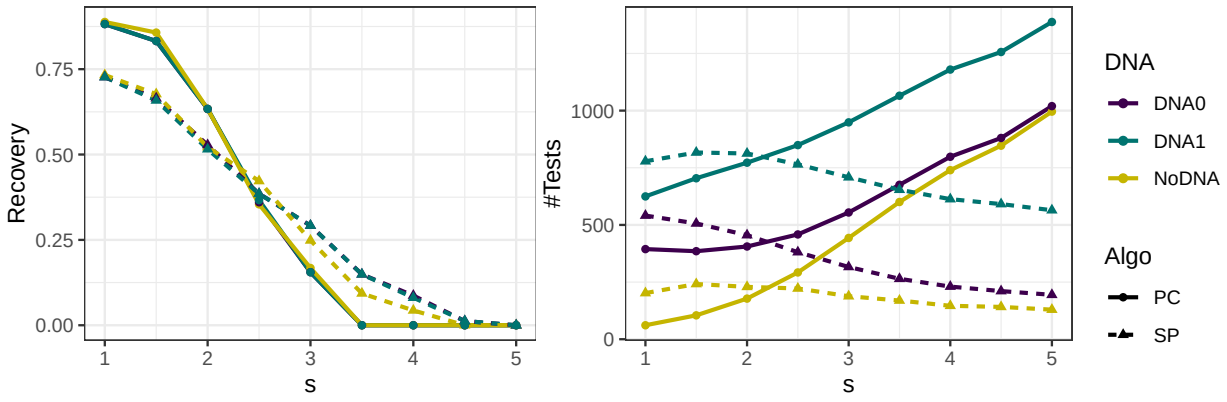


Figure 4.7: Recovery rate (Left) and Number of conditional independence tests (right) of the sample version of SP, PC and their DNA modifications performed on $n = 10000$ samples from random ER graphs with $p = 10$ nodes and expected neighborhood size s on x-axis.

4.5.1 DNA Learning with low learning levels

In this section we demonstrate that a large proportion of DNA relations can be learned by Algorithm 5 with low learning levels. We generate 1000 random Erdős-Renyi or power-law DAGs, and then run Algorithm 5 at learning level $K = 0, 1$ with the conditional independence oracle. We then deduce a layering of G using Algorithm 7. We report the proportion of all DNA learned and the proportion of edges in G that lie between the learned layers. A high proportion of inter-layer edges means L is informative about G.

The results are shown in Figure 4.8. It is evident that a large proportion of DNA can be learned by Algorithm 5 with very early stopping, even if the underlying graphs are large or dense. For this reason we recommend running Algorithm 5 with small $K = (0 \text{ or } 1)$. The results also show that the corresponding layering discovered by Algorithm 7 are most informative when the true DAGs are not too dense nor too sparse. This is due to the characterization of Lemma 4 which relies on a conditional independence and a conditional dependence: if too dense, there are not enough independences; and if too sparse, there are not enough dependences.

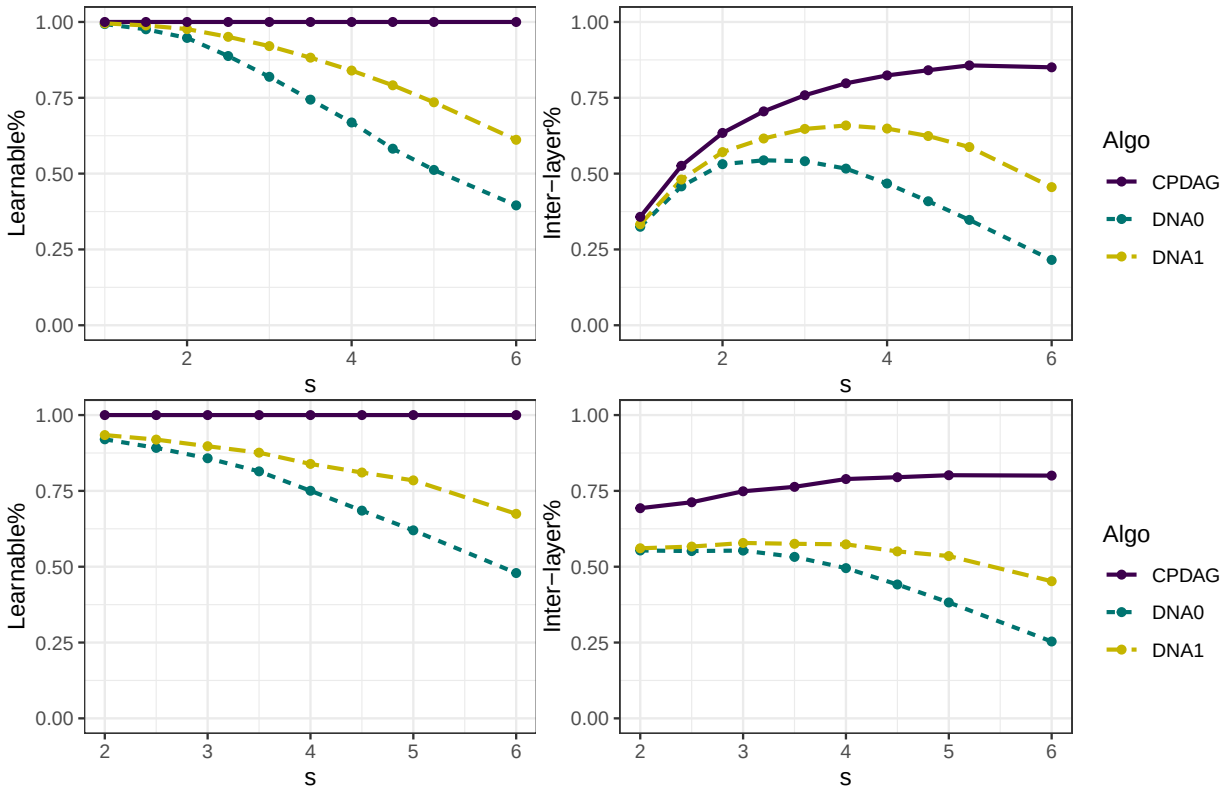


Figure 4.8: Proportion of DNA learned (left column), and inter-layer edges (right column) for Algorithm 5 with levels 0 and 1 in random Erdős-Renyi graphs (top row) and power-law graphs (bottom row) with $p = 10$ vertices. The x-axis is average-node degree.

4.6 Conclusion

We introduced definite non-ancestral (DNA) relations as intermediate targets of inference in structure learning. DNA relations can be learned from simple conditional independencies and lead to computational and statistical gains in DAG structure learning. DNA applications in graphs with latent variables would be interesting area of future research.

Chapter 5

LEARNING DIRECTED ACYCLIC GRAPHS FROM PARTIAL ORDERINGS

5.1 Introduction

Directed acyclic graphs (DAGs) are widely used to capture causal relationships among components of complex systems (Spirtes et al., 2001; Pearl, 2009; Maathuis et al., 2018). They also form a foundation for causal discovery and inference (Pearl, 2009). Probabilistic graphical models defined on DAGs, known as Bayesian networks (Pearl, 2009) have thus found broad applications in various scientific disciplines, from biology (Markowetz and Spang, 2007; Zhang et al., 2013) and social sciences (Gupta and Kim, 2008) to knowledge representation and machine learning and data mining (Heckerman, 1997). However, learning the structure of DAGs from observational data is very challenging, if not impossible. There are two main reasons for this difficulty. Firstly, it may not be possible to infer the direction of edges from observational data alone. In fact, unless the model is *identifiable* (see, e.g., Peters et al., 2014), observational data only reveal the structure of the Markov equivalent class of DAGs (Maathuis et al., 2018), captured by a complete partially directed acyclic graph (CPDAG) (Andersson et al., 1997). The second reason is computational: Learning DAGs from observational data is an NP-complete problem (Chickering, 1996). In fact, while a few polynomial time algorithms have been proposed for special cases, including sparse graphs (Kalisch and Bühlmann, 2007) or identifiable models (Chen et al., 2019; Ghoshal and Honorio, 2018; Peters et al., 2014; Wang and Drton, 2020; Shimizu et al., 2006; Yu et al., 2020), existing algorithms are not scalable to large-scale problems.

In spite of the many challenges of learning DAGs in general settings, the problem becomes very manageable if a *valid causal ordering* among variables is known (Shojaie and Michailidis,

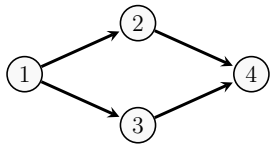


Figure 5.1: A directed graph with four nodes.

2010). In a valid causal ordering for a DAG G with node set V, any node j can appear before another node k (denoted $j \prec k$) only if there is no directed path from k to j. Multiple causal orderings may exists for a given DAG, as illustrated in the simple example of Figure 5.1, where both $O_1 = \{1 \prec 2 \prec 3 \prec 4\}$ and $O_2 = \{1 \prec 3 \prec 2 \prec 4\}$ are valid causal orderings.

Clearly, a known causal ordering of variables resolves any ambiguities about directions of edges in a DAG and hence addresses the first source of difficulty in estimation of DAGs discussed above. However, this knowledge also significantly simplifies the computation: given a valid causal ordering, DAG learning reduces to a variable selection problem that can be solved efficiently even in the high-dimensional setting (Shojaie and Michailidis, 2010), when the number of variables is much larger than the sample size.

In the simplest case, the idea of Shojaie and Michailidis (2010) is to regress each variable k on all preceding variables in the ordering, $\{j : j \prec k\}$. While simple and efficient, this idea, and its extensions (Shojaie et al., 2014), require a *complete* (or full) casual ordering of variables, i.e., a permutation of the list of m variables in DAG G. However, complete causal orderings are rarely available in practice. To relax this assumption, a few recent proposals have combined regularization strategies with algorithms that search over the space of orderings (Raskutti and Uhler, 2018). These algorithms are more efficient than those which search over the super-exponentially large space of DAGs (Friedman and Koller, 2003). Nonetheless, the computation for these algorithms remains prohibitive for moderate to large size problems (Manzour et al., 2021; Kucukyavuz et al., 2022).

In this paper, we relax the assumption of Shojaie and Michailidis (2010) and consider

the setting where a *partial* causal ordering of variables is known. This scenario—which is an intermediate between assuming a complete causal ordering and no assumption on causal ordering—occurs commonly in practice. An important example is the problem of identifying *direct* causal effects of multiple exposure variables on multiple outcomes (assuming no unmeasured confounders). More formally, let $X = \{X_1, \ldots, X_p\}$ and $Y = \{Y_1, \ldots, Y_q\}$ denote the set of p exposure and q outcome variables, respectively. Then, we have a partial ordering among X and Y variables, namely, $X \prec Y$. But, we do not have any knowledge of the ordering among X or Y variables themselves. Nonetheless, we are interested in identifying direct causal effects of exposures X on outcomes Y. This corresponds to learning edges from $X_1, \ldots, X_p$ to $Y_1, \ldots, Y_q$, which would form a bipartite graph.

Estimation of DAGs from partial orderings also arises naturally in the analysis of biological systems. For instance, in gene regulatory networks, often the set of transcription factors are known *a priori* and they are not expected to be affected by other 'target' genes. Similar to the previous example, here the set of transcription factors, X, appear before the set of target genes, Y, and the goal is to infer gene regulatory interactions. Similar problems also occur in integrative genomics, including in eQTL mapping (Ha and Sun, 2020).

Despite its importance and many applications, the problem of learning DAGs from partial orderings has not been satisfactorily addressed. In particular, as we show in the next section, various regression-based strategies currently used in applications result in incorrect estimates. As an alternative to these methods, one can use general DAG learning algorithms, such as the PC algorithm (Spirtes et al., 2001; Kalisch and Bühlmann, 2007), to learn the structure of the CPDAG and then orient the edges between X and Y according to the known partial ordering. However, such an approach would not utilize the partial ordering to estimate the edges and is thus inefficient. Finally, the recent proposal of (Wang and Michailidis, 2019) is also not computationally feasible as it requires a search over all possible orderings of variables.

To address the need for efficient algorithms for learning DAGs from partial ordering, we propose a new framework for incorporating the partial ordering information in Section 5.3

after formulating the problem in Section 5.2. We then investigate two popular Bayesian network models in Section 5.4, and propose consistent algorithms for estimation of high-dimensional DAGs. To simplify the presentation, the main ideas in these sections are presented for the special case of two-layer networks and we discuss the more general case of the algorithm and its extensions in Section 5.5. To the best of our knowledge, this proposal is the first approach for efficient estimation of DAGs from partial orderings.

Extensive simulations and an application in integrative genomics illustrate the advantages of the proposed approach.

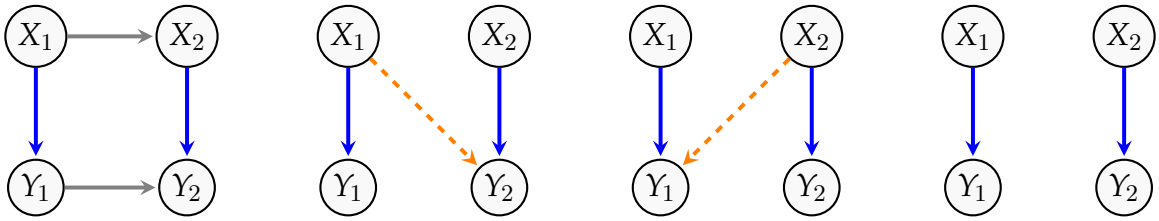
5.2 Learning Directed Graphs from Partial Orderings

5.2.1 Problem Formulation

Consider a DAG $G = (V, E)$ with the node set V, and the edge set $E \subset V \times V$. For the general problem, we assume that V is partitioned into L sets, $V_1, \ldots, V_L$ such that for any $\ell' \in \{1, \ldots, L\}$, nodes in $V_{\ell'}$ cannot be parents of nodes in any set $V_\ell, \ell < \ell'$. Such a partition defines a *layering* of G (Manzour et al., 2021), denoted $V_1 \prec V_2 \prec \cdots \prec V_L$. In fact, a valid layering can be found for any DAG, even though some V_ℓs may contain a single node. As such, the notion of layering here is general and we make no assumption on the size of each set nor the causal ordering of variables in each set V_ℓ or interactions among them, except that G is a DAG.

To simplify the presentation, we primarily focus on the case of two-layer, or, bipartite, DAGs, and defer the discussion of more general case to Section 5.5. We denote the nodes in the first layer, $V_1 = \mathcal{X} \equiv \{X_1, \ldots, X_p\}$ and those in the second layer, $V_2 = \mathcal{Y} \equiv \{Y_1, \ldots, Y_q\}$. In the case of causal inference for multiple outcomes discussed in Section 5.1, $X_1, \ldots, X_p$ represent the *exposure* variables, and $Y_1, \ldots, Y_q$ represent the *outcome* variables.

Throughout this paper, we often represent the edges of G using its adjacency matrix Θ, which satisfies $A_{jj} = 0$ and $A_{jj'} \neq 0$ if and only if $j' \in \mathrm{pa}_j$. For any bipartite DAG with layers



(a) full DAG (b) $H^{(0)}$ (c) $H^{(-j)}$ (d) H

Figure 5.2: Toy example illustrating the problem of estimation of directed graphs from partial orderings. a) The full DAGG, where the edges in B, defined in (5.1), are drawn in light blue, while the edges in A and C are shown in light gray. Here the true causal relations are linear and the goal is to estimate the bipartite graph H defined by edges $X_1 \longrightarrow Y_1$ and $X_2 \longrightarrow Y_2$; b) Estimate of H using $H^{(0)}$ in (5.2), obtained by regressing each Y_j, $j = 1, 2$ on $\{X_1, X_2\}$ using a linear model with $n = 1000$ observations, and drawing an edge if the corresponding coefficient is significant at $\alpha = 0.05$ significance level; c) estimate of H using $H^{(-j)}$ in (5.3) obtained by regressing each Y_j, $j = 1, 2$ on $\{X_1, X_2, Y_{-j}\}$ using a linear regression similar to (b); d) estimate of H obtained using the proposed approach.

V_1 and V_2 ($V_1 \prec V_2$), we can then partition Θ into the following block matrix

$$\Theta = \begin{bmatrix} A & 0 \\ B & C \end{bmatrix}, \tag{5.1}$$

where the zero constraint on the upper right block of Θ follows from the partial ordering. Here A and C contain the information on the edges amongst X_k, $k = 1, \ldots p$ and Y_j, $j = 1, \ldots q$, respectively. Both of these matrices can be written as lower-triangular matrices (Shojaie and Michailidis, 2010). However, there are generally no constraints on the entries of B. We denote by H the subgraph of G containing edges from X_ks to Y_js only, i.e. those corresponding to entries in B. Our goal is to estimate H using the fact that $V_1 \prec V_2$.

Figure 5.2a shows a simple example of a two-layer DAG G with two layers consisting of $p = q = 2$ nodes. This example illustrates the difference between full causal orderings of

variables in a DAG and partial causal orderings: In this case, the graph admits two full causal orderings, namely $\mathcal{O}_1 = \{X_1, X_2, Y_1, Y_2\}$ and $\mathcal{O}_2 = \{X_1, Y_1, X_2, Y_2\}$; either of these orderings can be used to correctly discover the structure of the graph. Here, the partial ordering of the variables defined by the layering of the graph to sets X and Y, i.e., $X \prec Y$, can be written as $\mathcal{O}' = \{\{X_1, X_2\}, \{Y_1, Y_2\}\}$. This ordering determines that X_js should appear before Y_ks in the causal ordering but does not restrict the ordering of $\{X_1, X_2\}$ and $\{Y_1, Y_2\}$. In this example, only $\mathcal{O}_1$ is consistent with the partial ordering $\mathcal{O}'$. However, $\mathcal{O}'$ is also consistent with $\{X_2, X_1, Y_2, Y_1\}$, which is not a valid causal ordering of G.

5.2.2 Failure of Simple Algorithms

In this section, we discuss the challenges of estimating the graph H and this problem cannot be solved using simple approaches.

Since the partial ordering of nodes, $V_1 \prec V_2$, provides information about direction of causality between the two layers of the network, we focus on simple *constraint-based* methods (Spirtes et al., 2001), which learn the network edges based on conditional independence relationships among nodes. This requires conditional independence relations among variables to be compatible with the edges in G; formally, the joint probability distribution $\mathcal{P}$ needs to be *faithful* to G (Spirtes et al., 2001). (As discussed in Section 5.4, our algorithm requires a weaker notion of faithfulness; however, for simplicity, we consider the classical notion of faithfulness in this section.)

Given the partial ordering of variables—which means that Y_js cannot be parents of X_ks—one approach for estimating the bipartite graph H is to draw an edge from X_k to Y_j whenever Y_j is dependent on X_k given all other nodes in the first layer, $\mathcal{X}_{-k} \equiv \{X_{k'}, k' \neq k\}$. Formally, denoting by $Y_j \perp\!\!\!\perp X_k$ the conditional independence of two variables Y_j and X_k, we define

$$H^{(0)} \equiv \left\{(k \to j) : Y_j \not\perp\!\!\!\perp X_k \mid \mathcal{X}_{-k}\right\}, \tag{5.2}$$

to emphasize that the estimate is obtained without conditioning on $Y_{j'} \neq Y_j$.

Figure 5.2b shows the estimated $H^{(0)}$ in the setting where true causal relationships are

linear. In this setting, the edges of H represent nonzero coefficients in linear regressions of each Y_j on its parents in G, pa_j. It can be seen that, in this example, the true causal effects from X_ks to Y_js are in fact captured in $H^{(0)}$; these are shown in solid blue lines in Figure 5.2b. However, this simple example suggests that $H^{(0)}$ may include spurious edges, shown by red dashed edges in the figure. The next lemma formalizes and generalizes this finding.

Lemma 16. *Assume that $\mathcal{P}$ is faithful with respect to G. Let $H^{(0)}$ be the directed bipartite graph defined in* (5.2)*. Then, if $\{X_1, \dots X_p\} \prec \{Y_1, \dots Y_q\}$,*

i) $X_k \to Y_j \in H^{(0)}$ whenever $X_k \to Y_j \in G$;

ii) for any path of the form $X_{k_0} \to Y_{j_1} \to \cdots \to Y_{j_0}$ such that $X_{k_0} \to Y_{j_0} \notin G$, $H^{(0)}$ will include a false positive *edge $X_{k_0} \to Y_{j_0}$.*

Lemma 16 suggests that if G includes edges among Y_js, failing to condition on X_js can result in falsely detected causal effects from X_ks to Y_js. Thus, one may consider an estimator that corrects for both X_{-k} and Y_{-j} when trying to detect the casual effects of X_k on Y_j, or in other words, declaring an edge from X_k to Y_j whenever $Y_j \not\perp\!\!\!\perp X_k \mid \{\boldsymbol{\mathcal{X}}_{-k}, \boldsymbol{\mathcal{Y}}_{-j}\}$. We denote the resulting estimate of H as $H^{(-j)}$. In other words,

$$H^{(-j)} = \{(k \to j) : Y_j \not\perp\!\!\!\perp X_k \mid \boldsymbol{\mathcal{X}}_{-k} \cup \boldsymbol{\mathcal{Y}}_{-j}\}. \tag{5.3}$$

Unfortunately, as Figure 5.2c shows, estimation based on this model may also include false positive edges. As the next lemma clarifies, the false positive edges in this case are due to the conditioning on common descendants of a pair of X_k and Y_j that are not connected in G, which is sometimes referred to as Berkson's Paradox (Pearl, 2009).

Lemma 17. *Assume that $\mathcal{P}$ is faithful with respect to G, and let $H^{(-j)}$ be the directed bipartite graph defined in $H^{(-j)}$. Then, if $\{X_1, \dots, X_q\} \prec \{Y_1, \dots Y_p\}$, for any $X_k \to Y_j \in G$, $X_k \to Y_j \in H^{(-j)}$. Moreover, for any triplets of nodes X_{k_0}, Y_{j_0} and Y_{j_1} that form an open collider in G (Pearl, 2009), i.e.*

- $X_{k_0} \rightarrow Y_{j_1} \leftarrow Y_{j_0}$

- $X_{k_0} \nrightarrow Y_{j_0}$

$H^{(-j)}$ *will include a* false positive *edge from* X_{k_0} *to* Y_{j_0}.

Remark 2. *Examining the proofs of Lemmas 16 and 17, if* $C = 0$ *in* (5.1)*, then* $H^{(0)} = H^{(-j)} = H$*. In other words, if* G *does not include any edges among* Y_k*s, then both* $H^{(0)}$ *and* $H^{(-j)}$ *provide valid estimates of* H*.*

While false positive edges in $H^{(0)}$ are caused by failing to condition on necessary variables, the false positive edges in $H^{(-j)}$ are caused by conditioning on extra variables that are not part of the correct causal order of variables. More generally, the partial ordering of nodes does not lead to a simple estimator that correctly identifies direct causal effects of covariates, $X_k\,(k = 1, \ldots, q)$, on outcomes, $Y_j\,(j = 1, \ldots, p)$. Of course, given an ideal test of conditional independence, we can estimate H by first learning the skeleton of G, using e.g. the PC Algorithm (Kalisch and Bühlmann, 2007), and then orienting the edges in H according to the partial ordering of nodes. However, such an algorithm uses the ordering information in a *post hoc* way, in the sense that the information is not utilized to learn the edges in H.

Building on the above findings, we next present a general framework for utilizing the partial ordering of variables, $V_1 \prec V_2$, when learning the graph H.

5.3 Incorporating Partial Orderings into DAG Learning

5.3.1 A new framework

In this section, we propose a new framework for estimating DAGs from partial orderings. The proposed approach is motivated by two key observations in Lemmas 16 and 17: First, the graphs $H^{(0)}$ and $H^{(-j)}$ include all true causal relationships from nodes $X_k, k = 1, \ldots, p$ in the first layer to nodes $Y_j, j = 1, \ldots, q$ in the second layer. Second, both graphs may also include additional edges; H^0 due to not conditioning on parents of Y_j, and H^{-j} due to conditioning on common children of X_k and Y_j.

Let $S_j^{(0)} := \{k : (k \to j) \in H^{(0)}\}$ and $S_j^{(-j)} := \{k : (k \to j) \in H^{(-j)}\}$. The next lemma, which is a direct consequence of Lemma 16 and 17, characterizes the intersection of these two sets.

Lemma 18. *Let G be a graph admitting the partial ordering $\boldsymbol{\mathcal{X}} \prec \boldsymbol{\mathcal{Y}}$. Then for any $j \in \boldsymbol{\mathcal{Y}}$, and $k \in S_j^{(0)} \cap S_j^{(-j)}$, either $k \in \mathrm{pa}(j)$, or k satisfies the following*

- *There exists a path $k \to j' \to \cdots \to j$ with $j' \in \boldsymbol{\mathcal{Y}}$; and*
- $\mathrm{ch}_j \cap \mathrm{ch}_k \neq \emptyset$.

Proof. This is a direct consequence of Lemma 16 and 17. □

Lemma 18 implies that even though $H_j^{(0)} \cap H_j^{(-j)}$ may contain more edges than the true edges in H, these additional edges must be in some special configuration. For each $j \in \boldsymbol{\mathcal{Y}}$, our approach evaluates the edges in $S_j^{(0)} \cap S_j^{(-j)}$ to identify the additional edges in an efficient way. Let $k \in S_j^{(0)} \cap S_j^{(-j)}$ and $(k \to j) \notin H$. Suppose, for simplicity, that conditional independencies are faithful to the graph G. Then, there must be some set of variables $Z \subset \boldsymbol{\mathcal{X}} \cup \boldsymbol{\mathcal{Y}}$ such that $X_l \perp\!\!\!\perp Y_j | Z$. In general, to find such a set of variables, we will need to search among subsets of $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$. However, utilizing the partial ordering, we can significantly reduce the complexity of this search. Specifically, we can restrict the search to the *conditional Markov blankets*, introduced next for general DAGs.

Definition 11 (Conditional Markov Blanket)**.** *Let $v \in V$ and $U \subset V \setminus v$ be arbitrary nodes and subsets in a DAG G. The* conditional Markov blanket *of v given U, denoted* $\mathrm{cmb}_U(v)$*, is the smallest set of nodes such that for any other set of nodes $W \subseteq V$,*

$$\mathbb{P}\{v \mid \mathrm{cmb}_U(v), U, W\} = \mathbb{P}\{v \mid \mathrm{cmb}_U(v), U\}. \tag{5.4}$$

The key idea in our framework is that by limiting the search to the conditional Markov blankets, *given the nodes in the previous layer*, we can significantly reduce the search space and the size of conditioning sets. Moreover, the conditional Markov blanket can be easily

Algorithm 9: Learning between-layer edges from Partial Orderings

Input : Observations from random variables $X_{1,\dots,p}$ and $Y_{1,\dots,q}$,

Output: A set of edges $\widehat{E}_{\mathcal{X}\to\mathcal{Y}}$

`/* Screening loop */`

for $j \in \mathcal{Y}$ **do**

  Infer $S_j^{(0)}$, $S_j^{(-j)}$, and $\mathrm{cmb}_{\mathcal{X}}(j)$;

$\widehat{E} \leftarrow \{(k,j) : j \in \mathcal{Y}, k \in S_j^{(0)} \cap S_j^{(-j)}\}$;

`/* Searching loop */`

for $\ell = 0, 1, \dots$ **do**

  for $(j,k) \in \widehat{E}$ **do**

    for $T \subseteq \mathrm{cmb}_{\mathcal{X}}(j)$, $|T| = \ell$ **do**

      if $X_k \perp\!\!\!\perp Y_j | X_{S_j^{(0)} \cap S_j^{(-j)} \setminus \{k\}} \cup Y_T$ **then** Remove (k,j) and break;

  if *No edge can be removed* **then** break;

return $\widehat{E}$.

inferred alongside with $H^{(0)}$ and $H^{(-j)}$. Together with the observations in Lemmas 16 and 17, especially the fact that conditioning on the nodes in the previous layer does not remove true causal edges, these reductions lead to improvements in both computational and statistical efficiency.

The new framework is summarized in Algorithm 9. In next section, we will show that that to learn the edges between any node in $\mathcal{X}$ and a node in $\mathcal{Y}$, it suffices to search over subsets of the conditional Markov blanket of nodes in $\mathcal{Y}$.

5.3.2 Graph Identification Using the Conditional Markov Blanket

The next lemma characterizes some key properties of the conditional Markov blanket. These properties enable learning the graph H using the conditional Markov blanket. Specifically,

we show that if a distribution satisfy the intersection property of conditional independence, i.e., $X \perp\!\!\!\perp A|B \cup C, X \perp\!\!\!\perp B|A \cup C \Rightarrow X \perp\!\!\!\perp A, B|\cup C$, then all Markov blankets and conditional Markov blankets are unique. The proof is left to the Appendix.

Lemma 19. *Let V be a set of random variables with joint distribution $\mathbb{P}$. Suppose the intersection property of conditional independence holds in $\mathbb{P}$. Then, for any variable $v \in V$, there exists a unique minimal Markov blanket* $\mathrm{mb}(v)$. *Moreover, for any $U \subset V \setminus v$,* $\mathrm{cmb}_U(v)$ *is also uniquely defined and* $\mathrm{cmb}_U(v) = \mathrm{mb}(v) \setminus U$.

We next define the faithfulness assumption needed for correct causal discovery from partial ordering. This assumption is trivially weaker than the general notion of strong faithfulness (Zhang and Spirtes, 2002).

Definition 12 (Layering-adjacency-faithfulness)**.** *Let $\boldsymbol{\mathcal{X}} \prec \boldsymbol{\mathcal{Y}}$ be a layering of random variables V with joint distribution $\mathbb{P}$. We say $\mathbb{P}$ is layering-adjacency-faithful to a DAG G with respect to the layering $\boldsymbol{\mathcal{X}} \prec \boldsymbol{\mathcal{Y}}$ if for all $k \in \boldsymbol{\mathcal{X}}$ and $j \in \boldsymbol{\mathcal{Y}}$, if $k \to j \in G$, then X_k and Y_j are (i) dependent conditional on $\boldsymbol{\mathcal{X}} \setminus \{k\}$ and (ii) dependent conditional on $\boldsymbol{\mathcal{X}}_{-k} \cup T$ for any $T \subseteq \boldsymbol{\mathcal{Y}}_{-j}$.*

Our main result, given below, describes how conditional Markov blankets can be used to effectively incorporate the knowledge of partial ordering into DAG learning and reduce the computational cost of learning causal effects of $\boldsymbol{\mathcal{X}}_k$s on $\boldsymbol{\mathcal{Y}}_j$s.

Theorem 14. *For a probability distribution that is Markov and layering-adjacency-faithful with respect to G, a pair of nodes $k \in \boldsymbol{\mathcal{X}}$ and $j \in \boldsymbol{\mathcal{Y}}$ are non-neighbor in G if and only if there exist a set $T \subseteq \mathrm{cmb}_{\boldsymbol{\mathcal{X}}}(j)$ such that $X_k \perp\!\!\!\perp Y_j|\boldsymbol{\mathcal{X}}_{\left(S_j^{(0)} \cap S_j^{(-j)}\right)\setminus k} \cup \boldsymbol{\mathcal{Y}}_T$. Consequently, Algorithm 9 correctly identifies direct causal effects of X_ks on Y_js.*

Theorem 14 shows that Algorithm 9—i.e., taking the intersection of $H^{(0)}$ and $H^{(-j)}$ and then searching within conditional Markov blankets to remove additional edges—recovers the correct bipartite DAG H. Next, we show that instead of inferring $H^{(0)}$ and $H^{(-j)}$ separately

and taking the interception, we can use $H^{(0)}$ to infer $H^{(0)} \cap H^{(-j)}$, since the algorithm only relies on $S_j^{(0)} \cap S_j^{(-j)}$ and $\mathrm{cmb}_{\mathcal{X}}(j)$ for each $j \in \mathcal{Y}$. More specifically, the next lemma shows that given $S_j^{(0)}$, we can learn $S_j^{(0)} \cap S_j^{(-j)}$ and $\mathrm{cmb}_{\mathcal{X}}(j)$ without having to learn $S_j^{(-j)}$ separately. This implies that Algorithm 9 does not require learning the unconditional Markov blankets, but instead only the conditional Markov blankets.

Lemma 20. *The followings hold for each $j \in \mathcal{Y}$:*

- $S_j^{(0)} = \{k \in \mathcal{X} : \mathcal{X}_k \not\perp\!\!\!\perp \mathcal{Y}_j | \mathcal{X}_{-k}\}$,
- $S_j^{(0)} \cap S_j^{(-j)} = \{k \in S_j^{(0)} : X_k \not\perp\!\!\!\perp Y_j | \mathcal{X}_{S_j^{(0)} \setminus k} \cup \mathcal{Y}_{-j}\}$,
- $\mathrm{cmb}_{\mathcal{X}}(j) = \{\ell \in \mathcal{Y}_{-j} : Y_j \not\perp\!\!\!\perp Y_\ell | \mathcal{X}_{S_j^{(0)}} \cap \mathcal{Y}_{-\{j,\ell\}}\}$.

Lemma 20 characterises the conditional dependencies of the sets used in Algorithm 9. For instance, using these characterizations, we can cast the set inference problems in Algorithm 9 as variable selection problems in general regression settings. In particular, defining

$$S_j^{(1)} = \{Z \in \mathcal{X}_{S_j^{(0)}} \cup \mathcal{Y}_{-j} : \mathcal{Y}_j \not\perp\!\!\!\perp Z \mid \mathcal{X}_{S_j^{(0)}} \cup \mathcal{Y}_{-j} \setminus Z\}, \tag{5.5}$$

$S_j^{(0)}$ and $S_j^{(1)}$ can be seen as selecting the relevant variables when regressing Y_j onto $\mathcal{X}$ and $\mathcal{X}_{S_j^{(0)}} \cup \mathcal{Y}_{-j}$, respectively. Note that $S_j^{(1)} \cap \mathcal{X} = S_j^{(0)} \cap S_j^{(-j)}$ and $S_j^{(1)} \cap \mathcal{Y} = \mathrm{cmb}_{\mathcal{X}}(j)$. In other words, the two sets used in Algorithm 9 can be deduced from $S_j^{(1)}$.

There are multiple benefits to directly inferring $H^{(0)} \cap H^{(-j)}$—i.e., by estimating $S_j^{(0)}$ and $S_j^{(1)}$—instead of separately inferring $H^{(0)}$ and $H^{(-j)}$. First, the graph $H^{(-j)}$ could be hard to estimate in high-dimensional settings, as learning $H^{(-j)}$ requires performing tests conditioned on $p+q-2$ variables. In contrast, using (5.5), we only perform tests conditioning on at most $\max(p-1, \max_j |S_j^{(0)}| + q - 2)$ variables. Moreover, the target $H^{(0)} \cap H^{(-j)}$ is often sparse (see Lemma 18) and can thus be efficiently learned in high-dimensional settings. In an extreme example, suppose each node in $\mathcal{X}$ has exactly one outgoing edge into $\mathcal{Y}$, and the nodes in $\mathcal{Y}$

have a common child. Then, $H^{(-j)}$ is fully connected and hard to learn, whereas $H^{(0)} \cap H^{(-j)}$ only has p edges.

The second advantage of directly inferring $H^{(0)} \cap H^{(-j)}$ is that, by using the conditional dependence formulation of sets as in Lemma 20, we can show that even if the sets $S^{(0)}$ and $S_j^{(1)}$ (consequently, $S_j^{(0)} \cap S_j^{(-j)}$ and $\text{cmb}_{\boldsymbol{X}}(j)$) are not inferred exactly, the algorithm is still correct as long as no false negative errors are made.

Lemma 21. *Algorithm 9 recovers exactly the truth DAG H if $S_j^{(0)}$ and $S_j^{(1)}$ are replaced with their arbitrary supersets.*

In the next section, we discuss specific algorithms for learning direct casual effects of X_ks on Y_js. These algorithms utilize the fact that given the partial ordering of nodes in G, the conditional Markov blanket of each node Y_j can be found efficiently by testing for conditional dependence of Y_j and $Y_{j'}, j' \neq j$ after adjusting for the effect of the nodes in the first layer, X. For instance, in the context of linear SEMs, this can be achieved efficiently by using (penalized) regressions. Alternatively, assuming multivariate normality, this can be achieved using the approach of Yin and Li (2011). The Gaussian copula transformation of Liu et al. (2009, 2012) and Xue and Zou (2012), can also be used to extend the applicability of the proposed approach to non-Gaussian distribution. In all of the above cases, existing results on variable selection consistency of network estimation methods can be coupled with a proof similar to that of PC Algorithm (Kalisch and Bühlmann, 2007) to establish consistency of the sample version of the proposed algorithm for learning high dimensional DAGs from partial orderings.

5.4 Learning High-Dimensional DAGs from Partial Orderings

Coupled with a consistent test of conditional independence, the general framework of Section 5.3 can be used to learn DAGs from any faithful probability distribution. This involves two main tasks for each for $j \in \boldsymbol{\mathcal{Y}}$: (a) obtaining a consistent estimate of the set of relevant variables, $S_j^{(0)}$, and (b) obtaining a consistent estimate of the conditional Markov blanket,

$\mathrm{cmb}_X(j)$ and the conditioning set in top layer, $S_j^{(0)} \cap S_j^{(-j)}$. By Lemma 20, this task is equivalent to learning $S_j^{(1)}$ in (5.5).

In this section, we propose efficient algorithms for two commonly-used families of probability distributions defined based on *linear* and *additive* structural equation models (SEMs). Suppose, without loss of generality, that the observed random vector $W = (W_1, \ldots, W_{p+q})$ is centered. In a structural equation model, W then solves an equation system $W_j = f_j(W_{\mathrm{pa}_j}, \varepsilon_j)$ for $j = 1, \ldots, p+q$, where ε_j are independent random variables with mean zero and f_js are unknown functions. A special family is the additive SEMs, also called Causal additive models (CAM), which can be written as

$$W_j = \sum_{k \in \mathrm{pa}_j} f_{jk}(W_k) + \epsilon_j \qquad j = 1, \ldots, p+q. \tag{5.6}$$

If, in addition, each f_{jk} is linear, then it is called a linear SEM,

$$W_j = \sum_{k \in \mathrm{pa}_j} \beta_{jk} X_k + \varepsilon_j, \qquad j = 1, \ldots, p+q. \tag{5.7}$$

For these families, we propose statistically and computationally efficient procedures for estimating the sets $S_j^{(0)}$ and $S_j^{(1)}$ for all $j \in \mathcal{Y}$. We will also discuss how these procedures can be applied in high-dimensional settings, when $p, q \gg n$.

5.4.1 Linear Structural Equation Models

Now we propose a concrete learning algorithm for data generated by linear structural equation models (SEMs) with Gaussian errors. In other words, we assume that the considered joint distribution $\mathbb{P}$ is that of a random vector $\mathcal{Y}$ that satisfies (5.1). We write

$$\begin{pmatrix} \mathcal{X} & \mathcal{Y} \end{pmatrix} = \begin{pmatrix} A & 0 \\ B & C \end{pmatrix} \begin{pmatrix} \mathcal{X} & \mathcal{Y} \end{pmatrix} + \varepsilon,$$

The vector ε is comprised of independent random variables independent of $\mathcal{X}$ and $\mathcal{Y}$.

In this section, we show that with good estimators for $S_j^{(0)}$ and $S_j^{(1)}$, as well as consistent test of conditional independence, Algorithm 9 is consistent even in high dimensions. For

convenience, we call the first part of the algorithm the *screening loop* and the second part the *searching loop*.

We first show that whenever the screening loop is successful—that is, when it returns a supergraph of H—the searching loop can consistently recover the truth. Given the multivariate-Gaussian observed data, the most direct way to test if disjoint sets T, U, Z satisfy $U \perp\!\!\!\perp T|Z$ is to estimate the sample partial correlation $\hat{\rho}(U, T|Z)$ and reject the hypothesis of conditional independence when $|\hat{\rho}(U, T|Z)| > \xi$ for some suitable threshold ξ. Consider the standard assumptions from PC theory below.

Assumption 7 (Graphical Model)**.** *The distribution of* $(\mathcal{X}, \mathcal{Y})$ *is multivariate Gaussian and layering-adjacency-faithful to* G*.*

Assumption 8 (Maximum reach level)**.** *Suppose that there exists some* $0 < b \leq 1$ *such that* $h_n := \max_{j \in \mathcal{Y}} |adj(j) \cap \mathcal{Y}| = O(n^{1-b})$ *and* $m_n = \max_{j \in \mathcal{Y}} |S_j^0 \cap S_j^{(-j)}| = O(n^{1-b})$*.*

Assumption 9 (Dimensions)**.** *The dimensions* p*,* q *satisfies* $pq^{m_n+1} = O(\exp(c_0 n^\kappa))$ *for some* $0 < c_0 < \infty$ *and* $0 \leq \kappa < 1$*.*

Assumption 10 (λ-strong-Layering-adjacency-Faithfulness)**.** *The partial correlations satisfy*

$$\inf_{j \in \mathcal{Y}, k \in \mathcal{X}, T \subseteq \mathcal{Y} \setminus \{j\}} \{|\rho(Y_j, X_k | \mathcal{Y}_T \cup \mathcal{X}_{-k})| : \rho(Y_j, X_k | \mathcal{Y}_T \cup \mathcal{X}_{-k}) \neq 0\} \geq c_n,$$
$$\sup_{j \in \mathcal{Y}, k \in \mathcal{X}, T \subseteq \mathcal{Y} \setminus \{j\}} \{|\rho(Y_j, X_k | \mathcal{Y}_T \cup \mathcal{X}_{-k})\} \leq M < 1 \text{ for some } M,$$

where $c_n^{-1} = O(n^d)$ *for some* $0 < d < \frac{1}{2} \min(b, 1 - \kappa)$ *and* κ *is defined in Assumption 9.*

The next result establishes the consistency of the searching step.

Theorem 15 (Searching step using partial correlation)**.** *Suppose Assumption 7-10 hold. Let the event* $\mathcal{A}\left(\widehat{S}^{(0)}, \widehat{S}^{(1)}\right) = \left\{\forall j \in \mathcal{Y} : \widehat{S}_j^{(0)} \supseteq S_j^{(0)}, \widehat{S}_j^{(1)} \supseteq S_j^{(1)}, |\widehat{S}_j^{(0)} \cap \widehat{S}_j^{(1)}| \leq n\right\}$ *denote the success of the screening step. Then, there exists some sequence of thresholds* $\alpha_n \to 0$ *as* $n \to \infty$ *such that the output* $\widehat{H}$ *of Algorithm 9 with test of partial correlation in search loop satisfies*

$$\mathbb{P}\left\{\widehat{H} = H | \mathcal{A}\left(\widehat{S}^{(0)}, \widehat{S}^{(1)}\right)\right\} = 1 - O(\exp(-Cn^{1-2d})).$$

The theorem above shows that the searching loop returns correct graph as long as the screening loop makes no type-II errors. Next, we discuss the screening steps, namely, the estimation problem for $S^{(0)}$ and $S^{(1)}$. We first demonstrate a screening guarantee for low-dimensional problems.

Proposition 1 (Screening with partial correlations)**.** *Suppose Assumption 7-10 holds and* $n \gg p + q$*. Then, with* $\widehat{S}_j^{(0)} = \{k \in \mathcal{X} : |\hat{\rho}(j, k|\mathcal{X}_{-k})| > c_n/2\}$ *and* $\widehat{S}_j^{(1)} = \{Z \in \mathcal{X}_{S_j^{(0)}} \cup \mathcal{Y}_{-j} : |\hat{\rho}(Y_j, Z|\mathcal{X}_{S_j^{(0)}} \cup \mathcal{Y}_{-j} \setminus Z)| > c_n/2\}$*, it holds that* $\mathbb{P}\left\{\mathcal{A}\left(\widehat{S}^{(0)}, \widehat{S}^{(1)}\right)\right\} = 1 - O(\exp(-Cn^{1-2d}))$.

In large graphs, learning $S^{(0)}$ and $S^{(1)}$ using partial correlations might be challenging. Inspired by Lemma 20, we can treat the screening problem as selecting non-zero regression coefficients from a linear regression model. (See Lemma 27.) A screening process is successful, i.e., the event $\mathcal{A}\left(\widehat{S}^{(0)}, \widehat{S}^{(1)}\right)$ is true, if we use some estimator satisfies a screen property, that is, it selects a superset of relevant variables.

One simple way to screen relevant variables is *Sure Independence Screening* (SIS, Fan and Lv, 2008). SIS selects the set of variables with largest marginal information at a give threshold t. Specifically, we define the SIS estimators

$$\widehat{S}_{j,\mathrm{SIS}}^{(0)}(t) = \{\text{variables corresponding to the } \lceil tn \rceil \text{ largest of all } \mathcal{X}^\top Y_j\}$$
$$\widehat{S}_{j,\mathrm{SIS}}^{(1)}(t) = \{\text{variables corresponding to the } \lceil tn \rceil \text{ largest of all } (\mathcal{X}_{\widehat{S}^{(0)}} \cup \mathcal{Y}_{-j})^\top Y_j\}.$$

The following result follows directly from Fan and Lv (2008).

Proposition 2. *Suppose Assumptions 7-10 hold with* $\kappa < 1 - 2d$*. Suppose the maximum eigenvalue of* $(\mathcal{X} \cup \mathcal{Y})$ *is lower bounded by* cn^ξ*. If* $2d + \xi < 1$ *then there exists some* $\delta < 1 - 2d - \xi$ *such that when* $t \sim cn^{-\delta}$ *with* $c > 0$*, we have for some* $C > 0$*,*

$$P\left[\mathcal{A}\left(\widehat{S}_{\mathrm{SIS}}^{(0)}(t), \widehat{S}_{\mathrm{SIS}}^{(1)}(t)\right)\right] = 1 - O(\exp(-Cn^{1-2d}/\log n))$$

Alternatively, we can select the sets $\widehat{S}_j^{(0)}$ and $\widehat{S}_j^{(1)}$ using two penalized regressions. For instance, using the lasso penalty, $\widehat{S}_j^{(0)}, j \in \mathcal{Y}$ can be found as

$$\widehat{\gamma}_j := \underset{\gamma \in \mathbb{R}^{p-1}}{\arg\min} \frac{1}{2n}\|Y_j - \gamma^\top \mathcal{X}\|_2^2 + \lambda_n^{(0)}\|\gamma_j\|_1, \qquad \widehat{S}_j^{(0)} = \left\{k : \widehat{\gamma}_{jk} \neq 0\right\}.$$

Similarly, $S^{(1)}$ can also be learned using a lasso regression on a different set of variables,

$$\widehat{\theta}_j(S) := \underset{\theta \in \mathbb{R}^{|S|+q-1}}{\arg\min} \frac{1}{2n} \|Y_j - \theta^\top [\mathcal{X}_S, \mathcal{Y}_{-j}]\|_2^2 + \lambda_n^{(1)} \|\theta_j\|_1, \qquad \widehat{S}_j^{(1)} = \left\{ k : \widehat{\theta}_{jk}(\widehat{S}_j^{(0)}) \neq 0 \right\}.$$

Using either approach, the screening step can be completed successfully under milder conditions in comparison to those needed for consistent variable selection. This is primarily because, the additional assumption required for the screening step is implied by the faithfulness assumption in the searching loop. We can obtain the following result for screening in high dimensions.

Proposition 3 (Screening with lasso)**.** *Suppose Assumption 7, 8 9 and 10 hold. Also assume that the minimal eigenvalue of* $\mathrm{Cov}(\mathcal{X} \cup \mathcal{Y})$ *is larger than some constant* $\Gamma_{\min}$*, and there exists some* $M > 0$ *such that* $\mathrm{Var}(Z|\mathcal{X} \cup \mathcal{Y} \setminus Z) > M$ *for all* $Z \in \mathcal{X} \cup \mathcal{Y}$*. Assume* $s_n = \max_j |\mathrm{mb}(j)| = O(n^{1-a})$ *and the rate parameters satisfies* $\kappa \leq \min(a, b)$*. Then lasso estimators with penalization level lower bounded with* $\lambda_n^{(0)} \asymp \sqrt{2 \log p / n}$ *and* $\lambda_n^{(1)} \asymp \sqrt{2 \log(p+q)/n}$ *satisfies* $\mathbb{P}\left\{ \mathcal{A}\left(\widehat{S}^{(0)}, \widehat{S}^{(1)} \right) \right\} \to 1$.

Together with Proposition 1, Proposition 2 or Proposition 3, Theorem 15 establishes the consistency of Algorithm 9 for linear Gaussian SEMs, in low- and high- dimensional settings.

In this paper we only provided a simple version of Algorithm 9 for linear Gaussian SEMs, These result can be extended to sub-Gaussian setting (Harris and Drton, 2013). Moreover, many other regularization methods can be used for screening instead of the lasso. The only requirement is the method is screening-consistent. Finally, inference-based procedures, such as debiased lasso (Zhang and Zhang, 2014; Geer et al., 2014; Javanmard and Montanari, 2014) can also fulfill the requirement of our screening step.

5.4.2 Causal Additive Models

As our second example, we consider causal additive models, that is, SEMs that are jointly additive in the form of (5.6). Like the linear SEM in the previous section, we will discuss the searching and screening steps separately.

For the searching step, we can use a general test of conditional independence. Here, we adopt the framework of Chakraborty and Shojaie (2022), based on conditional distance covariance (CdCov). Let two vectors $T \in \mathbb{R}^a$ and $U \in \mathbb{R}^b$, their CdCov given Z is defined as

$$\mathrm{CdCov}(T, U|Z) = \frac{1}{c_a c_b} \int_{\mathbb{R}^{a+b}} \frac{|f_{T,U|Z}(t,s) - f_{T|Z}(t) f_{U|Z}(s)|^2}{\|t\|_a^{1+a} \|s\|_b^{1+b}}.$$

We define $\rho^*(T, U|Z) = \mathbb{E}\left[\mathrm{CdCov}^2(T, U|Z)\right]$. It is easy to see that $\rho^*(T, U|Z) = 0 \Leftrightarrow T \perp\!\!\!\perp U|Z$.

Let $K_H(\omega) = |H|^{-1} K(H^{-1}\omega)$ be some kernel function where H is the diagonal matrix determined by bandwidth h and denote $K_{iu} = K_H(Z_i - Z_u)$. We also write $d_{ij}^T = \|T_i - T_j\|_a$ and $d_{ij}^U = \|U_i - U_j\|_b$. Define $d_{ijkl} := (d_{ij}^T + d_{kl}^T - d_{ik}^T - d_{jl}^T)(d_{ij}^U + d_{kl}^U - d_{ik}^U - d_{jl}^U)$ and the symmetric form $d_{ijkl}^S = d_{ijkl} + d_{ijlk} + d_{ilkj}$, We can use a plug-in estimate for $\rho^*(T, U|Z)$:

$$\widehat{\rho}^*(T, U|Z) := \frac{1}{n} \sum_{u=1}^{n} \Delta_{i,j,k,l;u} \text{ where } \Delta_{i,j,k,l;u} := \sum_{i,j,k,l} \frac{K_{iu} K_{ju} K_{ku} K_{lu}}{12(\sum_{m=1}^n K_{mu})^4} d_{ijkl}^S.$$

Following the derivation of Theorem 3.3 in Chakraborty and Shojaie (2022), we obtain the following result.

Proposition 4 (Searching with CdCov test)**.** *Suppose Assumption 8 and 9 holds with* $\kappa < 1/4$*. Assume there exists* $s_0 > 0$ *such that for all* $0 \le s < s_0$, $\max_{W \in \mathcal{X} \cup \mathcal{Y}} \mathbb{E}[\{] \exp(sW^2)\} < \infty$*, and the kernel function* $K(\cdot)$ *used to compute* $\widehat{\rho}$ *is non-negative and uniformly bounded over its support. Assume in addition the faithfulness condition that there exists some* c_n *such that*

$$\inf_{j \in \mathcal{Y}, k \in \mathcal{X}, T \subseteq \mathcal{Y}_{-j}} \{|\rho^*(j, k|\mathcal{Y}_T \cup \mathcal{X}_{-k})| : \rho^*(j, k|\mathcal{Y}_T \cup \mathcal{X}_{-k}) \neq 0\} > c_n,$$

where $c_n^{-1} = O(n^d)$ *with* $d < \frac{1}{4} - \frac{1}{2}\kappa$*, then condition on* $\mathcal{A}(\widehat{S}^{(0)}, S^{(1)})$*, the searching loop in Algorithm 9 using test of CdCOV returns the correct edge set with probability at least* $1 - O(\exp(-n^{1-2\gamma-2d})) - O(\exp(-n^\gamma))$.

Next we discuss the screening step for nonlinear SEM models. We consider the family of functions $f_{uv}^{(r)}(x_u) = \Psi_{uv}\beta_{uv}$, where Ψ_{uv} is a $n \times r$ matrix whose columns are basis functions

used to model the additive components f_{uv}, and β_{uv} is a r-vector containing the associated effects. We write ϕ_{uvt} as the t-th coefficient in Φ_{uv}. We denote Ψ_S as the concatenated basis functions in $\{\Psi_{uv} : u \in S\}$. Denote $\Sigma_{S,S} = (n^{-1}\Psi_S^\top \Psi_S)$.

In this model k has no direct causal effect on j if and only if $f_{jk} \equiv 0$. Under some regularity conditions, there exists a truncation parameter r large enough such that $f_{uv} \equiv 0 \Leftrightarrow f_{uv}^{(r)} \equiv 0 \Leftrightarrow \beta_{uv} = [0, \ldots, 0]^\top$ for all u and v. In high-dimensional problems, we estimate the coefficients with ℓ_1/ℓ_2 norm penalization. Concretely, fix a node $u \in V$, we can estimate optimize the following problem

$$\widehat{S}_j^{(0)} = \left\{ k : \hat{f}_{jk}^{(r)} \equiv 0 : \hat{f}_{jk}^{(r)} = \underset{\{f_{jl}\}_{l \in \mathcal{X}} \in \mathcal{F}^{(r)}}{\arg\min} \|Y_j - \sum_{l \in \mathcal{X}} f_{jl}(X_l)\|_n^2 + \lambda \sum_{l \in \mathcal{X}} \|f_{jl}(X_l)\|_n \right\}$$

$$\widehat{S}_j^{(1)} = \left\{ k : \hat{f}_{jk}^{(r)} \equiv 0 : \hat{f}_{jk}^{(r)} = \underset{\{f_{jl}\}_{l \in \mathcal{X}_{\widehat{S}^{(0)}} \cup \mathcal{Y}_{-j}} \in \mathcal{F}^{(r)}}{\arg\min} \|Y_j - \sum_{l \in \mathcal{X}_{\widehat{S}^{(0)}} \cup \mathcal{Y}_{-j}} f_{jl}(W_l)\|_n^2 \right.$$

$$\left. + \lambda \sum_{l \in \mathcal{X}_{\widehat{S}^{(0)}} \cup \mathcal{Y}_{-j}} \|f_{jl}(W_l)\|_n \right\}$$

where the group lasso joint penalization groups the r smoothing basis of each variable. Note that this estimator is similar to SPACEJAM (Voorman et al., 2014) with edges disagreeing with the layering information hard-coded as zero. The following results relies on the general error computation in Haris et al. (2019). A similar result can be found in Tan and Zhang (2019).

First we need to assume some smoothness assumption on the basis approximation.

Assumption 11 (Truncated basis)**.** *Suppose there exists* $r = O(1)$ *such that* $\{f_{uv}\}$ *are sufficiently sooth in the sense that*

$$|f_{uv}^{(r)}(x_u) - f_{uv}(x_u)| = O_p(1/r^t)$$

uniformly for all $u, v \in V$ *for some* $t \in \mathbb{N}$.

Next we assume two standard conditions for GAMs. We note that the compatibility condition maybe shown for random design, but for simplicity we just assume the conditions.

Assumption 12 (Compatibility)**.** *Suppose there exists some compatibility constant* $\phi > 0$ *such that if for all* $j \in \mathcal{Y}$ *and all functions in the form of* $f_j^{(r)}(x) = \sum_{k\in\mathcal{X}} f_{jk}^{(r)}(x_k)$ *that satisfy* $\sum_{k\in\mathcal{X}\setminus S_j^{(0)}} \|f_{jk}^{(r)}\|_n \le 3\sum_{k\in S_j^{(0)}} \|f_k^{(r)}\|_n$*, it holds that*

$$\sum_{k\in S_j^{(0)}} \|f_{jk}^{(r)}\|_n \le \|f_k^{(r)}\|\sqrt{|S_j^{(0)}|}/\phi,$$

for some norm $\|\cdot\|$*.*

Also assume for all $j \in \mathcal{Y}$ *and all functions in the form of* $f_j^{(r)}(x) = \sum_{k\in\mathcal{X}\cup\mathcal{Y}_{-j}} f_{jk}^{(r)}(w_k)$ *that satisfy* $\sum_{k\in\mathcal{X}\cup\mathcal{Y}_{-j}\setminus S_j^{(1)}} \|f_{jk}^{(r)}\|_n \le 3\sum_{k\in S_j^{(1)}} \|f_k^{(r)}\|_n$*, it holds that*

$$\sum_{k\in S_j^{(1)}} \|f_{jk}^{(r)}\|_n \le \|f_k^{(r)}\|\sqrt{|S_j^{(1)}|}/\phi,$$

Assumption 13 (GAM screening)**.** *Denote* $s_{\max} = \max_{j\in\mathcal{Y}} \max_{i\in\{0,1\}} |S_j^{(i)}|$ *and suppose* $s_{\max} = o(n/\log(p+q))$*. Let* $\lambda \asymp \sqrt{\log(p+q)/n}$*. Suppose*

$$\min_{u\in V} \min_{i\in\{0,1\}} \min_{v\in S_j^{(i)}} \|f_{vu}^{(r),0}\| = \Omega\left(s_{\max}\frac{\log(p+q)}{n}\right).$$

Let f_u^* *be an arbitrary function such that* $\sum_{i=1}^n f_u^*(x_{u,i}) = 0$*. Suppose there exists some constant* M^* *satisfying* $M^* = O(|S_j^{(1)}|\lambda/\phi^2)$ *such that* $f_u \in \mathcal{F}_{local}^{(r)}$ *if and only if* $\|f_u - f_u^*\|_n \le M^*$*. Further suppose that* $\mathcal{E}(f_u^*) = O(s_{\max}\lambda^2/\phi^2)$ *for all* u, k*.*

Proposition 5 (Screening with GAM)**.** *Suppose Assumption 11-13 hold. Then with* $n \to \infty$*,* $p + q = O(n^\xi)$ *for some* $\xi \ge 0$*, and the penalty level stated in Assumption 8, the resulting undirected graph from the screening loop of Algorithm 9 is a supergraph of* H*.*

Propositions 5 and 4 establish the consistency of the proposed framework in Algorithm 9 for learning DAGs from joint-additive models, facilitating causal structure learning for a expressive family of distributions.

5.5 Extensions and Other Considerations

5.5.1 Learning Edges within Layers

In the previous sections we focused on the problem of learning edges between the layers $\mathcal{X}$ and $\mathcal{Y}$. In particular, Algorithm 9 provides a framework that first finds a set that contains just a little bit more than all the true edges, and then use a search loop to remove the additional edges. The same idea can be applied to learning edges within the variables $\mathcal{Y}$: As suggested by Lemma 19, for each $j \in \mathcal{Y}$, all of its adjacent nodes are contained in $S_j^{(1)}$. This suggests that we can learn edges in $\mathcal{Y}$ by simply modifying Algorithm 9 to run the search loop on $\{(k, j) : k \in S_j^{(1)}, j \in \mathcal{Y}\}$ instead of only on those between $\mathcal{X}$ and $\mathcal{Y}$. After recovering the skeleton, edges among $\mathcal{Y}$ can be oriented using Meek's orientation rules Meek (1995). Given correct d-separators and skeleton, the rules can orient as many edges as possible without making mistake.

In order to successfully recover within layer edges from observational data, we need to assume faithfulness among these edges. Following Ramsey et al. (2006), we characterize the faithfulness condition required to learn DAG via constraint-based algorithm as two parts: (a) adjacency faithfulness, which means that neighboring nodes are not associated with conditional independence; and (b) orientation faithfulness, which means that v-structures can be identified by exclusion of common child in separator. The orientation faithfulness can be defined in our context as follows.

Assumption 14 (Within-layer-faithfulness)**.** *Assume that for all adjacent pair $j, i \in \mathcal{Y}$, it holds that $Y_i \not\perp\!\!\!\perp Y_j | X \cup T$ for any set $T \subseteq Y \setminus \{i, j\}$. Also assume that for any unshielded triple (ℓ, j, k) with $j \in \mathcal{Y}$, if $\ell \to j \leftarrow k$ then the variables corresponding to ℓ and k are dependent given any subset of $\mathcal{X} \cup \mathcal{Y} \setminus \{\ell, k\}$ that contains j; otherwise the variables corresponding to ℓ and k are dependent given any subset of $\mathcal{X} \cup \mathcal{Y} \setminus \{\ell, k\}$ that does not contains j.*

To describe the graphical object learned by the new framework, we need to introduce a new notion of equivalence, as with known partial ordering, the Markov equivalent class of

DAGs can be reduced. For example, if among three variables X_1, X_2, X_3, the only conditional independent relation is $X_1 \perp\!\!\!\perp X_3 | X_2$, but we know $X_1 \prec \{X_2, X_3\}$, then the edge $X_2 \to X_3$ is identifiable. We define partial-ordering-Markov-equivalence simply as Markov equivalence restricted to partial ordering. This equivalent class can be represented by a maximally oriented partial DAG (maximal PDAG, Perkovic et al., 2018). With this notion, we can describe the target of learning of within- and between-layer edges as learning the maximal PDAG of the true G given the background information $\boldsymbol{\mathcal{X}} \prec \boldsymbol{\mathcal{Y}}$.

Lemma 22 (within-layer nodes)**.** *Suppose the conditions for Theorem 15 and Assumption 14 hold. Then, Algorithm 9 with an additional orientation step by Meek's rules recovers the maximal PDAG of G given the background information $\boldsymbol{\mathcal{X}} \prec \boldsymbol{\mathcal{Y}}$.*

Proof. The proof is identical to that of Theorem 14. The only additional piece needed for successful recovery of PDAG is the orientation step. The known partial ordering is a form of background knowledge of edge orientation, which, combined with Meek's rules of orientation, returns the maximal PDAG (see Meek, 1995, Problem (D)). □

5.5.2 Directed Graphs with Multiple Layers

The theory and algorithm developed in Section 5.4 can be extended to scenarios with multiple layers. To facilitate this discussions, we introduce a general representation of the problem. Suppose V is a random vector following some distribution Markov to a graph $G = (V, E)$. Suppose G admits a partial-ordering $\boldsymbol{O} = \{V_1 \prec \ldots \prec V_r\}$ where $V = \cup_{\ell=1}^{L} V_\ell$. Parallel to the notation in 2-layer case, we define, for each $j \in V_\ell$, $1 \le \ell \le L$,

$$S_j^{(0)} := \begin{cases} \emptyset & \text{if } j \in V_1 \\ \left\{ k \in \cup_{i=1}^{\ell-1} V_i : V_k \not\perp\!\!\!\perp V_j | \left(\cup_{i=1}^{\ell-1} V_i\right) \setminus \{k\} \right\} & \text{otherwise} \end{cases}$$

$$S_j^{(1)} := \left\{ k \in V_\ell \setminus \{j\} : V_k \not\perp\!\!\!\perp V_j | V_{S_j^{(0)}} \cup V_\ell \setminus \{k, j\} \right\}$$

The following lemma is an extension of Lemma 18.

Lemma 23. *Let G be a graph admitting the partial ordering $\mathcal{O} = \{V_1 \prec \ldots \prec V_L\}$. The following holds:*

- *For each $j \in V_\ell$, $2 \leq \ell \leq r$, a node $k \in S_j^{(0)} \cap S_j^{(1)}$ if and only if $(k, j) \in G$ or $(k, j) \notin G$ but there exists a path $k \to j' \to \cdots \to j$ with $j' \in V_\ell$ and $\mathrm{ch}(j) \cap \mathrm{ch}(k) \cap V_\ell \neq \emptyset$.*

- *For each $j \in V_\ell$, $1 \leq \ell \leq r$, a node $k \in S_j^{(1)} \cap V_\ell$ if and only if $k \in adj(G, j)$ or $k \notin adj(G, j)$ but $\mathrm{ch}(j) \cap \mathrm{ch}(k) \cap V_\ell \neq \emptyset$.*

Proof. The first statement follows directly from Lemma 18 (treating $\cup_{i=1}^{\ell-1} V_i$ as $\mathcal{X}$ and V_ℓ as $\mathcal{Y}$). The second statement follows from Lemma 19 that $S_j^{(1)} \cap V_\ell = \mathrm{mb}(j) \cap V_\ell$. □

The above lemma suggests a general framework for utilizing any layering-information to facilitate DAG learning. The multi-layer version of the algorithm is presented as Algorithm 10.

The faithfulness condition required for the success of this framework is given below.

Assumption 15 (Layering-faithfulness)**.** *For a graph $G = (V, E)$ admitting a partial-ordering $\mathcal{O} = \{V_1 \prec \ldots \prec V_L\}$, we say a distribution $\mathbb{P}$ is layering faithful to G with respect to $\mathcal{O}$ if the followings hold:*

- ***Adjacency faithfulness**: For all non-adjacent pair j, k with $j \in V_\ell$, $k \in V_{\ell'}$, $\ell \geq \ell'$, it holds that $W_j \not\perp\!\!\!\perp W_k | W_{\cup_{i=1}^{\ell-1} V_i \setminus \{k\}}$ and $W_j \not\perp\!\!\!\perp W_k | W_{(\cup_{i=1}^{\ell-1} V_i) \cup T \setminus \{k\}}$ for all $T \subseteq V_s \setminus \{j, k\}$.*

- ***Orientation faithfulness**: For all unshielded triples (ℓ, j, k) such that j, k are in the same layer V_s and ℓ is in some previous layer, if the configuration of the path (ℓ, j, k) is $\ell \to j \leftarrow k$ then $W_\ell \not\perp\!\!\!\perp W_k | W_T \cup \{j\}$ for all $T \subseteq \cup_{i=1}^{s} V_i \setminus \{\ell, k\}$; otherwise $W_\ell \not\perp\!\!\!\perp W_k | W_T$ for all $T \subseteq \cup_{i=1}^{s} V_i \setminus \{\ell, k, j\}$.*

We note that orientation faithfulness is only needed for triplets when at least two of the three nodes are in the same layer. This is because otherwise the orientation is already implied by partial ordering.

Algorithm 10: DAG learning from Partial Orderings

Input : Observations from random variables $W_{1,\dots,p} \sim \mathbb{P}_G$,
Partial Ordering $\mathbb{O} = \{V_1 \prec \dots \prec V_r\}$ where $V = \cup_{i=1}^{r} V_i$

Output: A estimated edge set of G

for $j \in V$ **do** Infer $S_j^{(0)}$ and $S_j^{(1)}$;

$\widehat{E}_{\text{between}} \leftarrow \{(k,j) : 2 \le \ell \le r, j \in V_\ell, k \in S_j^{(0)} \cap S_j^{(1)}\}$;

$\widehat{E}_{\text{within}} \leftarrow \{(k,j) : 1 \le \ell \le r, j \in V_\ell, k \in S_j^{(1)} \cap V_\ell\}$;

for $d = 0, 1, \dots$ **do**

  for $\ell = 0, 1, \dots, r$ **do**

    for $(k,j) \in \widehat{E}_{\text{between}} \cup \widehat{E}_{\text{within}}$, $j \in V_\ell$ **do**

      for $T \subseteq S_j^{(1)} \cap V_\ell$, $|T| = d$ **do**

        if $V_k \perp\!\!\!\perp V_j | V_{\left(S_j^{(0)} \cap S_j^{(1)}\right) \cup T \setminus \{k\}}$ **then** Remove (k,j) and break;

  if *No edge can be removed* **then** break;

Orient all edges in $\widehat{E}_{\text{between}}$ by $\mathbb{O}$, then apply Meek's rules to orient edges in $\widehat{E}_{\text{within}}$;

return $\widehat{E}_{\text{between}} \cup \widehat{E}_{\text{within}}$.

Theorem 16. *Under Assumption 15, the population version of Algorithm 10 recovers the maximal PDAG of G.*

Proof. By Lemma 23, the result of screening step is a superset of the edges in G. Under the adjacency faithfulness assumption, all conditional independencies checked by the algorithm corresponds to d-separation and therefore the correct skeleton is recovered. Finally, given correct d-sepration relations, the orientation rules are complete and maximal Perkovic et al. (2018). □

5.5.3 Weaker notions of Partial-Ordering

We note that Algorithm 10 can be successful even if ordering information is not available for some variables. The idea of using a screening loop and a search loop to reduce computational burden can be generally applied. In general, suppose that for every variable $j \in V$ there exist a set $W_j^{\prec} \prec j$ and $W_j^{\succ} \succ j$, then we slightly modify the construction of $S^{(0)}$ and $S^{(1)}$: for each $j \in V_\ell$,

$$S_j^{(0)} := \left\{ k \in W_j^{\prec} : V_k \not\perp\!\!\!\perp V_j | W_j^{\prec} \setminus \{k\} \right\}$$
$$S_j^{(1)} := \left\{ k \in V \setminus (W_j^{\prec} \cup W_j^{\succ} \cup \{j\}) : V_k \not\perp\!\!\!\perp V_j | V_{S_j^{(0)}} \cup \left(V \setminus (W_j^{\prec} \cup W_j^{\succ} \cup \{j,k\}) \right) \right\}$$

It is easy to see that Lemma 23 hold with this generalized notion too.

Specifically, this means that Algorithm 10 can also handle the setting in which V can be partitioned into disjoint sets $V_1, \ldots, V_r, V'$, such that $V_1 \prec V_2 \ldots \prec V_r$ and V' contains nodes with no partial ordering information. This situation can arise in experimental settings where we have layered experiment design $(V_1, \ldots, V_{r-1})$ and outcome (V_r) variables, as well as demographic variable (V') which are ambiguous in the causal ordering.

5.6 Numerical Experiments

5.6.1 Simulation Studies

In this section we compare our proposed method Algorithm 10, referred to as PODAG, with the PC algorithm. We also include a modified version of PC that utilizes the partial ordering information. In PC, with a pair of nodes j, k and a working skeleton C, we checks all subsets of size ℓ among the neighbors $\mathrm{adj}(C, j)$ (or $\mathrm{adj}(C, k)$); in PC+, if $j \succeq k$, then we exclude the neighbors that are non-ancestral to $\{j, k\}$ and only check subsets subsets among $\mathrm{adj}(C, j) \cap \{i : i \preceq k\}$ (or $\mathrm{adj}(C, k) \cap \{i : i \preceq k\}$). It is a well-known fact that d-separators do not have to include non-ancestral nodes, that is, if S is a d-separator for j, k, then $S \cap \mathrm{an}(j, k)$ is also a d-separator. Consequently, the population version of PC+ is correct given a valid partial ordering.

Faithfulness condition

In this study we look at the minimal absolute value of partial correlations checked by the different algorithms. For a constraint-based structure learning method that tests condition independencies in the collection of tuples $\mathcal{L}$, we define

$$\rho^*_{\min}(\mathcal{L}) = \min_{\{j,k,S\}\in\mathcal{L}} \{|\rho(j,k|S)| : \rho(j,k|S) \neq 0\}.$$

This quantity can be viewed as the strength of faithfulness condition required to learn this DAG. A small value of $\rho^*_{\min}(\mathcal{L})$ indicates a weaker separation between signal and noise for the method $\mathcal{L}$, and it would be harder to recover the correct DAG using sample partial correlations. On the other hand, a larger value of $\rho^*_{\min}(\mathcal{L})$ indicates superior statistical efficiency.

We randomly generated 100 DAGs with 20 nodes and expected number of edges equal to 2. For each DAG, we construct a linear Gaussian SEM with parameters draw uniformly from $(-1, 0.1) \cup (0.1, 1)$. We inspect three algorithms, PC, PC+, and PODAG, and computed their corresponding $\rho^*_{\min}(\mathcal{L}_{\text{PC}}), \rho^*_{\min}(\mathcal{L}_{\text{PC+}}), \rho^*_{\min}(\mathcal{L}_{\text{PODAG}})$. We also counted the number of conditional independence tests performed by each method as a measure of computational efficiency. The results are shown in Figure 5.3. It is evident that the faithfulness requirement for PC is stronger than PC+, which are both stronger than PODAG. Noticeably, though PC+ utilizes the partial ordering information, still its computational and statistical efficiency is subpar compared to PODAG.

Linear Gaussian SEMs

We generate ER graph graphs with $p = 50, 100$ vertices and $2p$ edges. A weight matrix of linear SEM is generated with respect to the graph, with non-zero parameters drawn uniformly from $\pm(0.1, 1)$. We generate $n = 100, 200, 500$ samples using the SEM with Gaussian error. With respect to the causal ordering, we split the vertex set into K equal sized layers. We compare the performance of PC, PC+ and POPC. In this study, we used debiased lasso (Javanmard and Montanari, 2018), which is tuning-free, in the screening loop, and used

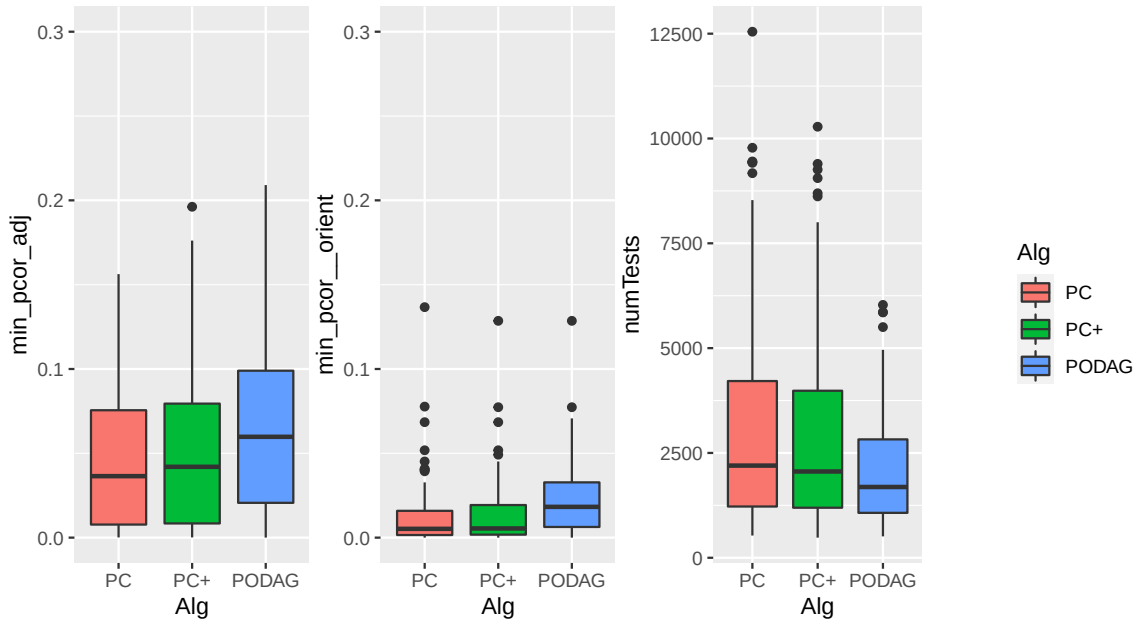


Figure 5.3: We computed $\rho^*_{\min}(\mathcal{L}_{\text{PC}}), \rho^*_{\min}(\mathcal{L}_{\text{PC+}}), \rho^*_{\min}(\mathcal{L}_{\text{PODAG}})$ for recovering the skeleton (left) and the entire DAG (middle). The number of conditional independence tests is shown on the right panel.

partial correlations in the searching loop. The performance of PODAG is substantially superior than PC and PC+.

JAMs

We generate ER graph with $p = 50, 100$ vertices and $2p$ edges. A weight matrix of SEM with cubic base function is generated with respect to the graph, with non-zero parameters drawn uniformly from $\pm(0.1, 1)$. We generate $n = 100, 200, 500$ samples using the SEM with Gaussian error. The joint distribution is non-Gaussian. With respect to the causal ordering, we split the vertex set into K equal sized layers. In this study, we used SPACEJAM (Voorman et al., 2014) in the screening loop, and used kernel-based CI test (Zhang et al., 2011) in the searching loop. The preformance is compared with PC with kernel-based CI test, also called kernel-PC, and the corresponding PC+ version. The performance of PODAG is substantially superior than PC and PC+.

5.6.2 Quantitative Trait Loci Mapping

In this section we present a practical application of PODAG to real data analysis problem Nica and Dermitzakis (2013). Expression Quantitative trait loci (eQTL) mapping is a powerful approach for identifying sequence variants that alter gene function. The analysis aims to recover the direct association between markers of genetic variation located at specific regions of the genome, and the expression level of the gene. In this study we consider the yeast expression data set (Brem and Kruglyak, 2005), which contains the eQTL data of 112 segments, each with 585 shared markers and 5428 target genes. Since this analysis focuses on the direct association from marker to expression level, they can be regarded as a 2-layer partially ordered network. We randomly select 50 markers and 50 genes and aim to recover the direct associations between the markers and expressions. The results are shown in Figure 5.8. The leftmost panel shows the estimated true associations, and the middle two panels show estimation from H^0 and H^{-j}. The right panel demonstrates that their intersection is very close to truth, as suggested by Lemma 18.

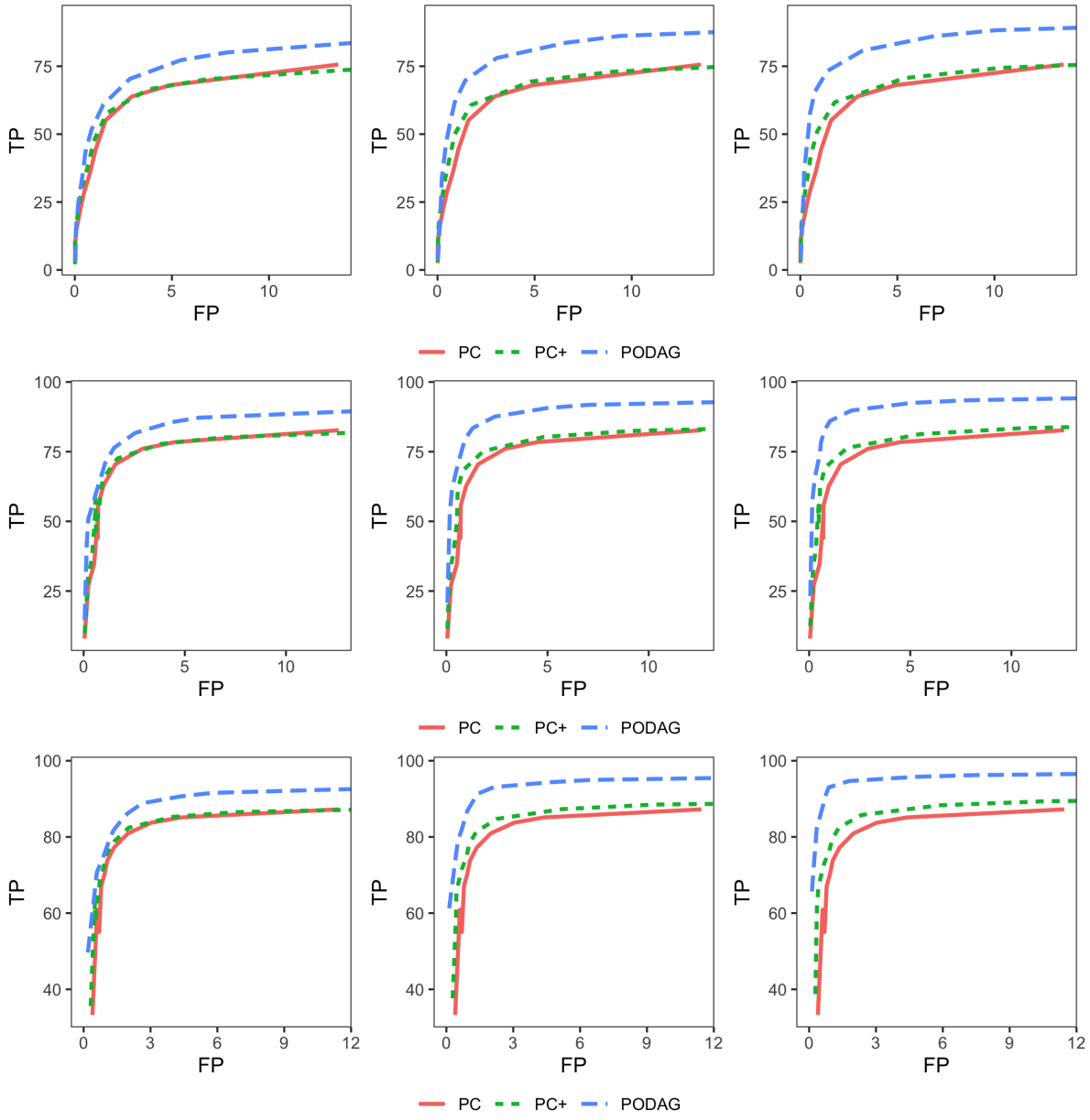


Figure 5.4: Number of true positive edges (TP) versus false positive edges (FP) in the entire graph. DAGs with skeleton of random ER graphs of $p = 50$ variables and expected number of edges $|E| = 2p$. Samples drawn from Gaussian SEM with sample size $n = 100$ (top), $n = 200$ (middle), $n = 500$ (bottom). Partial ordering information supplied to the algorithms in the form of 2 (left),5 (middle), 10 (right) layers.

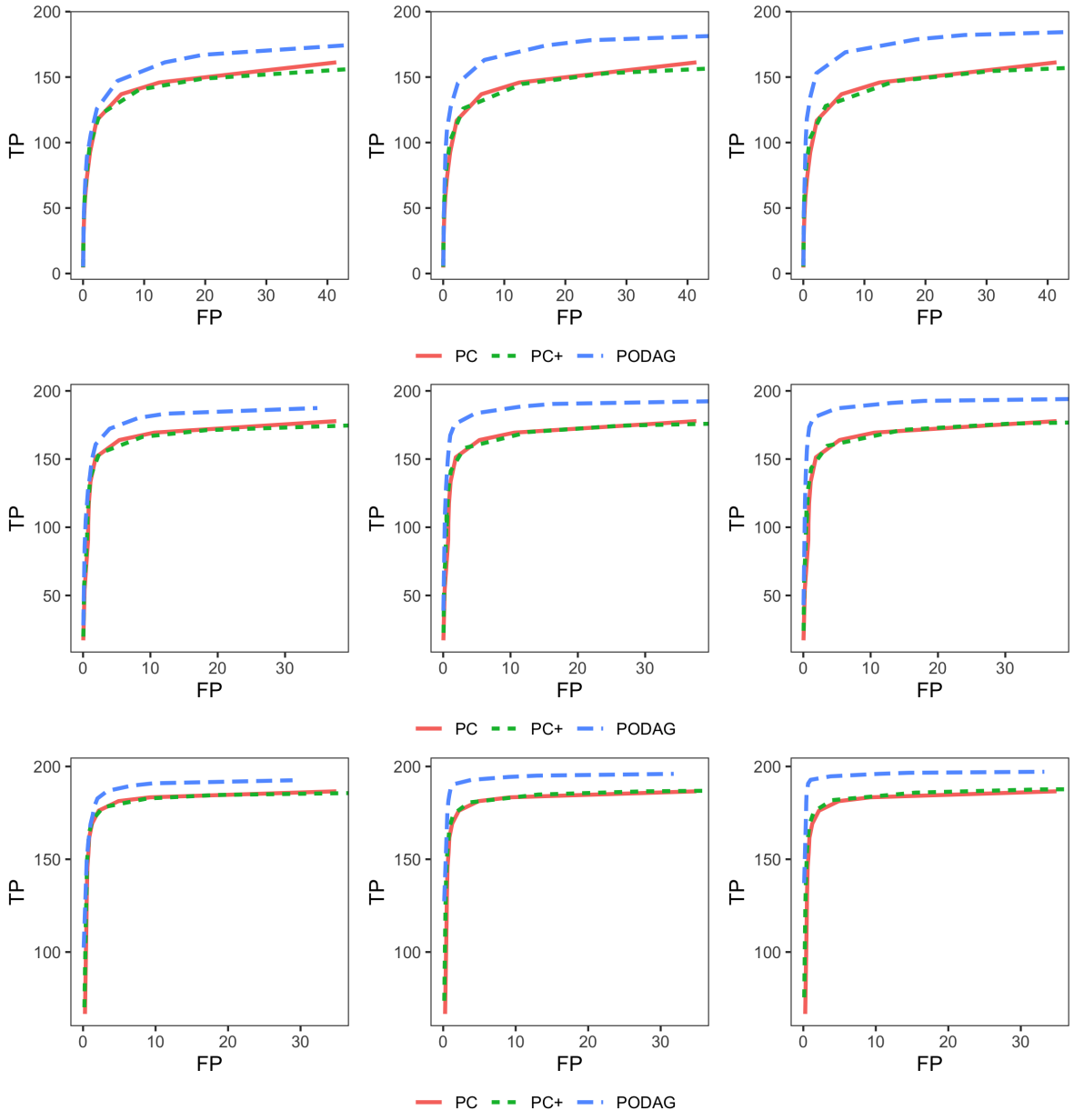


Figure 5.5: Number of true positive edges (TP) versus false positive edges (FP) in the entire graph. DAGs with skeleton of random ER graphs of $p = 100$ variables and expected number of edges $|E| = 2p$. Samples drawn from Gaussian SEM with sample size $n = 100$ (top), $n = 200$ (middle), $n = 500$ (bottom). Partial ordering information supplied to the algorithms in the form of 2 (left),5 (middle), 10 (right) layers.

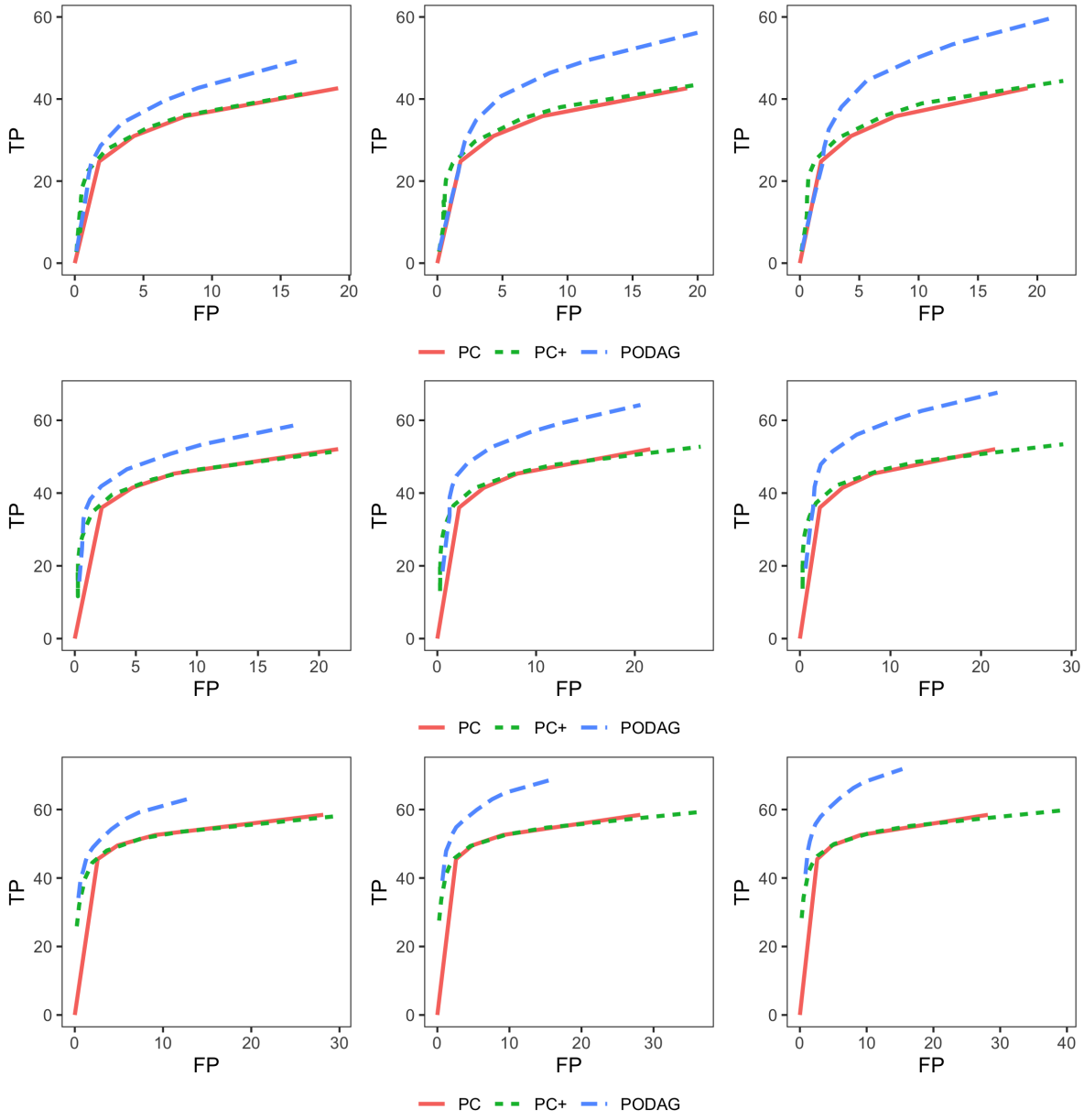


Figure 5.6: Number of true positive edges (TP) versus false positive edges (FP) in the entire graph. DAGs with skeleton of random ER graphs of $p = 50$ variables and expected number of edges $|E| = 2p$. Samples drawn from JAM with qubic spline bases and sample size $n = 100$ (top), $n = 200$ (middle), $n = 500$ (bottom). Partial ordering information supplied to the algorithms in the form of 2 (left),5 (middle), 10 (right) layers.

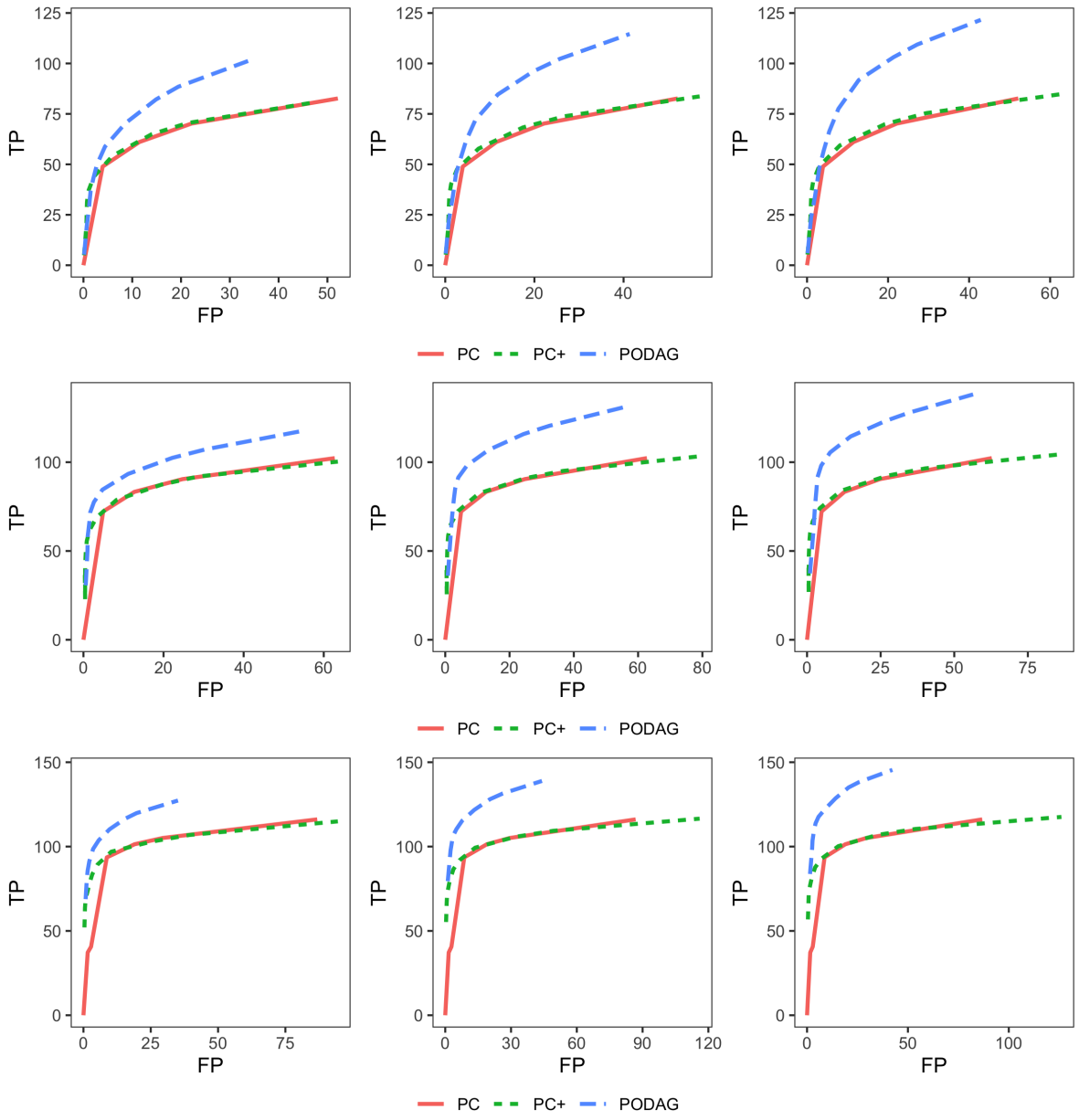


Figure 5.7: Number of true positive edges (TP) versus false positive edges (FP) in the entire graph. DAGs with skeleton of random ER graphs of $p = 100$ variables and expected number of edges $|E| = 2p$. Samples drawn from JAM with qubic spline bases and sample size $n = 100$ (top), $n = 200$ (middle), $n = 500$ (bottom). Partial ordering information supplied to the algorithms in the form of 2 (left),5 (middle), 10 (right) layers.

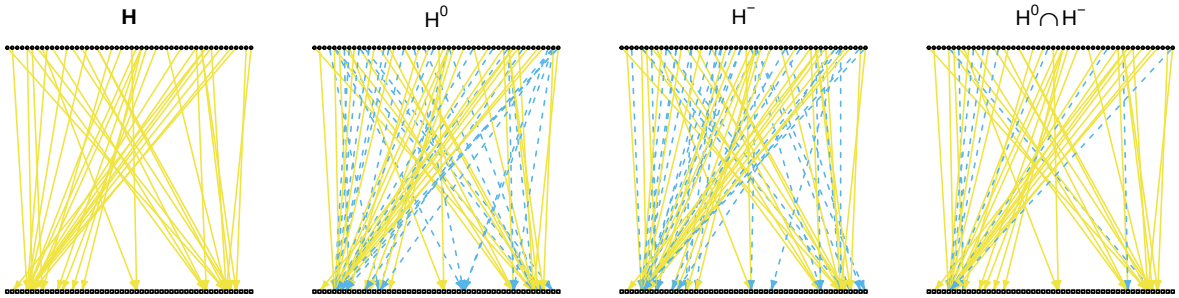


Figure 5.8: Estimated quantitative trait mappings for yeast.

5.7 Discussion

In this work we proposed a new view on the structure learning problem on partially ordered variables. We demonstrate that the two simple estimators, H^0 and H^{-j}, each may overestimate the truth but their intersection is usually close. Using a conditional independence search loop on the intersection can efficiently recover the edges between layers. In this work we limited our discussion to DAGs. However, Lemma 23 can be made more general. For example, similar results might apply to structure learning with latent and selection variables, in which case the target of learning is a mixed graph instead of a DAG.

Chapter 6

DISCUSSION AND FUTURE WORK

In this dissertation we presented four related problems involving structural equation models and structure learning in the context of high dimensional statistics.

We first proposed in Chapter 2 a simple method for causal discovery under a linear structural equation model with equal error variances. The procedure consistently estimates a topological ordering of the underlying graph and easily extends to the high-dimensional setting. Simulations demonstrate that the procedure is an attractive alternative to previously considered greedy search methods in terms of both accuracy and computational effort. In fact, it is shown recently that a slight modification of this algorithm achieves the minimax optimal sample complexity Gao et al. (2022). The empirical performance of our algorithm is also compared with other popular method in a recent paper (Yu et al., 2021).

In Chapter 3 we observe that many biological and physical system have unbounded maximum degrees and the traditional gold-standard approach for learning these networks could be inefficient. To facilitate causal structure discovery in such settings, our local FCI (lFCI) algorithm utilizes the local separation property of large (random) networks by considering an alternative local-graph-based search strategy focused on short paths between pairs of observed nodes. This idea applies naturally to linear Gaussian structural equation models (SEMs). We provided theoretical and empirical evidence for the advantage of our approach.

In Chapter 4 we introduced definite non-ancestral (DNA) relations as intermediate targets of inference in structure learning. DNA relations can be learned from simple conditional independencies and lead to computational and statistical gains in DAG structure learning. We proposed a framework for learning and using such information in several popular DAG learning schemes, including PC and Sparsest Permutation.

In Chapter 5 we considered the problem of structure learning when a partial ordering of the variable is known. We proposed an efficient algorithm based on a screening loop and a searching loop. The former narrow downs possible edges, and the latter removed additional edges. With suitable screening and searching methods, this method is proven to be consistent and efficient.

For future works related to Chapter 2, we note that our approach as well as the modified approach in Gao et al. (2022) rely on Best Subset Selection (BSS) and is hence computationally more demanding for graphs with higher in-degree. From a practical perspective, one may be able to improve the performance using computationally efficient approach to select parents given an estimated ordering. For example, in the loop to infer the causal ordering, we can use any unbiased regression estimator (not necessarily BSS). The regression coefficients can then be used to screen potential parents and reduce the search space for the pruning loop. We also note that similar algorithms can be applied to a broad set of problems, for example, estimating multiple DAGs with same ordering and error variance. This problem is covered in Ghoshal et al. (2021).

In Chapter 3, the proposed algorithm only relies on conditional independence tests, and can be, in principle, applied to a wider collection of models in which causal relations are well-characterized by local structures. Extending this idea to more general distributions, using, e.g., Gaussian copulas (Harris and Drton, 2013), or using conditional mutual information (Anandkumar et al., 2012b) can be fruitful directions of future research.

The ideas in Chapter 4 has been recently studied in the context of counterfactual analysis Zuo et al. (2022). One major challenge remained in our work is the statistical efficiency of the DNA-learning framework, which in practice requires a large amount of data to eliminate type-II errors. It could be fruitful to look at the DNA-learning problem with similar lens from Chapter 3 and Sondhi and Shojaie (2019) to improve the efficiency. Future works could also look at ancestral and non-ancestral relations in MAGs, which represent networks with unobserved latent and selection variables.

As for Chapter 5, currently the results are limited to DAGs, but we conjecture that

similar approaches can be applied to analyze networks with latent and selection variables, where the target of learning are mixed graphs.

BIBLIOGRAPHY

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

Anandkumar, A., Tan, V. Y. F., Huang, F., and Willsky, A. S. (2012a). High-dimensional Gaussian graphical model selection: walk summability and local separation criterion. *The Journal of Machine Learning Research*, 13(1):2293–2337.

Anandkumar, A., Tan, V. Y. F., Huang, F., and Willsky, A. S. (2012b). High-dimensional structure estimation in Ising models: Local separation criterion. *The Annals of Statistics*, 40(3):1346–1375. Publisher: Institute of Mathematical Statistics.

Andersson, S. A., Madigan, D., and Perlman, M. D. (1997). A Characterization of Markov Equivalence Classes for Acyclic Digraphs. *The Annals of Statistics*, 25(2):505–541. Publisher: Institute of Mathematical Statistics.

Andersson, S. A. and Perlman, M. D. (2006). Characterizing Markov Equivalence Classes for AMP Chain Graph Models. *The Annals of Statistics*, 34(2):939–972. Publisher: Institute of Mathematical Statistics.

Bollobás, B. (2001). *Random Graphs*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, 2 edition.

Brem, R. B. and Kruglyak, L. (2005). The landscape of genetic complexity across 5,700 gene expression traits in yeast. *Proceedings of the National Academy of Sciences*, 102(5):1572–1577. Publisher: Proceedings of the National Academy of Sciences.

Cai, T., Liu, W., and Luo, X. (2011). A Constrained l1 Minimization Approach to Sparse Pre-

cision Matrix Estimation. *Journal of the American Statistical Association*, 106(494):594–607. Publisher: Taylor & Francis.

Chakraborty, S. and Shojaie, A. (2022). Nonparametric Causal Structure Learning in High Dimensions. *Entropy*, 24(3):351. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.

Chen, H. and Sharp, B. M. (2004). Content-rich biological network constructed by mining PubMed abstracts. *BMC bioinformatics*, 5:147.

Chen, W., Drton, M., and Wang, Y. S. (2019). On causal discovery with an equal-variance assumption. *Biometrika*, 106(4):973–980.

Chickering, D. M. (1996). Learning Bayesian Networks is NP-Complete. In Fisher, D. and Lenz, H.-J., editors, *Learning from Data: Artificial Intelligence and Statistics V*, Lecture Notes in Statistics, pages 121–130. Springer, New York, NY.

Chickering, D. M. (2002). Optimal Structure Identification With Greedy Search. *Journal of Machine Learning Research*, 3(Nov):507–554.

Chung, F. and Lu, L. (2006). *Complex Graphs and Networks (Cbms Regional Conference Series in Mathematics)*. American Mathematical Society, USA.

Claassen, T. and Heskes, T. (2011). A logical characterization of constraint-based causal discovery. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, UAI'11, pages 135–144, Arlington, Virginia, USA. AUAI Press.

Claassen, T., Mooij, J. M., and Heskes, T. (2013). Learning sparse causal models is not NP-hard. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, pages 172–181, Arlington, Virginia, USA. AUAI Press.

Colombo, D. and Maathuis, M. H. (2014). Order-Independent Constraint-Based Causal Structure Learning. *Journal of Machine Learning Research*, 15(116):3921–3962.

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

Dembo, A. and Montanari, A. (2010). Ising models on locally tree-like graphs. *The Annals of Applied Probability*, 20(2):565–592. Publisher: Institute of Mathematical Statistics.

Dommers, S., Giardinà, C., and van der Hofstad, R. (2010). Ising Models on Power-Law Random Graphs. *Journal of Statistical Physics*, 141(4):638–660.

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

Draisma, J., Sullivant, S., and Talaska, K. (2013). Positivity for Gaussian graphical models. *Advances in Applied Mathematics*, 50(5):661–674.

Drton, M. (2018). Algebraic problems in structural equation modeling. *The 50th Anniversary of Gröbner Bases*, 77:35–87. Publisher: Mathematical Society of Japan.

Drton, M. and Maathuis, M. H. (2017). Structure Learning in Graphical Modeling. *Annual Review of Statistics and Its Application*, 4(1):365–393.

Drton, M. and Perlman, M. D. (2007). Multiple Testing and Error Control in Gaussian Graphical Model Selection. *Statistical Science*, 22(3):430–449. Publisher: Institute of Mathematical Statistics.

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

Entner, D., Hoyer, P., and Spirtes, P. (2013). Data-driven covariate selection for nonparametric estimation of causal effects. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, pages 256–264. PMLR. ISSN: 1938-7228.

Fan, J. and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911.

Foygel, R. and Drton, M. (2010). Extended Bayesian Information Criteria for Gaussian Graphical Models. In *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics (Oxford, England)*, 9(3):432–441.

Friedman, N. and Koller, D. (2003). Being Bayesian About Network Structure. A Bayesian Approach to Structure Discovery in Bayesian Networks. *Machine Learning*, 50(1):95–125.

Gao, M., Tai, W. M., and Aragam, B. (2022). Optimal estimation of Gaussian DAG models. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 8738–8757. PMLR. ISSN: 2640-3498.

Geer, S. v. d., Bühlmann, P., Ritov, Y., and Dezeure, R. (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics*, 42(3):1166–1202. Publisher: Institute of Mathematical Statistics.

Ghoshal, A., Bello, K., and Honorio, J. (2021). Direct Learning with Guarantees of the Difference DAG Between Structural Equation Models. *Preprint*.

Ghoshal, A. and Honorio, J. (2018). Learning linear structural equation models in polynomial time and sample complexity. In *International Conference on Artificial Intelligence and Statistics*, pages 1466–1475. PMLR. ISSN: 2640-3498.

Glymour, C., Scheines, R., Spirtes, P., and Kelly, K. (1987). *Discovering causal structure: Artificial intelligence, philosophy of science, and statistical modeling*. Discovering causal structure: Artificial intelligence, philosophy of science, and statistical modeling. Academic Press, San Diego, CA, US. Pages: xvii, 394.

Gupta, S. and Kim, H. W. (2008). Linking structural equation modeling to Bayesian networks: Decision support for customer retention in virtual communities. *European Journal of Operational Research*, 190(3):818–833.

Ha, M. J. and Sun, W. (2020). Estimation of high-dimensional directed acyclic graphs with surrogate intervention. *Biostatistics*, 21(4):659–675.

Haris, A., Simon, N., and Shojaie, A. (2019). Generalized Sparse Additive Models. *arXiv:1903.04641 [math, stat]*. arXiv: 1903.04641.

Haris, A., Simon, N., and Shojaie, A. (2022). Generalized Sparse Additive Models. *Journal of Machine Learning Research*, 23(70):1–56.

Harris, N. and Drton, M. (2013). PC Algorithm for Nonparanormal Graphical Models. *Journal of Machine Learning Research*, 14(69):3365–3383.

Heckerman, D. (1997). Bayesian Networks for Data Mining. *Data Mining and Knowledge Discovery*, 1(1):79–119.

Heinze-Deml, C., Maathuis, M. H., and Meinshausen, N. (2018). Causal Structure Learning. *Annual Review of Statistics and Its Application*, 5(1):371–391.

Hoyer, P., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. (2008). Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.

Ideker, T. and Krogan, N. J. (2012). Differential network biology. *Molecular Systems Biology*, 8(1):565.

Javanmard, A. and Montanari, A. (2014). Confidence Intervals and Hypothesis Testing for High-Dimensional Regression. *Journal of Machine Learning Research*, 15(82):2869–2909.

Javanmard, A. and Montanari, A. (2018). Debiasing the lasso: Optimal sample size for Gaussian designs. *The Annals of Statistics*, 46(6A):2593–2622. Publisher: Institute of Mathematical Statistics.

Kalisch, M. and Bühlmann, P. (2007). Estimating High-Dimensional Directed Acyclic Graphs with the PC-Algorithm. *Journal of Machine Learning Research*, 8(22):613–636.

Kalisch, M., Hauser, A., Maechler, M., Colombo, D., Entner, D., Hoyer, P., Hyttinen, A., Peters, J., Andri, N., Perkovic, E., Nandy, P., Ruetimann, P., Stekhoven, D., Schuerch, M., Eigenmann, M., Henckel, L., and Mooij, J. (2022). pcalg: Methods for Graphical Models and Causal Inference.

Kleinberg, J. M., Kumar, R., Raghavan, P., Rajagopalan, S., and Tomkins, A. S. (1999). The Web as a Graph: Measurements, Models, and Methods. In Asano, T., Imai, H., Lee, D. T., Nakano, S.-i., and Tokuyama, T., editors, *Computing and Combinatorics*, Lecture Notes in Computer Science, pages 1–17, Berlin, Heidelberg. Springer.

Kucukyavuz, S., Shojaie, A., Manzour, H., Wei, L., and Wu, H.-H. (2022). Consistent Second-Order Conic Integer Programming for Learning Bayesian Networks. *ArXiv*. Number: arXiv:2005.14346 arXiv:2005.14346 [cs, math, stat].

Lin, L., Drton, M., and Shojaie, A. (2016). Estimation of High-Dimensional Graphical Models Using Regularized Score Matching. *Electronic Journal of Statistics*, 10(1):806–854.

Liu, H., Han, F., Yuan, M., Lafferty, J., and Wasserman, L. (2012). High-dimensional semiparametric Gaussian copula graphical models. *The Annals of Statistics*, 40(4):2293–2326. Publisher: Institute of Mathematical Statistics.

Liu, H., Lafferty, J., and Wasserman, L. (2009). The Nonparanormal: Semiparametric Estimation of High Dimensional Undirected Graphs. *Journal of Machine Learning Research*, 10(80):2295–2328.

Liu, W. and Luo, X. (2015). Fast and adaptive sparse precision matrix estimation in high dimensions. *Journal of Multivariate Analysis*, 135:153–162.

Loh, P.-L. and Bühlmann, P. (2014). High-dimensional learning of linear causal networks via inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1):3065–3105.

Maathuis, M., Drton, M., Lauritzen, S., and Wainwright, M., editors (2018). *Handbook of Graphical Models*. CRC Press, Boca Raton.

Magliacane, S., Claassen, T., and Mooij, J. M. (2016). Ancestral causal inference. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 4473–4481, Red Hook, NY, USA. Curran Associates Inc.

Malioutov, D. M., Johnson, J. K., and Willsky, A. S. (2006). Walk-Sums and Belief Propagation in Gaussian Graphical Models. *Journal of Machine Learning Research*, 7(73):2031–2064.

Manzour, H., Küçükyavuz, S., Wu, H.-H., and Shojaie, A. (2021). Integer Programming for Learning Directed Acyclic Graphs from Continuous Data. *INFORMS Journal on Optimization*, 3(1):46–73. Publisher: INFORMS.

Markowetz, F. and Spang, R. (2007). Inferring cellular networks – a review. *BMC Bioinformatics*, 8(6):S5.

McKay, B. D., Wormald, N. C., and Wysocka, B. (2004). Short Cycles in Random Regular Graphs. *The Electronic Journal of Combinatorics*, pages R66–R66.

Meek, C. (1995). Causal inference and causal explanation with background knowledge. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, UAI'95, pages 403–410, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Miller, T. L. (2020). leaps: Regression Subset Selection. based on Fortran code by Alan.

Molloy, M. and Reed, B. (1995). A Critical Point for Random Graphs with a Given Degree Sequence. *Random Struct. Algorithms*.

Mooij, J., Janzing, D., Peters, J., and Schölkopf, B. (2009). Regression by dependence minimization and its application to causal inference in additive noise models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 745–752, New York, NY, USA. Association for Computing Machinery.

Nica, A. C. and Dermitzakis, E. T. (2013). Expression quantitative trait loci: present and future. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1620):20120362.

Ogarrio, J. M., Spirtes, P., and Ramsey, J. (2016). A Hybrid Causal Search Algorithm for Latent Variable Models. In *Proceedings of the Eighth International Conference on Probabilistic Graphical Models*, pages 368–379. PMLR. ISSN: 1938-7228.

Park, G. (2020). Identifiability of Additive Noise Models Using Conditional Variances. *Journal of Machine Learning Research*, 21(75):1–34.

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

Perkovic, E., Textor, J., Kalisch, M., and Maathuis, M. H. (2018). Complete Graphical Characterization and Construction of Adjustment Sets in Markov Equivalence Classes of Ancestral Graphs. *Journal of Machine Learning Research*, 18(220):1–62.

Peters, J. and Bühlmann, P. (2014). Identifiability of Gaussian structural equation models with equal error variances. *Biometrika*, 101(1):219–228.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2011). Identifiability of causal graphs using functional Models. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, UAI'11, pages 589–598, Arlington, Virginia, USA. AUAI Press.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal Discovery with Continuous Additive Noise Models. *Journal of Machine Learning Research*, 15(58):2009–2053.

Ramsey, J., Spirtes, P., and Zhang, J. (2006). Adjacency-faithfulness and conservative causal inference. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'06, pages 401–408, Arlington, Virginia, USA. AUAI Press.

Raskutti, G. and Uhler, C. (2018). Learning directed acyclic graph models based on sparsest permutations. *Stat*, 7(1):e183.

Ravikumar, P., Wainwright, M. J., Raskutti, G., and Yu, B. (2011). High-dimensional covariance estimation by minimizing l1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5(0):935–980.

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

Shimizu, S., Hoyer, P. O., Hyv&#228, A., rinen, and Kerminen, A. (2006). A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Journal of Machine Learning Research*, 7(72):2003–2030.

Shojaie, A. (2021). Differential network analysis: A statistical perspective. *WIREs Computational Statistics*, 13(2):e1508.

Shojaie, A., Jauhiainen, A., Kallitsis, M., and Michailidis, G. (2014). Inferring regulatory networks by combining perturbation screens and steady state gene expression profiles. *PloS One*, 9(2):e82393.

Shojaie, A. and Michailidis, G. (2010). Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs. *Biometrika*, 97(3):519–538.

Solus, L., Wang, Y., and Uhler, C. (2021). Consistency guarantees for greedy permutation-based causal inference algorithms. *Biometrika*, 108(4):795–814.

Sondhi, A. and Shojaie, A. (2019). The Reduced PC-Algorithm: Improved Causal Structure Learning in Large Random Networks. *Journal of Machine Learning Research*, 20(164):1–31.

Spirtes, P., Glymour, C., and Scheines, R. (2001). *Causation, Prediction, and Search, 2nd Edition*, volume 1. The MIT Press, 1 edition.

Squires, C., Amaniampong, J., and Uhler, C. (2020). Efficient Permutation Discovery in Causal DAGs. *arXiv:2011.03610 [cs, stat]*. arXiv: 2011.03610 version: 1.

Stark, C., Breitkreutz, B.-J., Reguly, T., Boucher, L., Breitkreutz, A., and Tyers, M. (2006). BioGRID: a general repository for interaction datasets. *Nucleic Acids Research*, 34(Database issue):D535–D539.

Sullivant, S., Talaska, K., and Draisma, J. (2010). Trek separation for Gaussian graphical models. *The Annals of Statistics*, 38(3):1665–1685. Publisher: Institute of Mathematical Statistics.

Tan, Z. and Zhang, C.-H. (2019). Doubly penalized estimation in additive regression with high-dimensional data. *The Annals of Statistics*, 47(5):2567–2600. Publisher: Institute of Mathematical Statistics.

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

Uhler, C., Raskutti, G., Bühlmann, P., and Yu, B. (2013). Geometry of the faithfulness

assumption in causal inference. *Annals of Statistics*, 41(2):436–463. Publisher: Institute of Mathematical Statistics.

Voorman, A., Shojaie, A., and Witten, D. (2014). Graph estimation with joint additive models. *Biometrika*, 101(1):85–101.

Wainwright, M. J. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge.

Wang, P.-L. and Michailidis, G. (2019). Directed Acyclic Graph Reconstruction Leveraging Prior Partial Ordering Information. In Nicosia, G., Pardalos, P., Umeton, R., Giuffrida, G., and Sciacca, V., editors, *Machine Learning, Optimization, and Data Science*, Lecture Notes in Computer Science, pages 458–471, Cham. Springer International Publishing.

Wang, Y. S. and Drton, M. (2020). High-dimensional causal discovery under non-Gaussianity. *Biometrika*, 107(1):41–59.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442.

Xue, L. and Zou, H. (2012). Regularized rank-based estimation of high-dimensional nonparanormal graphical models. *The Annals of Statistics*, 40(5):2541–2571. Publisher: Institute of Mathematical Statistics.

Yin, J. and Li, H. (2011). A sparse conditional Gaussian graphical model for analysis of genetical genomics data. *The Annals of Applied Statistics*, 5(4):2630–2650. Publisher: Institute of Mathematical Statistics.

Yu, G. and Bien, J. (2019). Estimating the error variance in a high-dimensional linear model. *Biometrika*, 106(3):533–546.

Yu, S., Drton, M., and Shojaie, A. (2019). Generalized Score Matching for Non-Negative Data. *Journal of Machine Learning Research*, 20(76):1–70.

Yu, S., Drton, M., and Shojaie, A. (2020). Directed Graphical Models and Causal Discovery for Zero-Inflated Data. *arXiv.2004.04150*. arXiv:2004.04150 [stat].

Yu, Y., Gao, T., Yin, N., and Ji, Q. (2021). DAGs with No Curl: An Efficient DAG Structure Learning Approach. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12156–12166. PMLR. ISSN: 2640-3498.

Zander, B. v. d. and Liśkiewicz, M. (2020). Finding Minimal d-separators in Linear Time and Applications. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, pages 637–647. PMLR. ISSN: 2640-3498.

Zhang, B., Gaiteri, C., Bodea, L.-G., Wang, Z., McElwee, J., Podtelezhnikov, A. A., Zhang, C., Xie, T., Tran, L., Dobrin, R., Fluder, E., Clurman, B., Melquist, S., Narayanan, M., Suver, C., Shah, H., Mahajan, M., Gillis, T., Mysore, J., MacDonald, M. E., Lamb, J. R., Bennett, D. A., Molony, C., Stone, D. J., Gudnason, V., Myers, A. J., Schadt, E. E., Neumann, H., Zhu, J., and Emilsson, V. (2013). Integrated Systems Approach Identifies Genetic Nodes and Networks in Late-Onset Alzheimer's Disease. *Cell*, 153(3):707–720. Publisher: Elsevier.

Zhang, C.-H. and Zhang, S. S. (2014). Confidence intervals for low dimensional parameters in high dimensional linear models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 76(1):217–242. Publisher: [Royal Statistical Society, Wiley].

Zhang, J. (2008). On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172(16-17):1873–1896.

Zhang, J. and Spirtes, P. (2002). Strong faithfulness and uniform consistency in causal inference. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, UAI'03, pages 632–639, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Zhang, K. and Hyvärinen, A. (2009a). Causality Discovery with Additive Disturbances: An Information-Theoretical Perspective. In Buntine, W., Grobelnik, M., Mladenić, D., and Shawe-Taylor, J., editors, *Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 570–585, Berlin, Heidelberg. Springer.

Zhang, K. and Hyvärinen, A. (2009b). On the identifiability of the post-nonlinear causal model. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 647–655, Arlington, Virginia, USA. AUAI Press.

Zhang, K., Peters, J., Janzing, D., and Schölkopf, B. (2011). Kernel-based conditional independence test and application in causal discovery. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, UAI'11, pages 804–813, Arlington, Virginia, USA. AUAI Press.

Zheng, X., Aragam, B., Ravikumar, P., and Xing, E. P. (2018). DAGs with NO TEARS: continuous optimization for structure learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 9492–9503, Red Hook, NY, USA. Curran Associates Inc.

Zuo, A., Wei, S., Liu, T., Han, B., Zhang, K., and Gong, M. (2022). Counterfactual Fairness with Partially Known Causal Graph. *Preprint*.

Appendix A

APPENDICES TO CHAPTER 2

A.1 Proof of Theorem 2

We first give a lemma that addresses the estimation error for inverse covariances.

Lemma 24. *Assume* $X \sim (B, \sigma^2, \gamma)$. *Suppose all* $(q+1)\times(q+1)$ *principal submatrices of* $\Sigma = \mathbb{E}\left[XX^T\right]$ *have minimum eigenvalue at least* $\lambda_{\min} > 0$. *If for* $\epsilon, \eta > 0$ *we have*

$$n \;\geq\; (q+1)^2\left\{\log(p^2+p) - \log(\epsilon/2)\right\} 128\left(1+4\frac{\gamma^2}{\sigma^2}\right)^2\left(\max_{j\in V}\Sigma_{j,j}\right)^2\left(\frac{\eta\lambda_{\min}+1}{\eta\lambda_{\min}^2}\right)^2. \quad \text{(A.1)}$$

then

$$\max_{C\subseteq V, |C|\leq q+1} \|(\Sigma_{C,C})^{-1} - (\hat{\Sigma}_{C,C})^{-1}\|\infty \leq \eta$$

with probability at least $1-\epsilon$.

Proof. Let $\delta = \frac{\eta\lambda_{\min}^2}{(q+1)(\eta\lambda_{\min}+1)}$. Because $\delta < \frac{\lambda_{\min}}{q+1}$, by Lemma 5 from Harris and Drton (2013), we have

$$\max_{C\subseteq V, |C|\leq (q+1)} \|(\Sigma_{C,C})^{-1} - (\hat{\Sigma}_{C,C})^{-1}\|_\infty \;\leq\; \frac{(q+1)\delta/\lambda_{\min}^2}{1-(q+1)\delta/\lambda_{\min}} \;=\; \eta$$

provided $\|\widehat{\Sigma}-\Sigma\|_\infty \leq \delta$. The proof is thus complete if we show that $\mathbb{P}\left\{\|\widehat{\Sigma}-\Sigma\|_\infty > \delta\right\} \leq \epsilon$.

Note that $X_j = \varepsilon_j + \sum_{k\in \mathrm{an}(j)} \pi_{jk}\varepsilon_k$ has variance $\sigma^2(1+\sum_{k\in\mathrm{an}(j)}\pi_{jk}^2)$. Since γ is a bound on the sub-Gaussian parameters of all ϵ_l, it follows that $X_j/\sqrt{\mathrm{Var}\,(()\,X_j)}$ is sub-Gaussian with parameter at most γ/σ. Lemma 1 of Ravikumar et al. (2011) applies and gives

$$\mathbb{P}\{|\widehat{\Sigma}_{i,j}-\Sigma_{i,j}| > \delta\} \leq 4\exp\left\{-\frac{n\delta^2}{128(1+4\gamma^2/\sigma^2)^2\max_j(\Sigma_{j,j})^2}\right\} \leq \frac{2}{p(p+1)}\epsilon.$$

A union bound over the entries of Σ yields that indeed $\mathbb{P}\left\{\|\widehat{\Sigma}-\Sigma\|_\infty > \delta\right\} \leq \epsilon$. □

Proof of Theorem 2. Our assumption on n is as in (A.1) with $\eta = \zeta/(2\sigma^2)$. Lemma 24 thus implies that, with probability at least $1-\epsilon$, we have for all subsets $\Theta \subseteq V$ with $|\Theta| < q+1$ that

$$\|(\hat{\Sigma}_{\Theta,\Theta})^{-1} - (\Sigma_{\Theta,\Theta})^{-1}\|_\infty \leq \frac{\zeta}{2\sigma^2}. \tag{A.2}$$

Let j be a source in $G(B)$, and let k be a non-source. Note that variance of j conditional on some set C_1 is

$$\sigma^2_{j|C_1} = \frac{1}{\left\{(\Sigma_{C_1\cup\{j\},C_1\cup\{j\}})^{-1}\right\}_{j,j}}.$$

By Lemma 5, for any $C_1, C_2 \subseteq \Theta \subseteq V \setminus \{j,k\}$ such that Θ is an ancestral set and $\mathrm{pa}(j) \subseteq C_1$

$$\left\{(\Sigma_{C_1\cup\{j\},C_1\cup\{j\}})^{-1}\right\}_{j,j} - \left\{(\Sigma_{C_2\cup\{k\},C_2\cup\{k\}})^{-1}\right\}_{k,k} \geq \frac{1}{\sigma^2} - \frac{1}{\sigma^2(1+\zeta)} \geq \frac{\zeta}{\sigma^2} \tag{A.3}$$

Using (A.2), when $|C_1|$ and $|C_2|$ are both at most q, we obtain that

$$\left\{(\hat{\Sigma}_{C_1\cup\{j\},C_1\cup\{j\}})^{-1}\right\}_{j,j} - \left\{(\hat{\Sigma}_{C_2\cup\{k\},C_2\cup\{k\}})^{-1}\right\}_{k,k} - \frac{\zeta}{\sigma^2} \;>\; 0. \tag{A.4}$$

Thus $\hat{\sigma}^2_{j|C_1} - \hat{\sigma}^2_{k|C_2} > 0$ which implies that Algorithm 1 correctly selects a source node at each step. On the first step, $\Theta = \emptyset$ which is trivially an ancestral set. By induction, each subsequent step then correctly adds a sink to Θ so Θ remains ancestral and a correct ordering is recovered. □

Now to show consistency of the high-dimensional bottom-up algorithm. We first state Theorem 9 from Yu and Bien (2019).

Theorem 17. *Suppose each column of X_j of the matrix $X \in \mathbb{R}^{n\times p}$ has been scaled so that $\|X_j\|_2^2 = n$ for all $j = 1,\ldots,p$, and $\epsilon \sim N(0,\sigma_0^2 I_n)$. Then, for any constant $M > 1$, the organic lasso estimate,*

$$\hat{\sigma}_\lambda = \min_{\beta\in\mathbb{R}^p}\left(\frac{1}{n}\|y - X\beta\|_2^2 + 2\lambda\|\beta\|_1^2\right),$$

with $\lambda = (2Mn^{-1}\log p)^{1/2}$ satisfies the following relative mean squared error bound

$$E\left\{\left(\frac{\hat{\sigma}^2_\lambda}{\sigma_0^2} - 1\right)^2\right\} \;\leq\; \left\{\left(8M + \frac{p^{1-8M}}{\log p}\right)\max\left(\frac{\|\beta^*\|_1^2}{\sigma_0^2}, \frac{\|\beta^*\|_1}{\sigma_0} + \frac{1}{4}\right)\left(\frac{\log p}{n}\right)^{1/2} + \left(\frac{2}{n}\right)^{1/2}\right\}^2.$$

proof of Theorem 4. When estimating $\sigma^2_{j,\Theta,\lambda}$, we regress j onto $V \setminus \{\Theta, j\}$. So in our setting,

$$\beta^*_j = \left(\Sigma_{V\setminus\{\Theta,j\},V\setminus\{\Theta,j\}}\right)^{-1}\Sigma_{V\setminus\{\Theta,j\},j},$$

where $\beta^*_j = 0$ for all j not in the Markov blanket (parents, children, and parents of children) of j in the sub-graph induced by $V \setminus \Theta$. For any j and $\Theta \subset V$,

$$\begin{aligned}\|\beta^*\|_1 &= \|\left(\Sigma_{V\setminus\{\Theta,j\},V\setminus\{\Theta,j\}}\right)^{-1}\Sigma_{V\setminus\{\Theta,j\},j}\|_1 \\ &\leq \|\left(\Sigma_{V\setminus\{\Theta,j\},V\setminus\{\Theta,j\}}\right)^{-1}\|_1\|\Sigma_{V\setminus\{\Theta,j\},j}\|_1 \\ &\leq \|\left(\Sigma_{V\setminus\{j\},V\setminus\{j\}}\right)^{-1}\|_1\|\Sigma_{V\setminus\{j\},j}\|_1.\end{aligned} \tag{A.5}$$

So A provides a uniform bound over the relevant quantity for each Θ. For the values of n and λ we specified, the above theorem suggests $E\left\{\left(\frac{\hat{\sigma}^2_\lambda}{\sigma^2_0} - 1\right)^2\right\} \leq \zeta^2\epsilon$. Then by Markov inequality, for any subset $\Theta \subseteq V$,

$$\mathbb{P}\left\{(\hat{\sigma}^2_{j,\Theta,\lambda} - \mathrm{Var}\left(X_j \mid X_{V\setminus(\Theta\cup\{j\})}\right)^2)^2 > \sigma^4\zeta^2\right\} \leq \epsilon,$$

Let j be a sink in $G(B)$ and let k be a non-sink, then,

$$\mathrm{Var}\left(X_j \mid X_{V\setminus(\Theta\cup\{j\})}\right)^2 - \mathrm{Var}\left(X_k \mid X_{V\setminus(\Theta\cup\{k\})}\right)^2 = \sigma^2 - 1/(\Sigma^{-1})_{k,k} = \sigma^2\sum_{q\in\mathrm{ch}(k)}\beta^2_{kq} > \zeta\sigma^2.$$

Hence, we obtain that

$$\hat{\sigma}^2_{j,\Theta,\lambda} - \hat{\sigma}^2_{k,\Theta,\lambda} \geq \mathrm{Var}\left(X_j \mid X_{V\setminus(\Theta\cup\{j\})}\right)^2 - \mathrm{Var}\left(X_k \mid X_{V\setminus(\Theta\cup\{k\})}\right)^2 - \sigma^2\zeta > 0.$$

This implies that in the first step the bottom-up algorithm correctly selects a sink node. By Lemma 4, we may repeat the argument just given in each step, and the bottom-up algorithm correctly estimates a reversed topological ordering of G. □

A.2 Simulations as in Peters and Bühlmann (2014)

We revisit the simulation study of Peters and Bühlmann (2014). DAGs are generated by first creating a random topological ordering, then between any two nodes, an edge is included with probability p_c. We simulate a sparse setting with $p_c = 3/(2p-2)$ and a dense setting with

$p_c = 0.3$. The linear coefficients are drawn uniformly from $[-1, -.1] \cup [.1, 1]$ and the errors are drawn from a standard Gaussian distribution. Since there may not be a unique ordering for the true graph, we compute the Hamming distance between the true and estimated adjacency matrix rather than Kendall's τ.

Tables A.1 and A.2 demonstrate that in both settings, the greedy algorithm performs better when p is small. However, when $p = 40$ the proposed algorithms infer the graph more accurately. In the dense setting, the proposed methods have similar FDR to greedy search, but substantially higher recall. In the sparse setting, the proposed methods have lower recall than greedy search, but also substantially lower FDR.

Table A.1: Dense setting

		Hamming Dist.			Recall %			Flipped %			FDR %		
p	n	TD	BU	GDS	TD	BU	GDS	TD	BU	GDS	TD	BU	GDS
5	100	1.3	1.3	1.1	73	73	78	7	7	7	16	15	18
	500	0.7	0.7	0.5	80	80	88	4	4	5	8	7	9
	1000	0.5	0.5	0.4	85	84	92	3	3	5	5	5	7
20	100	31	32	30	73	73	74	4	3	6	27	28	25
	500	22	22	14	91	91	91	2	3	4	24	24	13
	1000	28	28	8	94	94	96	2	2	2	21	21	10
40	100	170	174	215	66	65	54	2	3	8	36	37	45
	500	152	155	186	93	93	76	2	2	9	38	39	42
	1000	136	137	168	96	95	83	1	1	8	36	36	38

A.3 Simulations as in Ghoshal and Honorio (2018)

We construct random graphs as in Section 2.4.2, but we follow the data sampling procedure as used in Ghoshal and Honorio (2018). All linear coefficients are drawn uniformly from

Table A.2: Sparse setting

		Hamming Dist.			Recall %			Flipped %			FDR %		
p	n	TD	BU	GDS	TD	BU	GDS	TD	BU	GDS	TD	BU	GDS
5	100	1.6	1.7	1.4	74	73	78	8	8	8	18	18	17
	500	0.8	0.9	0.6	85	84	91	3	4	5	7	7	9
	1000	0.6	0.6	0.4	88	88	94	3	4	5	6	6	7
20	100	7	7	12	69	69	81	4	4	6	16	17	43
	500	3.5	3.5	4.5	85	84	93	4	4	4	9	8	21
	1000	2.2	2.2	2.8	90	90	97	3	2	3	5	5	14
40	100	14	15	45	64	63	78	3	4	8	16	18	62
	500	7	7	16	84	84	94	3	3	3	8	7	33
	1000	5	5	10	90	89	97	3	3	3	6	6	24

$\pm[.5, 1]$, and errors are drown from the Rademacher distribution and scaled to have $\sigma_i^2 = 0.8$. Table A.3 demonstrates that both methods performs reasonably well when Markov blankets are restricted to be small, and the top-down approach performs substantially better when there are hubs.

A.4 Simulations of fully connected graphs

We run simulations with fully connected graphs, as suggested by a reviewer. The linear coefficients are drawn uniformly from $\pm[.3, 1]$ and the errors are drawn from a standard Gaussian distribution. The results confirm the advantages of the proposed methods and are shown in Table A.4. In general, the estimated graphs from the top-down and bottom-up procedure differ only slightly, and the values reported in the table differ in the 3rd or 4th digit.

Table A.3: High-dimensional setting with Rademacher noise and maximum in-degree $q = 3$

		Small k		Hub graph	
n	p	HTD	HBU	HTD	HBU
80	$0.5n$	0.99	0.95	0.98	0.73
	$0.75n$	0.98	0.90	0.89	0.46
	n	0.96	0.90	0.76	0.36
	$1.5n$	0.84	0.86	0.52	0.23
	$2n$	0.71	0.80	0.35	0.10
100	$0.5n$	0.99	0.97	0.99	0.69
	$0.75n$	0.99	0.95	0.92	0.46
	n	0.96	0.93	0.76	0.34
	$1.5n$	0.84	0.88	0.52	0.26
	$2n$	0.72	0.82	0.39	0.13
200	$0.5n$	1.00	0.99	1.00	0.79
	$0.75n$	1.00	0.98	0.98	0.59
	n	0.98	0.97	0.86	0.47
	$1.5n$	0.86	0.84	0.61	0.20
	$2n$	0.73	0.77	0.48	0.10

A.5 As initializer for greedy search

As suggested by a reviewer, we explore the performance of the greedy DAG search (GDS) algorithm initialized with the estimates from the proposed procedures. We run simulations with the same data as in Section 2.4.1. Tables A.5 and A.6 show averages over 500 random realizations for the top-down procedure (TD), the greedy DAG search with random initialization (GR), and the greedy DAG search with warm initialization (GW). The GR procedure is

Table A.4: Fully connected setting

		Kendall's τ			Recall %			Flipped %			FDR %		
p	n	TD	BU	GDS	TD	BU	GDS	TD	BU	GDS	TD	BU	GDS
5	100	0.92	0.93	0.83	91	92	80	4	3	7	4	4	9
	500	0.99	0.99	0.97	98	98	98	1	1	1	1	1	1
	1000	1.00	1.00	0.99	99	100	99	0	0	1	0	0	1
20	100	0.98	0.98	0.62	74	74	45	1	1	9	1	1	17
	500	1.00	1.00	0.73	90	90	66	0	0	8	0	0	12
	1000	1.00	1.00	0.81	92	92	76	0	0	7	0	0	8
40	100	0.99	0.99	0.55	42	42	33	0	0	7	1	1	17
	500	1.00	1.00	0.62	50	50	49	0	0	8	0	0	14
	1000	1.00	1.00	0.67	52	52	59	0	0	8	0	0	12

identical to the GDS procedure described in Section 2.4.1 and Peters and Bühlmann (2014). In the GW procedure, we initialize with the output from the top-down method, then search through a large number of graph neighbors ($k = 300$) at each greedy step. Since the GW procedure is supplied with a good initializer, we do not restart the greedy search after it terminates, while 5 random restarting with $k = p, 2p, 3p, 5p, 300$ is used in GR to insure performance. For simplicity, we omitted the experiment with the bottom-up procedure (BU).

Tables A.5 and A.6 shows that in all the settings, GW performs better than the other two methods, especially when p is large. For reference, the average run time in the dense setting with $p = 40$ and $n = 1000$ is 8 seconds for the top-down method, 4,500 seconds for GR, and 400 seconds for GW.

Table A.5: Low-dimensional dense settings

		Kendall's τ			Recall %			Flipped %			FDR %		
p	n	TD	GR	GW	TD	GR	GW	TD	GR	GW	TD	GR	GW
5	100	0.85	0.88	0.88	91	91	91	7	6	6	17	9	10
	500	0.98	0.98	0.99	99	99	99	1	1	1	4	2	2
	1000	0.99	0.99	0.99	99	99	99	1	1	1	3	1	1
20	100	0.92	0.61	0.94	85	62	90	3	13	3	32	43	15
	500	0.99	0.75	0.99	99	81	99	1	11	0	28	35	3
	1000	1.00	0.82	1.00	100	88	100	0	8	0	26	28	2
40	100	0.96	0.53	0.96	71	44	84	2	11	2	41	58	20
	500	0.99	0.59	1.00	96	63	100	0	14	0	41	57	4
	1000	1.00	0.64	1.00	97	71	100	0	14	0	40	57	2

Table A.6: Low-dimensional sparse settings

		Kendall's τ			Recall %			Flipped %			FDR %		
p	n	TD	GR	GW	TD	GR	GW	TD	GR	GW	TD	GR	GW
5	100	0.87	0.88	0.87	91	90	91	6	6	6	16	9	10
	500	0.98	0.98	0.98	98	99	99	1	1	1	5	2	2
	1000	0.99	0.99	0.99	99	99	99	1	1	1	3	1	1
20	100	0.77	0.60	0.82	85	77	90	9	15	7	35	39	25
	500	0.96	0.77	0.98	98	89	99	2	10	1	19	26	8
	1000	0.99	0.81	0.99	100	90	100	0	9	0	14	23	4
40	100	0.72	0.47	0.79	81	72	89	10	20	7	38	54	36
	500	0.96	0.58	0.98	98	81	99	2	18	1	24	47	13
	1000	0.99	0.61	0.99	99	82	100	1	17	0	17	48	8

Appendix B

APPENDICES TO CHAPTER 3

B.1 Proofs

Proof of Lemma 7. We prove the case of $\eta = 3$ by enumerating all possible configurations of $G_\gamma(i,j)$ in the extended neighborhood of i and constructing a small m-separator for each one. Since G is a MAG, i and j cannot be ancestors of each other. Without loss of generality, we suppose $i \notin \text{an}(G, j)$. We use the following three facts.

1. Let $D_{G_\gamma(i,j)}(u,v)$ be the shortest path distance in the local graph, and let $N_k = \{v \in V_\gamma(i,j) : D_{G_\gamma(i,j)}(v,i) = k\}$ for each $k = 1, 2, \ldots, \gamma$. Then for each $k \leq \gamma$, there are at most η paths from i to N_k. Hence, $|\text{ne}(G_\gamma(i,j), u)| \leq \eta$.

2. Consider the two set of edges $e_k^b = \{(u,v) : u \in \text{adj}(G_\gamma(i,j), v), u \in N_k, v \in N_{k+1}\}$ and $e_k^m = \{(u,v) : u \in \text{adj}(G_\gamma(i,j), v), u, v \in N_k\}$. For each $k \leq \gamma$, every node in N_k has at least one edge in e^b or e^m (it cannot be a "deadend"), and $|e_k^b| + |e_k^m| \leq \eta$.

3. In the MAG G, if (u,v) are non-adjacent, then there is no inducing path between them, i.e., no path on which every node is a collider and ancestor of u or v.

The details are given in the Supplementary Material. See FigureB.1,B.2 and B.3. □

Proof of Lemma 9. By Lemma 1, it suffices to consider two nodes i and j that are non-adjacent in G and show that $V_\gamma(i,j) \backslash \{i,j\} \subseteq J_\gamma(i,j,C)$. By definition, any $v \in V_\gamma(i,j) \backslash \{i,j\}$ lies on a short path between i, j, so $D_G(i,v) + D_G(j,v) \leq \gamma$. Since i, j are not adjacent, we know $C = C_{-ij}$ is a supergraph of G. Hence, $D_{C_{-ij}}(i,v) \leq D_G(i,v)$ and $D_{C_{-ij}}(j,v) \leq D_G(j,v)$, which implies $D_{C_{-ij}}(i,v) + D_{C_{-ij}}(j,v) \leq \gamma$, and $v \in V_\gamma(i,j) \setminus \{i,j\} \subseteq J_\gamma(i,j,C)$. □

To prove Theorem 6, we first show an error bound for sample partial correlations.

Lemma 25. *Assume* $W = (W_1, \ldots, W_p)$ *is a zero-mean random vector with covariance matrix* Σ *such that each* $W_i/\Sigma_{ii}^{1/2}$ *is sub-Gaussian with parameter* σ*. Assume* σ *is bounded, and the minimal eigenvalue of all* $(\eta+2)\times(\eta+2)$ *submatrices of* Σ *are bounded below by* $\lambda_{\min} > 0$*. If for* $\zeta > 0$ *and* $\epsilon > 0$ *satisfying* $\epsilon < 16(\eta+2)\lambda_{\min}^{-2}\max_i(\Sigma_{ii})(1+4\sigma^2))$ *we have*

$$n \geq \{\log(p^2+p) - \log(2\zeta)\}128(1+4\sigma^2)^2 \max_i(\Sigma_{ii})^2(\eta+2)^2\left(\lambda_{\min}^{-1} + \lambda_{\min}^{-2}(1+2/\epsilon)\right)^2, \quad \text{(B.1)}$$

then the empirical partial correlation obtained from n *samples satisfies*

$$\mathbb{P}\left\{\max_{i\neq j, |S|\leq\eta}\left|\rho(i,j|S) - \widehat{\rho}(i,j|S)\right| \geq \epsilon\right\} \leq \zeta. \quad \text{(B.2)}$$

Proof. Set $\delta = \epsilon\lambda_{\min}^2/[(\epsilon+\epsilon\lambda_{\min}+2)(\eta+2)]$. Our choice of δ satisfies $\delta \in (0, 8\max_i(\Sigma_{ii})(1+4\sigma^2))$. Then by Lemma 1 in Ravikumar et al. (2011), we obtain the following inequality,

$$\mathbb{P}\left(|\widehat{\Sigma}_n(i,j) - \Sigma(i,j)| > \delta\right) \leq 4\exp\left\{-\frac{n\delta^2}{128(1+4\sigma^2)^2\max_i(\Sigma_{ii})^2}\right\}$$

With the stated n, we have $\mathbb{P}\left(|\widehat{\Sigma}_n(i,j) - \Sigma(i,j)| > \delta\right) \leq 2\zeta/(p^2+p)$. A union bound over all the entries yields $\mathbb{P}\left(\|\widehat{\Sigma}_n - \Sigma\|_\infty > \delta\right) \leq \zeta$. By Lemma 4 of Harris and Drton (2013), for all i, j and $|S| \leq \eta$, $\|\widehat{\Sigma}_n - \Sigma\|_\infty \leq \delta$ implies $|\rho(i,j|S) - \widehat{\rho}(i,j|S)| < \epsilon$. Therefore (B.2) holds. □

Proof of Theorem 6. First suppose we determine conditional independence by thresholding on $|\widehat{\rho}(i,j|S)|$. For λ from Assumption 4, define the event,

$$A = \left[\max_{i\neq j, |S|\leq\eta}\left|\rho(i,j|S) - \widehat{\rho}(i,j|S)\right| \leq \lambda/4\right].$$

By Lemma 25, for any $\zeta > 0$, with $n = \Omega((\log p)^{1/(1-2c)})$ we have $\mathbb{P}\{A\} > 1-\zeta$. Given A, it holds that for all i, j and $|S| \leq \eta$, $|\widehat{\rho}(i,j|S)| > \frac{3}{4}\lambda$ if and only if $|\rho(i,j|S)| > \lambda$. Under Assumption 5, all non-adjacent pairs i, j have some local- or full-graph- separator such that $|\rho(i,j|S)| < \lambda/2$. Therefore, with high probability, all the conditional independence decisions are correct and the output is the true skeleton. The orientation result follows Lemma 10.

Next, suppose we make conditional independence decisions by comparing z-transformed partial correlations to normal quantiles. It is shown in Appendix A of Harris and Drton (2013) that this approach is equivalent to the thresholding rule with significance levels $\alpha_n = 2\left(1 - \Phi\left(0.5\sqrt{n-3}\log\left(\frac{1+\lambda/3}{1-\lambda/3}\right)\right)\right)$. □

Proof of Corollary 1. Under the assumptions, there are always small neighborhood-based separators between non-adjacent nodes (see the proof for Lemma 6), and therefore the approximate-rPC, like rPC, consistently recovers correct skeletons. The orientation results are correct since only $\mathcal{R}_{0-3}$ are applied. □

B.2 Theoretical Guarantee for Algorithm 4

We next show theoretical guarantees for Algorithm 4 with estimated Markov blanket.

Theorem 18. *Under Assumptions 2, 4, 3, 5, and suppose* $n = \Omega((\log p)^{1/(1-2c)})$. *Suppose* γ *is large enough such that* $\mathrm{mb}_\gamma(G, v) = \mathrm{mb}(G, v)$ *for all* $v \in V$. *Suppose there exists a sequence* $\tau_{n,p} \to 0$ *such that the estimated precision matrix satisfies* $\|\widehat{\Theta} - \Theta\|_\infty \leq \tau_{n,p}$ *with high probability. Also assume* $\min_{i \in \mathrm{adj}(G,j)} |\Theta_{ij}| \geq 2\tau_{n,p}$. *Then there exists a sequence* $\alpha_n \to 0$ *such that Algorithm 4 consistently learns a PAG for* $[G]$. *Moreover, if Assumption 1 holds, then it consistently learns the maximally informative PAG.*

Remark 3. *To estimate the precision matrix, we can use in particular generalized score matching (Lin et al., 2016; Yu and Bien, 2019) or equivalently the SCIO algorithm (Liu and Luo, 2015), which satisfies* $\|\widehat{\Theta} - \Theta\|_\infty = O_p\left(\sqrt{s_p \log(p)/n}\right)$, *where* $s_p = \max_{i \in V} |\mathrm{mb}_\gamma(G, i)|$.

Proof of Theorem 18. Under stated assumptions on precision and beta-min, with high probability, $\{(i,j) : i \in \mathrm{adj}(G,j)\} \subseteq \left\{(i,j) : (i,j) \in \mathrm{supp}(\widehat{\Theta})\right\} \subseteq \left\{(i,j) : i \in \mathrm{mb}_\gamma(G,j)\right\}$. The rest of this proof is identical to Theorem 6, with η replaced by $\eta - 1$, following Lemma 8. □

Proof of Lemma 7. 1. $|\mathrm{ne}(G_\gamma(i,j), i)| = 1$. (Figure B.1) Denote the only neighbor as u. If $i \to u$ or $i \leftrightarrow u$ but $u \notin \mathrm{an}(G_\gamma(i,j), j)$, then $S_\gamma(i,j) = \emptyset$. If $i \leftarrow u$ or $i - u$, then $S_\gamma(i,j) = \{u\}$. If $i \leftrightarrow u$ and $u \in \mathrm{an}(G_\gamma(i,j), j)$, then there must be an edge $u \to w$

and $w \in \text{an}(G_\gamma(i,j),j)$. If $\text{ne}(G_\gamma(i,j),u) = \{i,w\}$ then $S_\gamma(i,j) = \{u\}$. Now discuss cases with additional neighbors.

(a) If $\text{ne}(G_\gamma(i,j),u) = \{i,w,v\}$, we have $S_\gamma(i,j) = \{u\}$ if $u \to v$ or $u \leftrightarrow v$ but $v \notin \text{an}(G_\gamma(i,j),j)$; also $S_\gamma(i,j) = \{u,v\}$ if $u \leftarrow v$ or $u \leftrightarrow v$ and $v \in \text{an}(G_\gamma(i,j),j)$ and v has exactly one neighbor other than u (it is allowed to be w). If v has 2 neighbors other than u, and neither is child of v, then $v \notin \text{an}(G_\gamma(i,j),j)$, which is covered in the previous case. Now suppose $v \to x$ and there is also an edge $v \star\!\!\star y$, in which case $|e_2^b| + |e_2^m| = 3$. We have $S_\gamma(i,j) = \{u,v,y\}$ if $v \leftarrow y$ or $v \leftrightarrow y$ and $y \in \text{an}(G_\gamma(i,j),j)$, and otherwise $S_\gamma(i,j) = \{u,v\}$.

(b) If $\text{ne}(G_\gamma(i,j),u) = \{i,w,v,x\}$, then $|e_2^b| = 3$ and $S_\gamma(i,j) \subseteq \{u,v,x\}$.

2. $|\text{ne}(G_\gamma(i,j),i)| = 2$. Denote the neighbors as u and v. We discuss the direction of the two edges $i \star\!\!\star u$ and $i \star\!\!\star v$. If the directions are $(\to,\to)$, then $S_\gamma(i,j) = \emptyset$. If $(\leftarrow,\to)$, then $S_\gamma(i,j) = \{u\}$. If $(\leftarrow,\leftarrow)$ or $(-,\leftarrow)$ or $(-,-)$, then $S_\gamma(i,j) = \{u,v\}$.

(a) If $(\leftrightarrow,\leftarrow)$, then we need to discuss neighbors of u, too. If $u \in \text{adj}(G_\gamma(i,j),v)$, then $u \star\!\!\star v$ is a merging edge, and by Fact 2, u and v have in total no more than 2 bearing edges of order 2. If u is not ancestral to i or j, then $S_\gamma(i,j) = \{v\}$. The case of $u \notin \text{an}(G_\gamma(i,j),\{i,j\})$ is trivial. If $u \in \text{an}(G_\gamma(i,j),\{i,j\})$, then u has at least one outgoing edge. If the outgoing edge is $u \to v$ (second row of Figure B.2), there are two sub-cases. If v has an bearing edge of order 2, then u has only one other neighbor, call it x, and we condition on x if and only if $u \leftarrow x$ or $u \leftrightarrow x$ and $x \in \text{an}(G_\gamma(i,j),j)$. If v has no bearing edge, then u can have at most 2 other neighbors. However, these bearing edges must not have arrow at u, due to the inducing path interpretation of MAG. Therefore we do not need to condition on these additional neighbors.

If the outgoing edge is not $u \to v$ (third row of Figure B.2), then there is some edge $u \to w$. If v has an bearing edge of order 2, then u no other neighbor than

$\{i, u, w\}$. If v has no bearing edge, then u could have one additional neighbor, call it x, and we condition on x if and only if $u \leftarrow x$ or $u \leftrightarrow x$ and $x \in \text{an}(G_\gamma(i,j), j)$.

If $u \notin \text{adj}(G_\gamma(i,j), v)$ (fourth row of Figure B.2), then u has at most two additional neighbors. If none of them are child of u, then $u \notin \text{an}(G_\gamma(i,j), \{i,j\})$ and $S_\gamma(i,j) = \{v\}$; If $u \to w$, $u \star\!\!-\!\!\star x$ then we condition on x if and only if $u \leftarrow x$ or $u \leftrightarrow x$ and $x \in \text{an}(G_\gamma(i,j), j)$.

(b) if $(\leftrightarrow, \to)$, the situations are simpler since we never condition on v. Since v must have either a bearing edge or a merging edge, u can have at most 2 bearing edges. If $u \in \text{an}(G_\gamma(i,j), j)$, then one of the edges is $u \to w$. As for the other one, $u \star\!\!-\!\!\star x$, we condition on x if and only if $u \leftarrow x$ or $u \leftrightarrow x$ and $x \in \text{an}(G_\gamma(i,j), j)$.

(c) If $(\leftrightarrow, \leftrightarrow)$: If neither of u and v are ancestral to j, then $S_\gamma(i,j) = \emptyset$. If both are ancestral to j (row 2-3 and first 2 figures of row 4 in Figure B.3), then they each has a outgoing edge. Then by Fact 2, there is at most one other bearing edge. WLOG, suppose u has $u \to w$ and $u \star\!\!-\!\!\star x$. Then $S_\gamma(i,j) = \{u, v, x\}$ if $x \in \text{an}(G_\gamma(i,j), \{u,j\})$ and otherwise $S_\gamma(i,j) = \{u, v\}$. If u is ancestral to j and v is not, then still u has one outgoing edge and at most one other edge. Then $S_\gamma(i,j) = \{u, x\}$ if $x \in \text{an}(G_\gamma(i,j), \{u,j\})$ and otherwise $S_\gamma(i,j) = \{u\}$.

3. $|\text{ne}(G_\gamma(i,j), i)| = 3$. Denote the three neighbors of i as u, v, w. By Fact 2, they each has at most one bearing edge. Therefore we do not need to look further, and $S_\gamma(i,j) \subseteq \{u, v, w\}$.

□

The following proof is similar to Lemma 2 of Sondhi and Shojaie (2019) with slight modification, we show the proof here for completeness.

Proof of Lemma 5.7. We write

$$\begin{aligned}\Sigma &= (I-B)^{-1}\Omega(I-B)^{-\top} \\ &= \left(\sum_{r=0}^{p-1} B^r\right)\Omega\left(\sum_{r=0}^{p-1} B^r\right)^\top \\ &= \left(\sum_{r=0}^{\gamma} B^r + \sum_{r=\gamma+1}^{p-1} B^r\right)\Omega\left(\sum_{r=0}^{\gamma} B^r + \sum_{r=\gamma+1}^{p-1} B^r\right)^\top\end{aligned}$$

Denote $\Lambda_H = \sum_{r=0}^{\gamma} B^r$ and $R_\gamma = \sum_{r=\gamma+1}^{\infty} B^r = \sum_{r=\gamma+1}^{p-1} B^r$. By the directed β-summability assumption, we have $\|\Lambda_H\| \le \frac{1-\beta^{\gamma+1}}{1-\beta}$ and $\|R_\gamma\| \le \frac{\beta^{\gamma+1}-\beta^p}{1-\beta}$. Now we can bound the difference between Σ and the local approximation version $\Sigma_H := \Lambda_H\Omega\Lambda_H^\top$, which only contains paths no longer than γ.

$$\begin{aligned}\|\Sigma - \Sigma_H\| &= \|\Lambda_H\Omega R_\gamma^\top + R_\gamma\Omega\Lambda_H^\top + R_\gamma\Omega R_\gamma^\top\| \\ &\le \|\Omega\|\left(2\|\Lambda_H\|\|R_\gamma\| + \|R_\gamma\|^2\right) \\ &\le \|\Omega\|\left(2\frac{(1-\beta^{\gamma+1})\beta^{\gamma+1}}{(1-\beta)^2} + \frac{\beta^{2\gamma+2}}{(1-\beta)^2}\right) \\ &= \|\Omega\|\frac{\beta^{\gamma+1}(2-\beta^{\gamma+1})}{(1-\beta)^2}\end{aligned}$$

We write $\gamma* = \log(\beta)^{-1}\left(\log M - \log 2 - \log\|\Omega\| - \log(\eta+2) - \log(1+3/\lambda)\right) - 1$. We invoke the error propagation lemma from Harris and Drton (2013). For any non-adjacent pair (i,j) and a set $S \subseteq V \setminus \{i,j\}$ with $|S| \le \eta$, whenever $\gamma \ge \gamma^*$, it holds that

$$|\rho(i,j|S) - \rho_H(i,j|S)| \le \lambda$$

where ρ_H is the partial correlation obtained from Σ_H. Since Σ_H only composes of short paths, $\rho_H(i,j|S_\gamma) = 0$ for every local-graph separator S_γ. Therefore $|\rho(i,j|S_\gamma)| < \lambda$. □

B.3 Treks

In this section we provide an algebraic explanation of Assumption 5. In particular, we review the trek representation of partial correlation in linear SEM. The representation clarifies that

conditional dependence in a linear SEM is tied to existence of paths/treks in the graph underlying the model. This allows us to argue that conditional dependence is typically induced by short versus long treks, which in turn provides the basis for exploiting small local separators in our algorithms.

To simplify the discussion, we present the following results assuming there is no selection variables in the graph. Let G be a mixed graph without undirected edges. We define a *trek* from node i to j as a tuple $\tau = (P_L, P_M, P_R)$, where P_L is a directed path from some node s to i, and P_R is a directed path from some node t to j, and P_M is either one bidirected edge $s \leftrightarrow t$ or the empty set when $s = t$. We define the *trek monomial* as $m_\tau = \beta^L \omega_{s,t} \beta^R$, where $\beta^L = \prod_{k \to l \in P_L} \beta_{kl}$ and $\beta^R = \prod_{k \to l \in P_R} \beta_{kl}$. Moreover, for sets C and D with $|C| = |D| = k$, we define a *trek system* T from C to D as a set of k treks whose initial nodes exhaust C and final nodes exhaust D. With abuse of notation we write T as a tuple of collections of paths (P_L, P_M, P_R), and define the *trek system monomial* as the product of trek monomials in the system, i.e., $m_T = \prod_{\tau \in T} m_\tau$. Each trek system determines a permutation of the initial and final nodes, which we call the sign of the system. Let $\mathcal{T}(C, D)$ denote the collection of all trek systems from C to D. By the Cauchy–Binet determinant expansion, we have,

$$\det \Sigma(C, D) = \sum_{R,S \subset V, |R|=|S|=k} \det \left((I - B)^{-\top}\right)_{C,R} \det \Omega_{R,S} \det \left((I - B)^{-1}\right)_{S,D} \tag{B.3}$$

$$= \sum_{T \in \mathcal{T}(C,D)} \operatorname{sign}(T) m_T. \tag{B.4}$$

We say a trek system T has *sided intersection* if two paths in P_L, P_R, or P_M have shared nodes. If T is a trek system between C and D with sided intersections, then its weight m_T is cancelled in the summation in (B.4), (for a proof, see Sullivant et al., 2010). In other words, the summation in (B.4) only needs to run over trek systems without sided intersections. Consequently, $\det(\Sigma(i \cup S, j \cup S)) = 0$ if and only if every system of treks from $i \cup S$ to $j \cup S$ has a sided intersection. The later condition is also called *t-separation*. For Gaussian SEMs, in which conditional independence is characterized by zero partial correlation, this means $W_i \perp\!\!\!\perp W_j | W_S$ if and only if $\sum_{T \in \mathcal{T}(i \cup S, j \cup S)} \operatorname{sign}(T) m_T = 0$.

We will show next that Assumption 5 can be expressed as a condition on trek weights. Let $G = (V, E)$ be a MAG. For non-adjacent nodes $i, j \in V$, and $S \subseteq V(G_\gamma) \setminus \{i, j\}$, we denote $\mathcal{T}_\gamma(i, j, S)$ as the collection of trek systems from $i \cup S$ to $j \cup S$ in $G_\gamma(i, j)$, and $\mathcal{T}_\gamma^C(i, j, S) := \mathcal{T}(i \cup S, j \cup S) \setminus \mathcal{T}_\gamma^C(i, j, S)$. By our definition, $\mathcal{T}_\gamma^C(i, j, S)$ only contains treks that goes through a node outside $G_\gamma(i, j)$.

Lemma 26. *Let G be a MAG. Under Assumption 3, if there exists $\beta \in (0, 1)$ such that*

$$\max_{i \notin \text{adj}(G,j)} \min_{S_\gamma \in \mathcal{S}_{\eta,\gamma}(i,j)} \left| \sum_{T \in \mathcal{T}_\gamma^C(i,j,S_\gamma)} \text{sign}(T) m_T \right| = O(\beta^\gamma),$$

where $\mathcal{S}_{\eta,\gamma}(i, j)$ is the collection of γ-local-graph separators of size at most η, then Assumption 5 holds.

Proof. By Definition 5, if a set S_γ is a γ-local-separator of (i, j), then it is a separator of i and j in $G_\gamma(i, j)$, so all trek systems between $i \cup S_\gamma$ and $j \cup S_\gamma$ have sided intersections in $G_\gamma(i, j)$, and hence also in G. Following Draisma et al. (2013), we only need to take summation over trek systems without sided intersection in G. Therefore,

$$\begin{aligned} \sum_{T \in \mathcal{T}(i,j,S_\gamma)} \text{sign}(T) m_T &= \sum_{T \in \mathcal{T}_\gamma(i,j,S_\gamma)} \text{sign}(T) m_T + \sum_{T \in \mathcal{T}_\gamma^C(i,j,S_\gamma)} \text{sign}(T) m_T \\ &= \sum_{T \in \mathcal{T}_\gamma^C(i,j,S_\gamma)} \text{sign}(T) m_T. \end{aligned}$$

Now denote $\Sigma(i, j|S_\gamma)$ as the (i, j)-th entry of the conditional variance matrix given S_γ. We have

$$\rho(i, j|S_\gamma) = \frac{\Sigma(i, j|S_\gamma)}{\sqrt{\Sigma(i, i|S_\gamma)\Sigma(j, j|S_\gamma)}} = \frac{\sum_{T \in \mathcal{T}(i,j,S_\gamma)} \text{sign}(T) m_T}{\det(\Sigma(S_\gamma, S_\gamma))} \frac{1}{\sqrt{\Sigma(i, i|S_\gamma)\Sigma(j, j|S_\gamma)}}.$$

By the fact that $\Sigma(j, j|S_\gamma) \geq \Sigma(j, j|V \setminus \{j\}) = 1/\omega_{jj}$, and $\det(\Sigma(S_\gamma, S_\gamma)) \geq M^{-\eta}$ under Assumption 3, we have $|\rho(i, j|S_\gamma)| = O(\beta^\gamma)$. □

B.4 Choice of γ

Recall the simulation study in Section 3.5. We randomly generate Erdős-Renyi graphs and power-law graphs with $p = |V| = 200$ nodes and average node degree 2. Edge weights are drawn uniformly from $\pm[.1, 1]$, and $n = 100$ observations are generated by the `rmvDAG` function. We randomly choose $q = 0.2p$ nodes as latent variables, and the rest as observed. We include no selection variables. We run lFCI with $\gamma = \{2, 3, 4, 5, 6, 7, 8, p/2, p-1\}$, and $\alpha = \{10^{-15,}, 10^{-9}, 10^{-8}, 10^{-7}, \cdot 10^{-6}, 10^{-3}, \cdot 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$. We repeat the experiment 100 times for each α, and compare the true positive and false positive discoveries of the skeleton of the true PAG.

Figure B.4 suggests that as long as γ is large enough, the algorithm yields almost identical outputs. The only exception is the case of power-law graph with $\gamma = 2$, in which the algorithm appears to be too aggressive, and the performance is sub-par on a part of the pROC curve. We also point out that in the "many false positive" part of the curves (i.e., to the right end), methods with smaller γ tends to perform better, since they perform fewer tests. However, that region is only relevant for "discovery". In general, we recommend using $\gamma = O(\log |V|)$.

B.5 Simulations with local-graph separation oracle

In this section we investigate the performance of population version of FCI, FCI+ and lFCI. We use the exact same settings as the simulation study in Section 3.5. We run FCI and FCI+ with oracle m-separations, and lFCI with oracle γ-local-separations, with $\gamma = 6$. The experiment is repeated 50 times. In the power-law setting with $p = 100$, FCI is ususally much more computationally burdensome than FCI+ and lFCI. Due to this limitation, we only report the runs in which FCI terminated within 8 hours. Results are shown in Table B.1.

Performances of the methods are evaluated by the proportion of cases in which the true (unique) maximally informative PAG is recovered. Computational costs are compared based on the total number of CI tests and the maximal reach levels. We also check Assumption 5 directly with $\rho^* = \max_{(i,j)\notin E} \min_{S\in\mathcal{S}_{\eta,\gamma}(i,j)} |\rho(i, j|S)|$ and report the median over all cases.

		ER				PL				WS			
	p	FCI	FCI+	lFCI	lFCImb	FCI	FCI+	lFCI	lFCImb	FCI	FCI+	lFCI	lFCImb
	20	1	1	1	1	1	1	1	1	1	1	1	1
%Recovered	50	1	1	1	1	1	1	1	1	1	1	1	1
	100	1	1	1	1	1	1	1	1	1	1	1	1
	20	0	0	0	0	0	0	0	0	0	0	0	0
ρ^*	50	0	0	0	0	0	0	0	0	0	0	0	0
	100	0	0	0.004	0	0	0	0.009	0	0	0	0.001	0
log(#CI)	20	8.3	6.7	6.3	5.7	10.9	9.3	6.2	5.7	6.6	6.1	5.8	5.3
	50	10.4	8.1	8.2	7.4	17.1	13.1	7.7	7.0	8.5	7.7	7.6	6.9
	100	11.9	8.9	8.9	8.2	15.2	12.1	8.9	8.2	9.8	8.8	8.9	8.2
m_{reach}	20	4.9	4.0	2.9	1.6	8.7	7.9	2.4	1.3	4	3	3	1
	50	6.2	5.2	3.2	2.0	13.6	12.5	2.8	1.4	5	4	3	2
	100	6.9	6.0	3.2	1.8	12.7	12.7	2.7	1.3	6	5	3	2

Table B.1: Average performance of population version of FCI, FCI+, lFCI and lFCImb with graphs of size $p \in \{20, 50, 100\}$ and fixed $\gamma = 6$.

Note that FCI/FCI+ are exact algorithms, meaning they recover the maximally informative PAG in all cases, because m-separations always correspond to zero partial correlations. lFCI/lFCI_mb are not guaranteed to be complete if Assumption 5 is violated — though their outputs are correct PAGs, they are sometimes not maximally informative. The proposed methods show improvement on the computational aspects: the number of tests are consistent over different graph generating schemes, whereas FCI/FCI+ could suffer in power-law cases. The maximum reach level (m_{reach}) confirms the results in Section 3.2 — in most cases the local separators are indeed as small as 3.

B.6 Simulations with standardized normal coefficients

In this section, we aim to provide evidence that Assumption 5 is satisfied in many common large networks when data is standardized. The fact that in many common scenarios the SEM

corresponding to the standardized data has almost all coefficients less than 1 is demonstrated in a simulation study in Appendix B of Sondhi and Shojaie (2019). We further conjecture that the sum of long trek weights are also minimal, by showing the covariance matrix is well approximated using only short treks. For this purpose, we generate a random ER or power-law graph and draw edge weights from either a uniform distribution on $(-10, 10)$ or a normal distribution with mean 0 and standard deviation 3. We intentionally choose wide ranges for the coefficient to allow large fluctuation in the network. Then a SEM in the form of (1.2) is constructed with this weighted adjacency matrix B and random error variance Ω, where Ω is a diagonal matrix with diagonal entries drawn from a uniform distribution on $(1, 2)$. We denote $\Sigma = (I - B)^{-1}\Omega(I - B)^{-\top}$ and $\widetilde{\Sigma}$ as its standardized version, where $\widetilde{\Sigma}_{ij} = \Sigma_{ij}/\sqrt{\Sigma_{ii}\Sigma_{jj}}$ for each (i, j)-entry. The standardized data can be seem as drawn from another SEM corresponding to the same graph G, but with different set of parameters $(\widetilde{B}, \widetilde{\Omega})$, which satisfies $\widetilde{\Sigma} = (I - \widetilde{B})^{-1}\widetilde{\Omega}(I - \widetilde{B})^{-\top}$. We compute the maximal entry-wise difference between $\widetilde{\Sigma}$ and its short-trek approximation $\widetilde{\Sigma}_\gamma = (\sum_{k=0}^{\gamma} \widetilde{B})\widetilde{\Omega}(\sum_{k=0}^{\gamma} \widetilde{B})^\top$. We define $d_\gamma := \max_{i,j}(|\widetilde{\Sigma} - \widetilde{\Sigma}_\gamma|_{i,j})$ and report the smallest γ such that $d_\gamma \leq 10^{-4}$ over 100 iterations. We use the quantity d_γ as a surrogate to check Assumption 5 because we have shown in the proof of Lemma 5.7 that $\|\widetilde{\Sigma} - \widetilde{\Sigma}_\gamma\| = O(\beta^\gamma)$ is a sufficient condition of Assumption 5.

Figure B.5 demonstrates d_γ is indeed very small in most settings with $\gamma \approx \log p$. The results suggest that Assumption 5 is indeed plausible for standardized data.

B.7 Simulations with local moral graphs

In this section we demonstrate that with large enough γ, the γ-local moral graphs usually coincide with moral graphs. Following the simulation settings in Section 3.5, in the numerical study below, we generate random DAGs with $p \in \{100, 200, 500\}$ nodes and average node degree 2. Similarly, we also use $\gamma = 5, 6, 7$ for $p = 100, 200, 500$ and randomly choose $q = 0.2p$ nodes as latent nodes, and compute the skeleton of the MAG over the observed ones. We do not introduce selection variables, simply because undirected edges do not contribute to the difference between local and non-local Markov blankets.

	Erdős-Renyi	Power Law	Watts-Strogatz
$p = 100$, $\gamma = 5$	0.99	1.00	0.96
$p = 200$, $\gamma = 6$	0.99	1.00	0.97
$p = 500$, $\gamma = 7$	0.99	1.00	0.97

Table B.2: Proportion of random graphs (out of 200 iterations) with γ-local moral graph equal to moral graph.

We compute the moral graph and γ-local moral graph for each MAG over 200 simulation iterations, and report the proportion of cases when local moral graph is different from the moral graph. The results are reported in Table B.2. We see for the choice of γ used in our simulations, almost all local moral graphs are identical to the moral graphs. This is especially likely to be true for power-law graphs, since they tends to have smaller diameter.

B.8 Search Pools

The graph in Figure B.6 is an example in which lFCI may needs to perform more conditional independence tests than FCI.

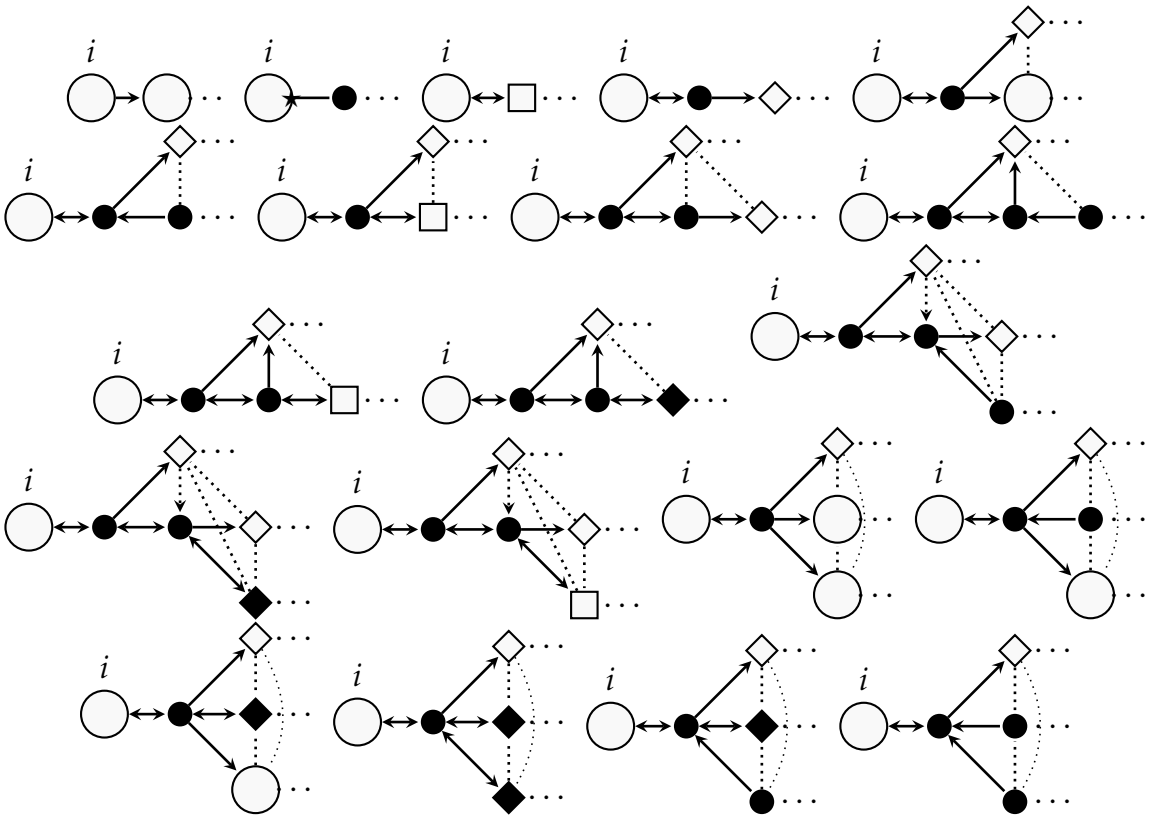


Figure B.1: Local graph configurations with $\eta = 3$ and $|\text{ne}(G_\gamma(i,j),i)| = 1$. A separator (not necessarily minimal) is marked with shade. Marked edge represents the pattern of $G_\gamma(i,j)$, while absence of an edge represents the absence pattern of $G_\gamma(i,j)$. Ellipses between nodes means this edge is allowed to occur in $G_\gamma(i,j)$, as long as it agrees with the MAG property and local-path property. The square shape represents a node with no outgoing edge (except the marked ones). The diamond shape represents a node that controls whether the separator is minimum — if this node is not ancestor of j, then smaller separator exists.

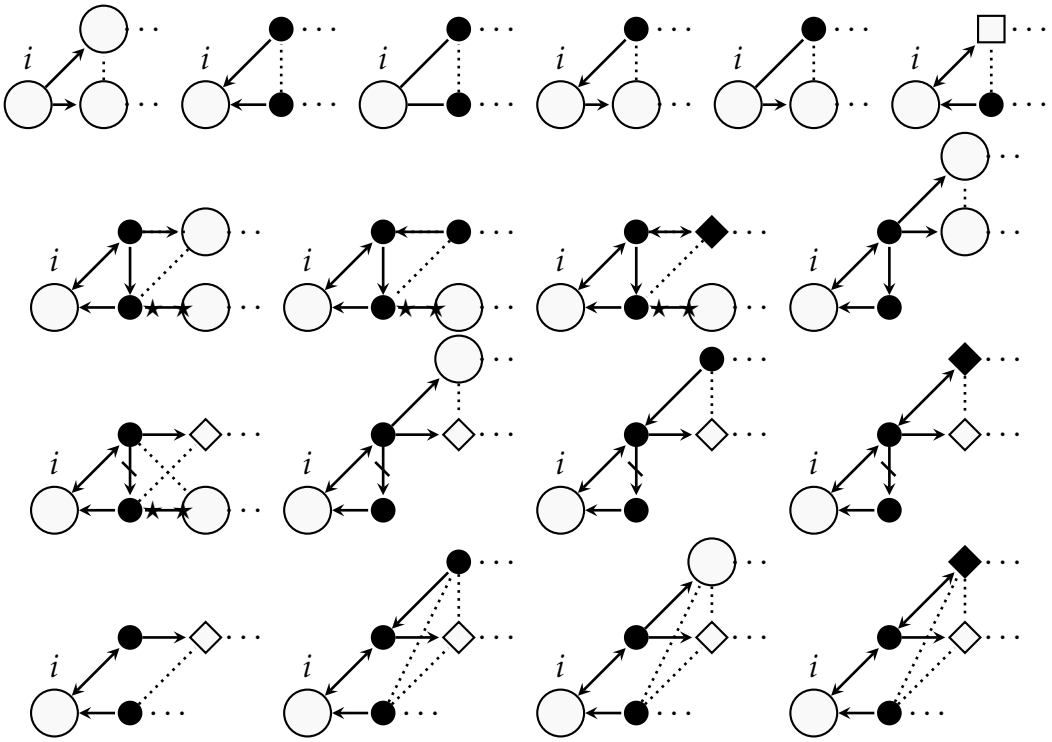


Figure B.2: Local graph configurations with $\eta = 3$ and $|\mathrm{ne}(G_\gamma(i,j),i)| = 2$. (continues in Figure B.3). A separator (not necessarily minimal) is marked with shade. Marked edge represents the pattern of $G_\gamma(i,j)$, while absence of an edge represents the absence pattern of $G_\gamma(i,j)$. Ellipses between nodes means this edge is allowed to occur in $G_\gamma(i,j)$, as long as it agrees with the MAG property and local-path property. The square shape represents a node with no outgoing edge (except the marked ones). The diamond shape represents a node that controls whether the separator is minimum — if this node is not ancestor of j, then smaller separator exists.

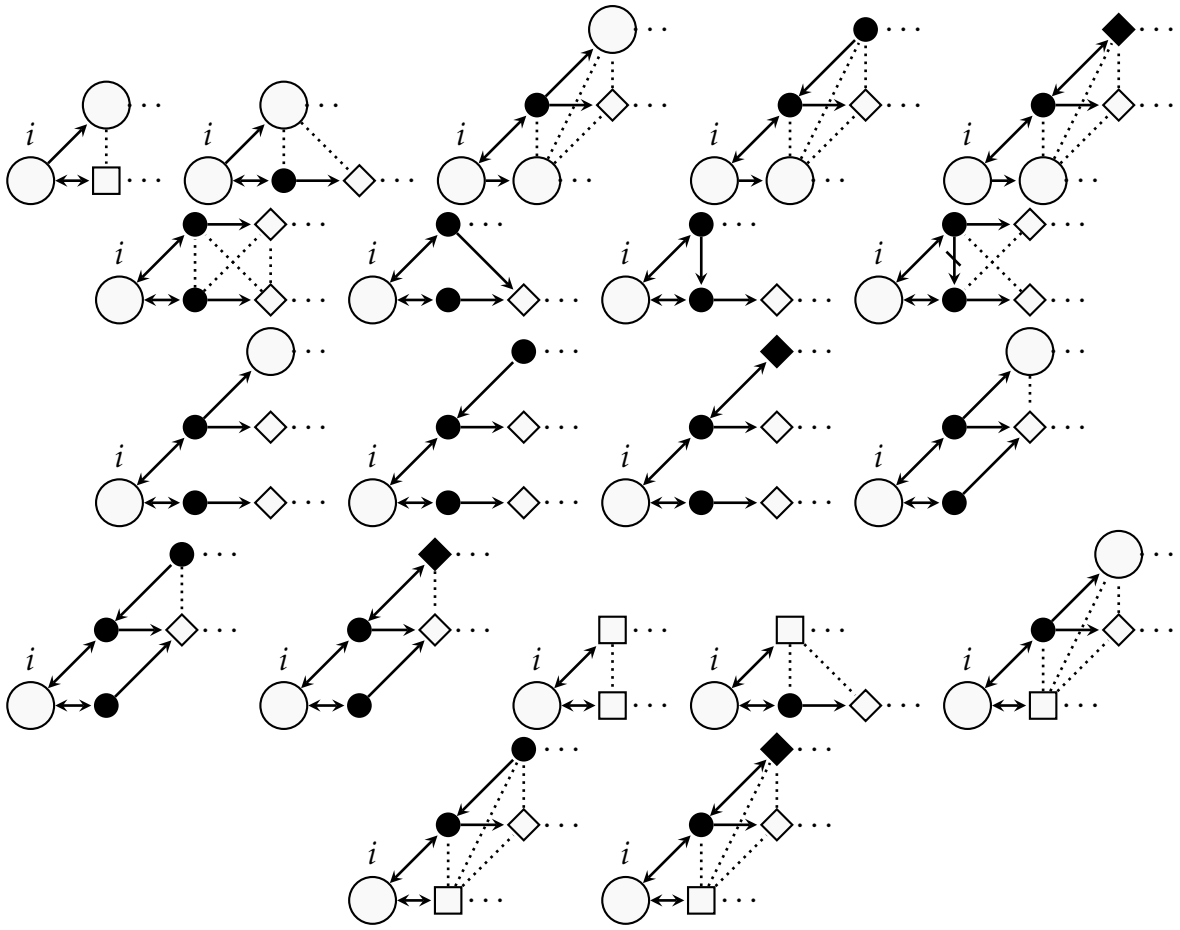


Figure B.3: Local graph configurations with $\eta = 3$ and $|\mathrm{ne}(G_\gamma(i,j),i)| = 2$. A separator (not necessarily minimal) is marked with shade. Marked edge represents the pattern of $G_\gamma(i,j)$, while absence of an edge represents the absence pattern of $G_\gamma(i,j)$. Ellipses between nodes means this edge is allowed to occur in $G_\gamma(i,j)$, as long as it agrees with the MAG property and local-path property. The square shape represents a node with no outgoing edge (except the marked ones). The diamond shape represents a node that controls whether the separator is minimum — if this node is not ancestor of j, then smaller separator exists.

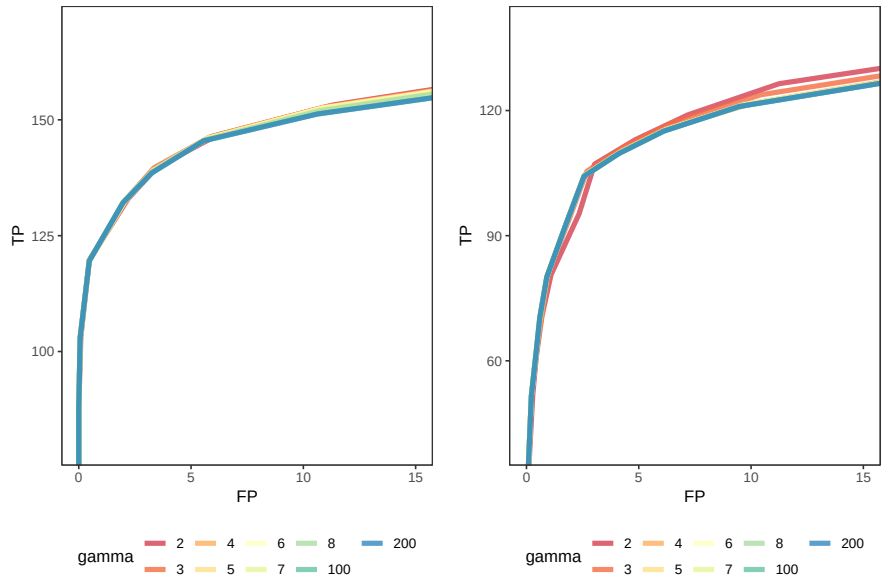


Figure B.4: pROC curves of Algorithm 15 with different choices of γ performed on ER graphs (left) and power-law graphs (right).

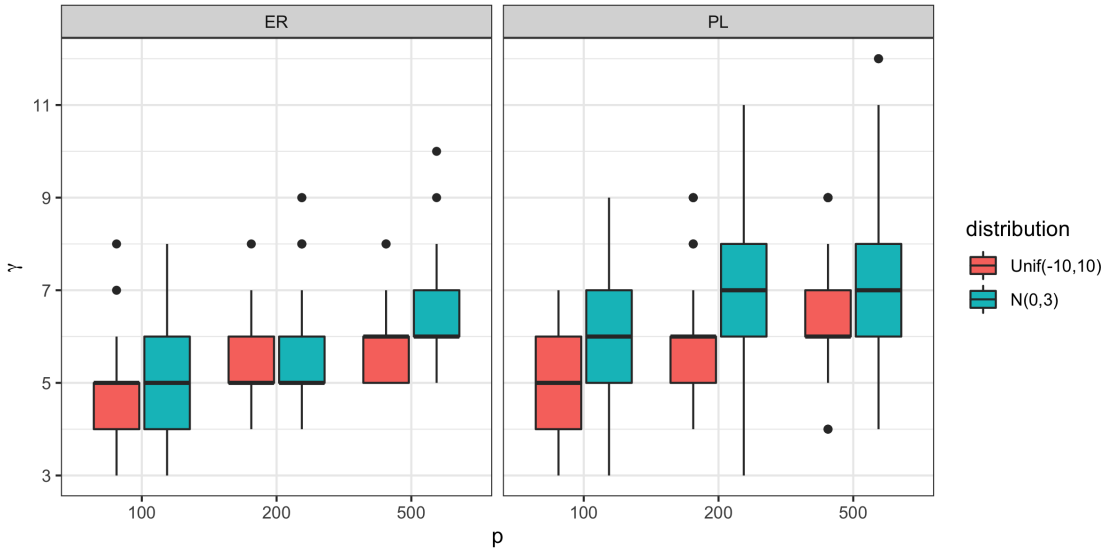


Figure B.5: Values of $\min\{\gamma : d_\gamma \leq 10^{-4}\}$ for various settings of ER and power-law graphs, with edge weights drawn from either Uniform $(-10, 10)$ or $N(0, 3^2)$, and $n = 100, 200, 500$. The minimal γ values scale with $\log p$.

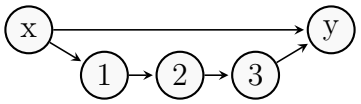


Figure B.6: No edge is removed at level 0. At level 1, if the edge (x, y) is checked after removing $(x, 2)$ and $(2, y)$, then FCI performs less CI tests than lFCI (with $\gamma = 5$), because the node 2 is local to x and y but not in their neighborhoods.

Appendix C

APPENDICES TO CHAPTER 4

C.1 Proofs

Proof of Lemma 12. We first show the $\Rightarrow$ direction of the first statement. Suppose $u \rightsquigarrow v$ and u does not have a directed arrow into A_{uv}. Suppose A_{uv} is fully connected (or a singleton) and define $B = \{a \to u : a \in A_{uv}\}$. If the CPDAG has an extension that is consistent with B, then $u \notin \text{an}(v)$ in this extension; for a definition of extension, see e.g. (Dor and Tarsi, 1992). We apply the background knowledge algorithm (Perkovic et al., 2018), which is guaranteed to be successful if such an extension exists. Since A_{uv} is fully connected, at each step, Meek's rules never orient edges between nodes in A_{uv}; therefore, also never orient any edge from u into A_{uv}, and the algorithm can enforce B without causing any conflict. But this means there exists a DAG in the Markov equivalence class (MEC) in which $u \notin \text{an}(v)$. Therefore we conclude the $\Rightarrow$ direction of statement 1. The $\Leftarrow$ direction: since A_{uv} is not a clique, there must be two nodes $a, a' \in A_{u,v}$ that are non-adjacent and (a, u, a') is not a v-structure. Then every DAG in the MEC must have either $u \to a$ or $u \to a'$, and therefore a directed path to v.

Now we show the second statement. The $\Rightarrow$ direction: if there exists a possibly directed path from u to v, then there must be an unshielded possibly directed path. However, this means the first edge on this path is oriented out of u in some DAG in the MEC, in which case $u \in \text{an}(v)$. The $\Leftarrow$ direction follows directly from our definition. □

Proof of Lemma 13. In the first statement, x must block some path $\pi = (u, \ldots, s, x, t, \ldots, v)$ that is d-connected given W. Therefore x is a non-collider on π, and the two subpaths $\pi_{xu} = (x, s \ldots, u)$ and $\pi_{xv} = (x, t, \ldots, v)$ are both d-connected given W. In each DAG of the MEC, we can pick π_{xu} or π_{xv}, whichever starts with an outgoing arrow (one of them

must do so, since x is not a collider) and follow non-collider arrows until we either reach $\{u, v\}$ or encounter a collider which is unblocked by W (meaning it is ancestral to W). In the first case $x \in \text{an}(u \cup v)$ and in the second case $x \in \text{an}(W)$.

In the second statement, x must unblock some path $\pi = (u, \ldots, t, \ldots, v)$ that is blocked by W, where t is a collider on π and ancestor of x (or $t = x$). Note that t (and also x) is d-connected to both u and v given W. For this reason x must not be ancestral to W. On the other hand, if x is ancestral to u, then there exists a directed path $\pi' = (x, \ldots, u)$. But then we obtain a d-connecting path between u, v given W by gluing together π', the directed path $(x, \ldots, t)$ and the subpath of π from t to v, which is a contradiction. □

Proof of Theorem 7, 8 and 10. These three results are direct consequences of Lemma 13 and DNA-faithfulness with respect to the corresponding conditional independence (CI) statements checked by the learning steps. □

Proof of Theorem 9. With the stated sample size, we may apply the error propagation computation in Lemma 1 of Ravikumar et al. (2011) and in Lemma 6 of Harris and Drton (2013), which gives the following bound:

$$\mathbb{P}\left\{\max_{i \neq j, |S| \leq K+1} \left|\rho(i, j|S) - \widehat{\rho}(i, j|S)\right| \geq \lambda/2\right\} \leq \zeta. \tag{C.1}$$

Consequently, with probability at least $1 - \zeta$, the sets $\Omega_\lambda^{\uparrow K}(\mathbb{P})$ and $\bar{\Omega}_\lambda^{K+1}(\mathbb{P})$ are correctly inferred from the data, and therefore the DNA output is correct. □

Proof of Theorem 11. The low-dimensional result can be obtained from an error propagation computation that is entirely analogous to the one from the proof of Theorem 9 above. In this case, we have $\mathbb{P}\left\{\|\Sigma^{-1} - S_n^{-1}\|_\infty > \lambda^*/2\right\} \leq \zeta$ and consequently all moral graphs are correctly inferred from data. In the high-dimensional case, the theory for the CLIME estimator $\widehat{\Omega}$ guarantees that $\|\widehat{\Omega} - \Sigma^{-1}\|_\infty \leq 4\|\Sigma^{-1}\|_1 \lambda_n$ for $\lambda_n \geq \|\Sigma^{-1}\|_1 \|\Sigma - S_n\|_\infty$ Cai et al. (2011). The moral graphs of subgraphs are also correct (see Lemma 5 of Ghoshal and Honorio, 2018). In both cases, the assumption on non-zero entries in the inverse covariance matrix

guarantees that non-sinks will not be misspecified as sink. Therefore the algorithm is correct by Theorem 10. □

Proof of Lemma 14. To show the first statement: Let π_0 be an arbitrary ordering of G and suppose $u \not\rightsquigarrow v$ is discordant with π_0. We claim we can swap the ordering to obtain a new ordering that is compatible with G and $u \not\rightsquigarrow v$. We write $\pi_0 = (X, u, Y, v, Z)$. Denote $A = Y \cap \mathrm{an}(v)$. Since $u \not\rightsquigarrow v$, we also have $u \not\rightsquigarrow A$. Since π_0 is valid for G, there is no edge between u and A, no edge between $Y \setminus A$ and v, and all edges between A and $Y \setminus A$ are in the form of $A \longrightarrow Y \setminus A$. So the new ordering $\pi_0' = (X, A, v, u, Y \setminus A, Z)$ is valid for G. If D is order-constraining, then applying the swap operation above does not create new discordant pairs, and therefore an ordering can be swapped according to D until it agrees with both D and $[G]$. □

Proof of Theorem 12. In the output $L = (L_1, \ldots, L_m)$, for each $k = 2, \ldots, m$, it holds that $v \not\rightsquigarrow \cup_{i=1}^{k-1} L_i$ for all $v \in L_k$. Consequently, all edges between L_k and layers preceding it must be directed into L_k in G. Hence, L satisfies the requirements to be a valid layering of G. □

Proof of Theorem 13. Under the Markov and the DNA-faithfulness assumption, D is a DNA set of G and, thus, by Theorem 12, L is a layering of G. It is now sufficient to show the DAG can be learned by recursively applying SP. We prove this claim by induction. Under SMR, no graph on L_1 sparser than the subgraph of G over L_1 is Markov to the pattern of conditional independence (CI) relations among L_1. Therefore the output of SP on L_1 is consistent with the target MEC. Moving on to $L_1 \cup L_2$, the sparsest graphs that are Markov to the CI relations among $L_1 \cup L_2$ have L_1 ordered exactly as π_1. Hence, the optimal ordering of L_2 can be in an optimization that holds π_1 fixed and considers the joint distribution conditional on X_{L_1}. The general induction steps to the layers after L_2 proceed in the same way. □

Proof of Lemma 15. We show if u, v are d-separated by S, then they are also d-separated by $S \cap (\mathrm{an}(u) \cup \mathrm{an}(v))$. Let $T = S \setminus (\mathrm{an}(u) \cup \mathrm{an}(v))$. The d-separation relation implies that every path between u and v either has a non-collider in S or a collider not in S whose descendants

are also not in S. Consider an arbitrary path π between u and v. If π does not go through T, then π is blocked by $S \setminus T$. If π contains some $z \in T$ and z is a collider on π, then π is blocked by $S \setminus T$ for not including z. If z is not a collider on π, then we can follow the arrows from z on π until we reach a collider, call it x, i.e., $(z, \ldots, x, \ldots)$ is a subpath of π. Since $x \in \mathrm{de}(z) \subset T$ and x is a collider on π that is not included in $S \setminus T$, the set $S \setminus T$ d-separates u, v. □

Appendix D

APPENDICES TO CHAPTER 5

D.1 Proofs

Proof of Lemma 16. The claim in (i) follows directly from the definition of d-separation. In particular, if $X_k \to Y_j \in G$ then the path from X_k to Y_j will not be d-separated by X_{-k}, and hence $Y_j \not\perp\!\!\!\perp X_k \mid Y_{-k}$.

To prove (ii), note that $\{X_1, \cdots X_q\} \prec \{Y_1, \cdots Y_p\}$ implies that $Y_{j_1} \to \cdots \to Y_{j_0}$ cannot include any X_ks. Thus, without loss of generality, suppose that

$$Y_{j_1} \to \cdots \to Y_{j_0} \equiv Y_{j_1} \to Y_{j_2} \to \cdots \to Y_{j_m} \to Y_{j_0},$$

where $m = 1$ is permitted. Now, considering that X_{k_0} is an ancestor of Y_{j_0}, by faithfulness of $\mathcal{P}$ with respect to G, in order for any set $S \subset V - \{k_0, j_0\}$ to d-separate X_{k_0} and Y_{j_0}, it must include at least one of the variables $Y_{j_1}, Y_{j_2}, \ldots, Y_{j_m}$. Noting that the construction used in H only adjusts for X_{-k_0}, and does not adjust for any Y_js, the path from X_{k_0} to Y_{j_0} will not be d-separated by X_{-k_0}. It follows from the definition of d-separation that X_{k_0} and Y_{j_0} are conditionally dependent given X_{-k_0}, which means that $X_{k_0} \to Y_{j_0} \in H^{(0)}$. □

Proof of Lemma 17. The result follows again from the definition of d-separation for DAGs, and the faithfulness of $\mathcal{P}$. In particular, if $X_k \to Y_j \in G$, $\{X_{-k}, Y_{-j}\}$ does not d-separate X_k from Y_j, and hence, $Y_j \not\perp\!\!\!\perp X_k \mid \{X_{-k}, Y_{-j}\}$.

Next, note that if X_{k_0}, Y_{j_0} and Y_{j_1} form an open collider in G, i.e. if $X_{k_0} \to Y_{j_1} \leftarrow Y_{j_0}$ and $X_{k_0} \nrightarrow Y_{j_0}$, then the above argument implies that $X_{k_0} \to Y_{j_1} \in H^{(-j)}$. On the other hand, since Y_{j_1} is a common descendent of X_{k_0} and Y_{j_0}, d-separation implies that $X_{k_0} \not\perp\!\!\!\perp Y_{j_0} \mid Y_{j_1}$, or more generally, $X_{k_0} \not\perp\!\!\!\perp Y_{j_0} \mid \{X_{-k_0}, Y_{-j_0}\}$. Thus, $X_{k_0} \to Y_{j_0} \in H^{(-j)}$. □

Proof of Lemma 19. Suppose $\mathrm{mb}'(v)$ and $\mathrm{mb}''(v)$ are two distinct valid minimal Markov blankets of v. Let $J = \mathrm{mb}'(v) \cap \mathrm{mb}''(v)$, $K = \mathrm{mb}'(v) \setminus J$, $L = \mathrm{mb}''(v) \setminus J$ and $W = V \setminus (J \cup K \cup L \cup v)$. By the intersection property, $v \perp\!\!\!\perp K | J \cup L$ and $v \perp\!\!\!\perp L | J \cup K$ implies $v \perp\!\!\!\perp K \cup L | J$. By the elementary formula of conditional independence, $v \perp\!\!\!\perp K \cup L | J$ and $v \perp\!\!\!\perp W \cup L | J \cup K$ implies $v \perp\!\!\!\perp W \cup K \cup L | J$, and therefore J is also a valid MB. But this is only possible when $C = \mathrm{mb}'(v) = \mathrm{mb}''(v)$. Hence minimal MB is unique.

Next we show $\mathrm{cmb}_U() = \mathrm{mb}(v) \setminus U$ in two parts. First we show $\mathrm{mb}(v) \subseteq \mathrm{cmb}_U(v) \cup U$. We write $W = \mathrm{mb}(v) \cap (\mathrm{cmb}_U(v) \cup U)$, $Z = \mathrm{mb}(v) \setminus W$, $S = (\mathrm{cmb}_U(v) \cup U) \setminus W$. Then, by the definitions of conditional Markov blanket and Markov blanket, $v \perp\!\!\!\perp Z | W, S$ and $v \perp\!\!\!\perp S | W, Z$. By the intersection rule, $v \perp\!\!\!\perp Z, S | W$ which establishes W as a valid Markov Blanket. However, since $W \subseteq \mathrm{mb}(v)$ and MB is minimal, this is only possible when $W = \mathrm{mb}(v)$ and $Z = \emptyset$, i.e., $\mathrm{mb}(v) \subseteq \mathrm{cmb}_U(v) \cup U$. Finally we show $\mathrm{cmb}_U(v) \subseteq \mathrm{mb}(v)$. We write $T = \mathrm{cmb}_U(v) \cap \mathrm{mb}(v)$. Since $T \cup U \supseteq \mathrm{mb}(v)$, it must holds that $v \perp\!\!\!\perp S | T, U$, which implies that T is also a valid conditional Markov blanket. However, since $T \subseteq \mathrm{cmb}_U(v)$ and the conditional Markov blanket is minimal, this is only possible when $T = \mathrm{cmb}_U(v)$, i.e., $\mathrm{cmb}_U(v) \subseteq \mathrm{mb}(v)$. This completes the proof. □

Proof of Lemma 21. Since both sets are replaced by their supersets, $\widehat{E}$ is still a superset of true edges. Under layering-adjacency-faithfulness, for any adjacent pairs $k \in \boldsymbol{\mathcal{X}}$ and $j \in \boldsymbol{\mathcal{Y}}$ and any set $T \subset \boldsymbol{\mathcal{Y}}_{-j}$, it holds that $X_k \not\perp\!\!\!\perp Y_j | \boldsymbol{\mathcal{X}}_{-k} \cup T$, so the search loop in Algorithm 9 will not make any false negative errors. Therefore, we only need to show that all additional edges can be removed by the search loop. For any nonadjacent $k \in \boldsymbol{\mathcal{X}}$ and $j \in \boldsymbol{\mathcal{Y}}$, let $W \subseteq \boldsymbol{\mathcal{X}}$ be an arbitrary set satisfying $W \supseteq \left(S^{(0)} \cap S_j^{(-j)}\right)$. Then, $X_k \perp\!\!\!\perp Y_j | \boldsymbol{\mathcal{X}}_{W \setminus k} \cup \boldsymbol{\mathcal{Y}}_{\mathrm{pa}_j \cap \boldsymbol{\mathcal{Y}}}$. The existence of this separator implies that the edge between k and j can be removed by the search loop. □

Proof of Theorem 14. Under layering-adjacency-faithfulness, if $k \in \boldsymbol{X}$ and $j \in \boldsymbol{Y}$ are adjacent, then $\boldsymbol{X}_k \not\perp\!\!\!\perp \boldsymbol{Y}_j | \boldsymbol{X}_{-k}$ and $\boldsymbol{X}_k \not\perp\!\!\!\perp \boldsymbol{Y}_j | \boldsymbol{X}_{-k} \cup \boldsymbol{Y}_{-j}$, so $k \in S_j^{(0)} \cap S_j^{(-j)}$. Therefore $\widehat{E}$ contains all true edges.

By the intersection property, it holds that $\boldsymbol{X}_k \perp\!\!\!\perp \boldsymbol{Y}_j | \boldsymbol{X}_{S_j^{(0)} \cap S_j^{(-j)} \setminus k} \cup \boldsymbol{Y}_T \Rightarrow \boldsymbol{X}_k \perp\!\!\!\perp \boldsymbol{Y}_j | \boldsymbol{X}_{-k} \cup$

$\boldsymbol{Y}_T$. Under layering-adjacency-faithfulness, this implies not type II error can be made by the search loop. Suppose $k \in \boldsymbol{X}$ and $j \in \boldsymbol{Y}$ are non-adjacent, then we must have $\boldsymbol{X}_k \perp\!\!\!\perp \boldsymbol{Y}_j | \text{pa}(j)$, which implies $\boldsymbol{X}_k \perp\!\!\!\perp \boldsymbol{Y}_j | \boldsymbol{X}_{S_j^{(0)} \cap S_j^{(-j)} \setminus k} \cup \boldsymbol{Y}_T$ with the set $T = \text{pa}(j) \cap \boldsymbol{Y}$. This construction provide one d-separator, and hence there could not by type I error either. □

Lemma 27 (Partial correlation and regression coefficient)**.** *Let β_j^S be the linear regression coefficients regressing a set of variables X_S onto X_j, and let $\rho(i, j|S)$ denote the partial correlation, then*

$$\beta_{ji}^{S \cup i} = 0 \Leftrightarrow \rho(i, j|S) = 0 \Leftrightarrow \Sigma_{i,j} - \Sigma_{S,i}^\top \Sigma_{S,S}^{-1} \Sigma_{S,j} = 0.$$

Proof. We prove this by direct computation. We denote the block covariance matrix $\Sigma_{S \cup \{i,j\}, S \cup \{i,j\}}$ as

$$\begin{pmatrix} A & B & C \\ B^\top & D & E \\ C^\top & E^\top & F \end{pmatrix} := \begin{pmatrix} \Sigma_{S,S} & \Sigma_{S,i} & \Sigma_{S,j} \\ \Sigma_{i,S} & \Sigma_{i,i} & \Sigma_{i,j} \\ \Sigma_{j,S} & \Sigma_{j,i} & \Sigma_{j,j} \end{pmatrix}$$

For the first term, we have $\beta_j^{S \cup i} = \left(\Sigma_{S \cup i, S \cup i}\right)^{-1} \Sigma_{S \cup i, j}$, and hence

$$\beta_{ji}^{S \cup i} = \left(D - B^\top A^{-1} B\right)^{-1} \left(E - B^\top A^{-1} C\right).$$

For the second term, we have $\rho(i, j|S) = -\frac{p_{ij}}{\sqrt{p_{ii} p_{jj}}}$ where P is the precision matrix $P = \Sigma^{-1}$. Therefore $\rho(i, j|S) = 0$ if and only if $[(\Sigma_{S \cup \{i,j\}, S \cup \{i,j\}})^{-1}]_{i,j} = 0$. Concretely,

$$[(\Sigma_{S \cup \{i,j\}, S \cup \{i,j\}})^{-1}]_{i,j} = -\beta_{ji}^{S \cup i} \left(F - C^\top A^{-1} C - (-C^\top A^{-1} B + E^\top) \beta_{ji}^{S \cup i}\right)^{-1}.$$

Both quantities equal zero if and only if $\Sigma_{i,j} - \Sigma_{S,i}^\top \Sigma_{S,S}^{-1} \Sigma_{S,j} = 0$. □

Proof of Theorem 15. Choose $\alpha_n = 2(1 - \Phi(n^{1/2} c_n / 2))$. Denote $h_n = \max_{j \in \boldsymbol{\mathcal{Y}}} |\widehat{S}_j^{(0)} \cap \widehat{S}_j^{(1)}|$. We first suppose we run Algorithm 9 until some level $m_n' \geq m_n$ such that $m_n' + h_n = O(n^{1-b})$ as in Assumption 8. By Lemma 3 of Kalisch and Bühlmann (2007), the probability of any type I error in edge k, j with conditioning set $T \subset \boldsymbol{\mathcal{Y}} \setminus j$, denoted as $E_{k,j|T}^{\text{I}}$, is bounded by an exponential term

$$\sup_{k \in \boldsymbol{\mathcal{X}}, j \in \boldsymbol{\mathcal{Y}}, T \subset \boldsymbol{\mathcal{Y}} \setminus j, |T| \leq m_n'} \mathbb{P}\left\{E_{k,j|T}^{\text{II}} | \boldsymbol{\mathcal{A}}\right\} \leq O(n - m_n' - h_n) \exp(-C_3 (n - m_n' - h_n) c_n^2),$$

and the probability of type II error bounded by an exponential term

$$\sup_{k\in\mathcal{X},j\in\mathcal{Y},T\subset\mathcal{Y}\setminus j,|T|\le m_n'} \mathbb{P}\left\{E^{\mathrm{II}}_{k,j|T}|\mathcal{A}\right\} \le O(n-m_n'-h_n)\exp(-C_4(n-m_n'-h_n)c_n^2)$$

for some finite constant C_3 and C_4. In our algorithm, $|\{k,j,T\} : |T| \le m_n| = O(pq^{1+m_n'}) = O(\exp(c_0 n^\kappa))$. So the total error probability can be bounded,

$$\begin{aligned}
&\mathbb{P}\{\text{an error occurs in Algorithm 9 with p-cor testing}|\mathcal{A}\}\\
&\le O(\exp(c_0 n^\kappa))O((n-m_n'-h_n)\exp(-C_5(n-m_n'-h_n)c_n^2))\\
&\le O(\exp(c_0 n^\kappa + \log(n-m_n'-h_n) - C_5(n^{1-2d} - m_n' n^{-2d} - h_n n^{-2d})))\\
&= O(\exp(-Cn^{1-2d})). \qquad \text{(D.1)}
\end{aligned}$$

By (D.1), with high probability, the sample version of searching loop makes no mistake up to search level m_n'. Then by the reasoning in Lemma 5 of Kalisch and Bühlmann (2007), the search step also have high probability of terminating at the same level as the population version, which is either $m_n - 1$ or m_n. This completes the proof. □

Proof of Proposition 1. Similar to the proof above, it holds that

$$\sup_{j\in\mathcal{Y}} \mathbb{P}\left\{E^{\mathrm{II}}_{j,\widehat{S}_j^{(0)}}\right\} \le O(n+1-p)\exp(-C_4(n+1-p)c_n^2),$$

$$\sup_{j\in\mathcal{Y}} \mathbb{P}\left\{E^{\mathrm{II}}_{j,\widehat{S}_j^{(1)}}\right\} \le O(n+2-p-q)\exp(-C_4(n+2-p-q)c_n^2).$$

Since there are in total $pq+(p+q-1)q$ many tests, the probability of errors can be bounded by

$$\mathbb{P}\{\neg\mathcal{A}\} \le q(2p+q-1)(n+2-p-q)\exp(-C'(n+2-p-q)c_n^2) = O(\exp(-C'n^{1-2d})),$$

using the fact that $n \gg p+q$. □

Proof of Proposition 2. By Theorem 1 of Fan and Lv (2008), for each selection problem, SIS has an error probability of $O(\exp(-Cn^{1-2d}/\log n))$. We apply a union bound over the total number of $2p$ problems and error probability is $O(\exp(n^\kappa - Cn^{1-2d}/\log n)) = O(\exp(n^{1-2d}/(n^{1-2d-\kappa} - C/\log n))) = O(\exp(-Cn^{1-2d}/\log n))$. □

Proof of Proposition 3. This proof will proceed as following: First we will show that under the Gaussian construction, all design matrices satisfy the Restricted Eigenvalue (RE) condition. Then we show that these conditions implies bounded error of lasso estimate. Lastly, we show that under Assumption 10 the minimal non-zero coefficients are large enough to guarantee successful screening.

The RE condition is a technical condition for lasso to have bounded error. In particular, a design matrix $\mathcal{D} \in \mathbb{R}^{n\times d}$ satisfies RE over a set $S \subseteq [d]$ with parameter η if $\frac{1}{n}\|\mathcal{D}\Delta\|_2^2 \geq \eta\|\Delta\|_2^2$ for all $\Delta \in \mathbb{R}^d$ such that $\|\Delta_{[d]\setminus S}\|_1 \leq 3\|\Delta_S\|_1$. Let $\Gamma_{\min}(\cdot)$ be the minimum eigenvalue function and $\rho^2(\cdot)$ the maximum diagonal entry. It is shown in Theorem 7.16 of Wainwright (2019) that for any random design matrix $\mathcal{D} \sim N(0, \Sigma_D)$, there exists universal positive constants $c_1 < 1 < c_2$ such that $\frac{\|\mathcal{D}w\|_2^2}{n} \geq c_1\|\sqrt{\Sigma_D}w\|_2^2 - c_2\rho^2(\Sigma_D)\frac{\log d}{n}\|w\|_1^2$ for all $w \in \mathbb{R}^d$, with probability at least $1 - \frac{\exp(-n/32)}{1-\exp(-n/32)}$. This inequality implies RE condition with $\eta = \frac{c_1}{2}\Gamma_{\min}(\Sigma_D)$ uniformly for any subset S of cardinality at most $|S| \leq \frac{c_1}{32c_2}\frac{\Gamma_{\min}(\Sigma_D)}{\rho^2(\Sigma_D)}\frac{n}{\log d}$. Note that our required rate of $h_n = O(n^{1-b})$ and $s_n = O(n^{1-a})$ satisfies this sparsity requirement. Therefore, under our assumption on minimum eigenvalues, the matrix $\mathcal{X} \cup \mathcal{Y}$ satisfied RE with some constant η with probability at least $1 - \frac{\exp(-n/32)}{1-\exp(-n/32)}$. Consequently, the design matrices used to learn $S_j^{(0)}$ and $S_j^{(1)}$, i.e., $\mathcal{X}$ and $\{(\mathcal{X} \cup \mathcal{Y}_{-j})\}_{j\in\mathcal{Y}}$, all satisfy RE condition.

Next we apply Theorem 7.13 from Wainwright (2019). In particular, for any solution of the lasso problem with regularization level lower bounded as $\lambda_n \geq 2\|\mathcal{D}^\top\omega/n\|_\infty$ where $\mathcal{D}$ is the design matrix, it holds that $\|\hat{\beta} - \beta^*\|_2 \leq \frac{2}{\eta}\sqrt{s}\lambda_n$. In our case, this guarantees all the ℓ_2 errors of estimation is bounded, in particular

$$\max_{j\in\mathcal{Y}}\left(\max\{\|\hat{\gamma}_j - \gamma_j^*\|_2, \|\hat{\theta}_j - \theta_j^*\|_2\}\right) \leq \frac{2}{\eta}\sqrt{s_n}\lambda_n.$$

For any Gaussian designs $\mathcal{D} \in \mathbb{R}^{n\times d}$ with columns standardized to $\max_j \frac{\|\mathcal{D}_j\|_2}{\sqrt{n}} \leq C$, the Gaussian tail bound guarantees $\mathbb{P}\left\{\|\frac{\mathcal{D}^\top\omega}{n}\|_\infty > C\sigma(\sqrt{\frac{2\log d}{n}} + \delta)\right\} \leq 2e^{-\frac{n\delta^2}{2}}$. To learn $S_j^{(0)}$, we set $\lambda_n \asymp \sqrt{2\log p/n}$, then the errors (in ℓ_2 and ℓ_∞) are in the order of $O(\sqrt{\frac{2s_n\log p}{n}}) = O(n^{1-b+\kappa})$ with probability at least $1 - 2e^{-\frac{n\delta^2}{2}}$. A union bound over all regression problems

bounds all errors with probability $1-4pe^{-\frac{n^2\delta}{2}} = 1-O(\exp(c_0 n^\kappa - \frac{\delta}{2}n^2)) = 1-O(\exp(-\frac{\delta}{2}n^2))$.

Last we show that Assumption 10 implies a beta-min condition for the regression problems. Note that

$$\gamma_{kj} = \rho(j,k|\mathcal{X}_{-k})\sqrt{\text{Var}(Y_j|\mathcal{X})\text{Var}(X_k|\mathcal{X}_{-k}\cup Y_j)},$$
$$\theta_{kj} = \rho(j,k|\mathcal{X}_{-k}\cup\mathcal{Y}_{-j})\sqrt{\text{Var}(Y_j|-j,k)\text{Var}(X_k|-j,k)},$$

and the last terms are both bounded below by some constant. Therefore we have $\min_{\gamma_{jk}\neq 0}|\gamma_{kj}| > c_n'$, $\min_{\theta_{jk}\neq 0}|\theta_{kj}| > c_n'$ for some ${c_n'}^{-1} = O(n^{-d})$.

Putting pieces together, we have shown that with probability at least $1-4qe^{-\frac{n\delta^2}{2}} - \frac{e^{-n/32}}{1-e^{-n/32}}$, it holds that the lasso solution exactly recovers $S_j^{(0)}$ and $S_j^{(1)}$ for all $j \in \mathcal{Y}$. □

Proof of Proposition 4. First, Theorem 2 from Chakraborty and Shojaie (2022), guarantees that for any $\epsilon > 0$ there exists positive constant c_1 and c_2 and $\gamma \in (0, 1/4)$ with $\gamma > \kappa$ such that

$$\sup_{j\in\mathcal{Y},k\in\mathcal{X},|T|\leq m_n} \mathbb{P}\left\{|\widehat{\rho}^*(j,k|Z\cup\mathcal{X}_{-k}) - \rho^*(j,k|Z\cup\mathcal{X}_{-k})| > \epsilon\right\}$$
$$\leq O(pq^{m_n+1}[2\exp(-c_1 n^{1-2\gamma}\epsilon^2) + n^4\exp(-c_2 n^\gamma)]), \quad \text{(D.2)}$$

where pq^{m_n+1} is an upper bound on the total number of tests needed. Under the required faithfulness assumption, this implies the probability of type-1 and type-2 error is bounded by $O(\exp(c_0 n^\kappa - c_1 n^{1-2\gamma-2d}) + n^4\exp(c_0 n^\kappa - c_2 n^\gamma))$. The first term is dominated by $O(\exp(-c_1 n^{1-2\gamma-2d}))$ and the second term by $O(\exp(-c_2 n^\gamma))$. This means with high probability, small sample partial correlations corresponds to correct conditional indepdencies and the searching loop is correct. □

Proof of Proposition 5. The following analysis applies the fast rate results from Haris et al. (2022). Note that the GAM problem defined above is a special case with the truncated basis functions $\mathcal{F}^{(r)}$. Following Corollary 13 of Haris et al. (2022), for each u, k there is a high

probability event on which the empirical process of the loss is controlled. Assumption 12 allows applying an union bound over u and k. Therefore, with high probability, it holds that

$$c\|f_{vu}^{(r),0} - \hat{f}_{vu}\|^2 \leq \boldsymbol{\varepsilon}(\hat{f}_{vu}) \lesssim \max(s_{\max} \log(p+q)/n).$$

Under the beta-min condition, this suggests the estimates support are supersets of $S_j^{(0)}$ and $S_j^{(1)}$. □