Causal Discovery for Linear Non-Gaussian Models with Disjoint Cycles

Mathias Drton[1] **Marina Garrote-López**[2] **Niko Nikov**[3] **Elina Robeva**[4] **Y. Samuel Wang**[5]

[1]Technical University of Munich and Munich Center for Machine Learning, Munich, Germany
[2]KTH Royal Institute of Technology, Stockholm, Sweden
[3]University of Edinburgh, Edinburgh, UK
[4]University of British Columbia, Vancouver, BC, Canada
[5]Cornell University, Ithaca, NY, USA

Abstract

The paradigm of linear structural equation modeling readily allows one to incorporate causal feedback loops in the model specification. These appear as directed cycles in the common graphical representation of the models. However, the presence of cycles entails difficulties such as the fact that models need no longer be characterized by conditional independence relations. As a result, learning cyclic causal structures remains a challenging problem. In this paper, we offer new insights on this problem in the context of linear non-Gaussian models. First, we precisely characterize when two directed graphs determine the same linear non-Gaussian model. Next, we take up a setting of cycle-disjoint graphs, for which we are able to show that simple quadratic and cubic polynomial relations among low-order moments of a non-Gaussian distribution allow one to locate source cycles. Complementing this with a strategy of decorrelating cycles and multivariate regression allows one to infer a block-topological order among the directed cycles, which leads to a consistent and computationally efficient algorithm for learning causal structures with disjoint cycles.

1 INTRODUCTION

In this work we study linear structural causal models [Pearl, 2013] that allow for feedback loops. Throughout, we consider a collection of observed random variables $(X_i)_{i\in V}$ that is indexed by the nodes of a directed graph $G = (V, E)$. The *linear structural equation model* (LSEM) associated to the graph G hypothesizes that the random variables $(X_i)_{i\in V}$ satisfy a system of structural equations

$$X_i = \sum_{j\in \mathrm{pa}(i)} \lambda_{ji} X_j + \varepsilon_i, \tag{1}$$

where $\mathrm{pa}(i) = \{j \in V : j \to i \in E\}$ is the set of parents of node $i \in V$, and the noise terms $(\varepsilon_i)_{i\in V}$ are centered and independent. Our problem of interest is causal discovery from observational data, i.e., given i.i.d. samples from this model, we are interested in learning the graph G and the unknown coefficients $\lambda_{ij} \in \mathbb{R}$.

Our focus is on discovery in models with non-Gaussian errors. The non-Gaussianity may render causal directions identifiable also in settings where classical Markov equivalence theory prevents identification [Maathuis et al., 2018]. In the well understood setting of directed acyclic graphs (DAGs), each linear non-Gaussian model is associated to a unique DAG [Shimizu et al., 2006]. This fact has spurred active development of discovery algorithms [Shimizu et al., 2011], also dealing with issues of high dimensionality [Wang and Drton, 2019, Tramontano et al., 2022] or latent variables [Salehkaleybar et al., 2020, Liu et al., 2021, Adams et al., 2021, Cai et al., 2023, Wang and Drton, 2023, Schkoda et al., 2024]. For a comprehensive review beyond these selective references, see Shimizu [2022].

A problem that remains challenging also in the linear non-Gaussian case, however, is discovery of causal structures that are allowed to contain directed cycles. Directed cycles encode feedback loops and lead to difficulties such as models that cannot be described in terms of conditional independence and whose linear causal effects do not correspond to coefficients in conditional expectations [Spirtes et al., 1998]. For linear Gaussian models, this results in the fact that conditional independence relations alone do not allow one to learn precisely which variables are adjacent to each other in the graph [Richardson, 1996b, Ghassami et al., 2020]. Hence, while algorithms focusing on conditional independence tests [Richardson, 1996a, Semnani and Robeva, 2025] can be useful, they are unable to infer all identifiable aspects of the underlying causal structure.

Work of Lacerda et al. [2008] stresses the possibility of different graphs with reversed cycles inducing the same linear non-Gaussian models. In the present work, we revisit this issue and give a complete characterization of when any two directed graphs are equivalent in the linear non-Gaussian setting (see Section 3).

Lacerda et al. [2008] and more recently Dai et al. [2024] show that cyclic models can be learned using independent component analysis (ICA) or Independent Subspace Analysis (ISA) combined with a sorting strategy over all permutations of the considered variables. Here we consider a setting that lies between the classical DAGs and the fully arbitrary structures considered by Lacerda et al. [2008]. Specifically, we focus on graphs in which cycles remain disjoint (i.e., each node belongs to at most one cycle). While such graphs do appear in applications [Grace et al., 2016], our motivation is computational and focused on a discovery algorithm that learns such graphs computationally efficiently (Section 4). In particular, our proposed algorithm avoids the search over permutations that features in ICA-based approaches.

We remark that the algorithm of Lacerda et al. [2008] outputs all graphs and edge weights for which the product of absolute edge weights along any cycle is smaller than 1. This can also be done in our setting of cycle-disjoint graphs.

A key insight for our algorithm is that the strong components of a *cycle-disjoint graph* can be learned using simple quadratic and cubic constraints on the second and third moments. The algorithm uses the constraints to recursively identify initial strong components and leverages further algebraic relations among moments to learn the edges inside each component. Multivariate regression then allows one to determine the edges between the components. Our procedure is reminiscent of that of Kawahara et al. [2010], which learns a DAG over groups of variables but does not offer an efficient way to find the right grouping of the variables.

The rest of the paper is organized as follows. Section 2 reviews preliminaries for linear non-Gaussian SEMs. Section 3 provides a complete characterization of the equivalence classes of directed graphs which give rise to the same linear non-Gaussian model (Thm. 3.5). In Section 4, we propose our algorithm for causal discovery based on second and third moments of the observed random vector. Our method works for cycle-disjoint directed graphs, i.e., directed graphs in which no two directed cycles intersect (Def. 4.1). In such graphs, if several vertices form a directed cycle, then there are no additional edges among those vertices. We conclude with Section 5, which presents numerical experiments.

2 PRELIMINARIES

Let $G = (V, E)$ be a directed graph (DG) with vertices V and directed edges E. Since we allow directed cycles in our graph, it is not always possible to find a topological ordering of the vertices (an ordering so that edges only go from a smaller to a larger numbered vertex). Instead, there exists such an ordering among the strong components of G.

Definition 2.1. *A subset of vertices $C \subseteq V$ is a* strong component *(or a* strongly connected component*) of G, if there exists a directed path from any vertex $i \in C$ to any other vertex $j \in C$, and C is a maximal such subset.*

Now, let $X = (X_i)_{i \in V}$ be a random vector that satisfies the linear structural equations of (1). We can rewrite the equations in (1) in matrix-vector format as

$$X = \Lambda^{\mathrm{T}} X + \varepsilon, \tag{2}$$

where $\epsilon = (\epsilon_i)_{i \in V}$ collects the noise terms, and the parameter matrix $\Lambda = (\lambda_{ij})_{i,j \in V}$ has $\lambda_{ij} = 0$ unless $i \to j$ is an edge in E. For a well-defined model, the matrix Λ should satisfy $I - \Lambda$ being invertible. Indeed, precisely then, the system in (2) admits a unique solution, $X = (I - \Lambda)^{-\mathrm{T}} \varepsilon$.

Definition 2.2. *We denote the set of $|V| \times |V|$ matrices $\Lambda = (\lambda_{ij})$ for which $I - \Lambda$ is invertible and $\lambda_{ij} = 0$ if $i \to j \notin E$ by $\mathbb{R}^E$. Denote the set of Λ which in addition satisfy $\lambda_{ij} \neq 0$ if $i \to j \in E$ by $\mathbb{R}^E_*$.*

We assume throughout that the random vector ε, and thus also $X = (I - \Lambda)^{-\mathrm{T}} \varepsilon$, is centered. The second moments of X form the covariance matrix

$$S = (s_{ij})_{i,j \in V} = (\mathbb{E}[X_i X_j])_{i,j \in V}. \tag{3}$$

Similarly, the third-order moments of X form a tensor of order 3, which we denote by

$$T = (t_{ijk})_{i,j,k \in V} = (\mathbb{E}[X_i X_j X_k])_{i,j,k \in V}. \tag{4}$$

Let $\Omega^{(2)} = (\mathbb{E}[\varepsilon_i \varepsilon_j])_{i,j \in V}$ and $\Omega^{(3)} = (\mathbb{E}[\varepsilon_i \varepsilon_j \varepsilon_k])_{i,j,k \in V}$ be the second and third-order moment tensors of ε. By the independence of the ε_i, both $\Omega^{(2)}$ and $\Omega^{(3)}$ are diagonal tensors whose diagonal entries we denote by $\omega_i^{(2)}$ and $\omega_i^{(3)}$, respectively. Since $X = (I - \Lambda)^{-\mathrm{T}} \varepsilon$, we have that

$$\begin{aligned} S &= S(\Lambda, \Omega^{(2)}) := (I - \Lambda)^{-\mathrm{T}} \Omega^{(2)} (I - \Lambda)^{-1}, \\ T &= T(\Lambda, \Omega^{(3)}) \\ &\qquad := \Omega^{(3)} \bullet (I - \Lambda)^{-1} \bullet (I - \Lambda)^{-1} \bullet (I - \Lambda)^{-1}, \end{aligned}$$

where $\bullet$ denotes *Tucker product*. In other words, the entry t_{ijk} of T equals

$$\sum_{a,b,c \in V} [(I - \Lambda)^{-1}]_{ai} [(I - \Lambda)^{-1}]_{bj} [(I - \Lambda)^{-1}]_{ck} \Omega^{(3)}_{abc} \tag{5}$$

$$= \sum_{a \in V} [(I - \Lambda)^{-1}]_{ai} [(I - \Lambda)^{-1}]_{aj} [(I - \Lambda)^{-1}]_{ak} \Omega^{(3)}_{aaa}. \tag{6}$$

Going forward, we will study the non-Gaussian model specified by (2) via the set of second- and third-order moments

$$\mathcal{M}^{2,3}(G) = \Big\{ \big(S(\Lambda, \Omega^{(2)}), T(\Lambda, \Omega^{(3)})\big) : \Lambda \in \mathbb{R}^E, \\ \Omega^{(2)} \in \mathbb{R}^{|V|\times|V|} \text{ diagonal and positive definite}, \\ \Omega^{(3)} \in \mathbb{R}^{|V|\times|V|\times|V|} \text{ diagonal} \Big\}.$$

We call this set the *model of second- and third-order moments* corresponding to G.

While the covariance matrix S captures the entire dependence structure in the Gaussian case, higher-order moments are relevant in the linear non-Gaussian setting. Assuming that the diagonal elements of $\Omega^{(3)}$ are nonzero (that is, the distribution of ε is skewed), the second- and third-order moments are enough to identify the graph G and the coefficient matrix Λ when G is acyclic [e.g., Wang and Drton, 2019]. In Section 4, we use the second- and third-order moments to learn graphs we term cycle-disjoint (Def. 4.1).

The classical notion of treks describes the source of nonzero covariances in linear SEMs. In our linear non-Gaussian setting, the more general notion of a k-trek is needed.

Definition 2.3. *A k-trek between the vertices $i_1, \dots, i_k$ is a k-tuple of paths $\tau = (P_1, \dots, P_k)$ with common source $\ell = \mathrm{top}(\tau)$ and respective sinks $i_1, \dots, i_k$. The node ℓ is called the top of the trek. The set of all k-treks between $i_1, \dots, i_k$ is denoted by $\mathcal{T}(i_1, \dots, i_k)$. A trek $\tau = (P_1, \dots, P_k) \in \mathcal{T}(i_1, \dots, i_k)$ is called a* simple trek *if $P_u \cap P_v = top(\tau)$ for all $u, v \in [k]$, that is, the only common vertex between each pair of paths P_u and P_v is the top of the trek. Moreover, we say that τ has* non-empty sides *if any path P_u in τ is non-empty, i.e., $top(\tau) \neq i_u$.*

The *trek rule* uses these treks to specify how the entries of S and T can be expressed in terms of the entries of Λ, $\Omega^{(2)}$, and $\Omega^{(3)}$. It was originally used in the Gaussian setting [Sullivant et al., 2010] and was later extended to higher-order moments for linear non-Gaussian models [Robeva and Seby, 2021, Améndola et al., 2023]. It states that the covariances s_{ij} and third moments t_{ijk} can be expressed as

$$s_{ij} = \sum_{(P_i,P_j)\in\mathcal{T}(i,j)} \lambda^{P_i}\lambda^{P_j}\omega^{(2)}_{\mathrm{top}(P_i,P_j)}, \tag{7}$$
$$t_{ijk} = \sum_{(P_i,P_j,P_k)\in\mathcal{T}(i,j,k)} \lambda^{P_i}\lambda^{P_j}\lambda^{P_k}\omega^{(3)}_{\mathrm{top}(P_i,P_j,P_k)},$$

where for a path $P = (u_1, \dots, u_r)$ we write $\lambda^P = \lambda_{u_1u_2}\lambda_{u_2u_3}\dots\lambda_{u_{r-1}u_r}$.

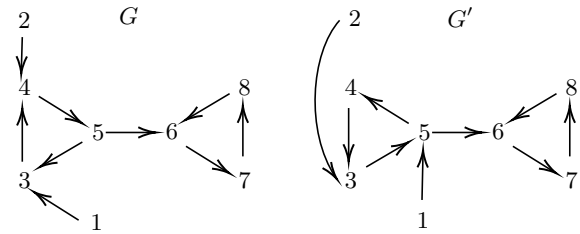

Figure 1: Distribution equivalent graphs G and G'.

3 DISTRIBUTION EQUIVALENT GRAPHS

In the case of acyclic graphs, observational data is enough to uniquely recover the underlying causal graph [Shimizu et al., 2006]. However, when we allow graphs containing cycles, we cannot identify the graph (and thus also parameters) uniquely. We now state a characterization of all the graphs and parameters that give rise to the same distributions. The discussion and results of this section hold for all directed graphs and do not require the disjoint cycles assumption we make later when discussing discovery.

For any directed graph $G = (V, E)$, let $\mathcal{P}(G)$ be the set of all joint distributions of random vectors $X = (I - \Lambda)^{-1}\varepsilon$ that solve the structural equations from (2) for $\Lambda \in \mathbb{R}^E_*$ and ε a random noise vector with independent components.

Definition 3.1. *Let $G = (V, E)$ and $G' = (V, E')$ be two graphs on the same set of vertices. Then G and G' are* distribution equivalent *if $\mathcal{P}(G) = \mathcal{P}(G')$.*

The following Example 3.2 shows that we can obtain distribution equivalent graphs by transforming the structural equations. We then prove in Theorem 3.5 that in fact all distribution equivalences arise from such transformations. Our consideration of nonzero parameters λ_{ij} in Definition 2.2 is tied to the fact that to transform the models, we need to divide by the λ_{ij} coefficients; see also Example 3.6 below.

Example 3.2. *Let G and G' be the DGs pictured in Figure 1. The graph G gives rise to the following structural equations in variables $3, 4, 5$:*

$$\begin{cases} X_3 = \lambda_{13}X_1 + \lambda_{53}X_5 + \varepsilon_3, \\ X_4 = \lambda_{24}X_2 + \lambda_{34}X_3 + \varepsilon_4, \\ X_5 = \lambda_{45}X_4 + \varepsilon_5. \end{cases}$$

Assuming all of the λs are nonzero, we can rearrange these equations to solve for the variables on the right-hand side

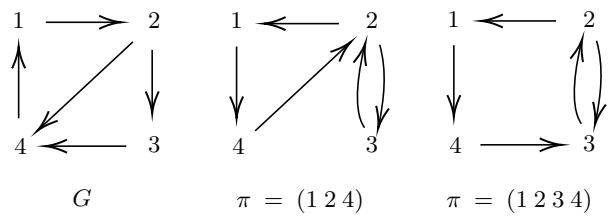

Figure 2: Permutations that factor according to vertex-disjoint cycles in G.

of each equation and obtain

$$\begin{cases} X_3 = \underbrace{(-\lambda_{24}/\lambda_{34})}_{\lambda'_{23}} X_2 + \underbrace{(1/\lambda_{34})}_{\lambda'_{43}} X_4 + \underbrace{(-1/\lambda_{34})\varepsilon_4}_{\varepsilon'_3}, \\ X_4 = \underbrace{(1/\lambda_{45})}_{\lambda'_{54}} X_5 + \underbrace{(-1/\lambda_{45})\varepsilon_5}_{\varepsilon'_4}, \\ X_5 = \underbrace{(-\lambda_{13}/\lambda_{53})}_{\lambda'_{15}} X_1 + \underbrace{(1/\lambda_{53})}_{\lambda'_{35}} X_3 + \underbrace{(-1/\lambda_{53})\varepsilon_3}_{\varepsilon'_5} \end{cases}$$

Note that this transformation does not affect the variables $1, 2, 6, 7, 8$. *In particular,* X *also lies in the model generated by* G'. *Therefore,* G *and* G' *are distribution equivalent.*

Definition 3.3. *Let* $G = (V, E)$ *be a DG, and let* π *be a permutation of* V. *We say that* π factors according to vertex-disjoint cycles *in* G *if for all* $\tau = (i_1\ i_2\ \cdots\ i_s)$ *in the disjoint cycle decomposition of* π, *the directed cycle* $i_1 \to i_2 \to \cdots \to i_s \to i_1$ *is in* G. *Such a permutation* π *determines a unique DG* $\pi(G) = (V, \pi(E))$ *via*

(i) if $i = \pi(j)$ *and* $i \neq j$, *then* $i \to j \in \pi(E)$, *and*

(ii) if $i \neq \pi(j)$ *and* $i \neq j$, *then* $i \to j \in \pi(E)$ *if and only if* $i \to \pi(j) \in E$.

Condition (ii) above is equivalent to "if $i \neq \pi(j)$, *then* $i \to \pi^{-1}(j) \in \pi(E)$ *if and only if* $i \to j \in E$."

Put differently, in order to construct the graph $\pi(G)$, we need to reverse one or several disjoint cycles in G, and if the edge $i \to j$ was on one of these cycles (and thus got reversed to $j \to i$), all edges $k \to j$ that originally pointed to j have to now be changed to point to i, i.e., $k \to i$.

Note that in Example 3.2 we had $G' = \pi(G)$ with $\pi = (3\ 4\ 5)$. In other words, we reversed the cycle $3 \to 4 \to 5 \to 3$, and we had to adjust the edges pointing to $3, 4, 5$ according to the rule above. Although in this example all cycles in G are vertex-disjoint, this need not be the case in general. Here we provide another example.

Example 3.4. *Figure 2 pictures* G *and* $\pi(G)$ *for two different choices of* π. *In fact, these three graphs comprise an entire distribution equivalence class.*

We remark that if π factors according to vertex-disjoint cycles in G, then the graphs G and $\pi(G)$ have the same sets of p-adjacencies as defined in Richardson [1996a].

The theorem below characterizes all graphs distribution equivalent to a given directed graph G. Here, we write $=_d$ for equality in distribution.

Theorem 3.5. *Two directed graphs* $G = (V, E)$ *and* $G' = (V, E')$ *are distribution equivalent if and only if* $G' = \pi(G)$ *for a permutation* π *on* V *that factors according to vertex-disjoint cycles in* G. *Moreover, given* $\Lambda \in \mathbb{R}_*^E$ *and a noise vector* ε *with independent components for the model corresponding to* G, *the permutation* π *uniquely determines a coefficient matrix* $\Lambda' \in \mathbb{R}_*^{E'}$ *and a noise vector* ε' *with independent components for the model given by* G' *such that* $X' := (I - \Lambda')^{\mathrm{T}}\varepsilon' =_d (I - \Lambda)^{-\mathrm{T}}\varepsilon =: X$.

PROOF. We first prove the necessity of our condition. A candidate (Λ', ε') must satisfy $\varepsilon' =_d (I - \Lambda')^{\mathrm{T}}(I - \Lambda)^{-\mathrm{T}}\varepsilon$. Since ε' must have independent components and the entries of ε are non-Gaussian, the Darmois-Skitovich Theorem [Darmois, 1953, Skitovich, 1953] implies that every row of $(I-\Lambda')^{\mathrm{T}}(I-\Lambda)^{-\mathrm{T}}$ has precisely one nonzero entry. In other words, there exists a permutation matrix P and a dilation matrix D so that

$$(I - \Lambda)PD = I - \Lambda'. \tag{8}$$

We can write $I - \Lambda = (-\lambda_{ij})_{i,j\in V}$ with $\lambda_{ii} = -1$ for $i = 1, \ldots, n$. Suppose that right multiplication by P permutes the columns of $I - \Lambda$ via the permutation π, so that the diagonal entries of $(I - \Lambda)P$ are $-\lambda_{i\pi(i)}$. But $I - \Lambda'$ has diagonal entries equal to 1, so we must have $\lambda_{i\pi(i)} \neq 0$. This implies that $\pi(i) = i$ or $i \to \pi(i) \in E$. In particular, π factors according to vertex-disjoint cycles in G. Moreover, since $I - \Lambda'$ has diagonal entries equal to 1, by (8), the i-th diagonal entry of D must equal $-1/\lambda_{i\pi(i)}$. Hence, if we write $I-\Lambda' = (-\lambda'_{ij})_{i,j\in V}$ with $\lambda'_{ii} = -1$ for $i = 1, \ldots, n$ as before, then

$$-\lambda'_{ij} = \begin{cases} \lambda_{i\pi(j)}/\lambda_{j\pi(j)}, & \text{if } i \neq \pi(j), \\ -1/\lambda_{j\pi(j)}, & \text{if } i = \pi(j). \end{cases}$$

Since $\lambda_{ij} \neq 0$ if and only if $i \to j \in E$, we have that $G' = \pi(G)$ as required.

To prove the sufficiency of our condition, we can simply calculate that $(I - \Lambda)^{-\mathrm{T}}\varepsilon =_d (I - \Lambda')^{-\mathrm{T}}\varepsilon'$, where Λ' is determined as above, and $\varepsilon' =_d (I - \Lambda')^{\mathrm{T}}(I - \Lambda)^{-\mathrm{T}}\varepsilon = DP^{\mathrm{T}}\varepsilon$. □

In our Definition 3.1 of equivalence, we can rearrange the structural equations, and thus transform the graph only if the Λ coefficients we divide by in our procedure are nonzero, which is satisfied whenever $\Lambda \in \mathbb{R}_*^E$. However, if some of the λ_{ij} are zero for an edge $i \to j \in E$, then the random variable does not necessarily lie in the transformed model.

Example 3.6. *Consider the graphs* G *and* G' *in Figure 3. Let* $\Lambda \in \mathbb{R}^E$ *satisfy the sparsity pattern arising from* G *but*

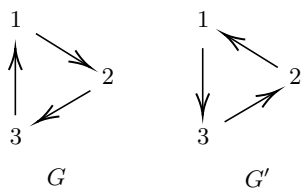

Figure 3: The 3-cycles of Example 3.6

have additional zeros so that it no longer lies in $\mathbb{R}^E_*$*:*

$$\Lambda = \begin{pmatrix} 0 & 3 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Let $\Omega^{(2)} = \Omega^{(3)} = diag(1, 2, 1)$*. Then, the second- and third-order moments* S *and* T *are*

$$S = \begin{pmatrix} 1 & 3 & 0 \\ 3 & 11 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad T_{1..} = \begin{pmatrix} 1 & 3 & 0 \\ 3 & 9 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$T_{2..} = \begin{pmatrix} 3 & 9 & 0 \\ 9 & 29 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad T_{3..} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

and $(S, T) \in \mathcal{M}^{(2,3)}(G)$*. We will show that* $(S, T) \notin \mathcal{M}^{(2,3)}(G')$*. Suppose that there exist*

$$\Lambda' = \begin{pmatrix} 0 & 0 & \lambda'_{13} \\ \lambda'_{21} & 0 & 0 \\ 0 & \lambda'_{32} & 0 \end{pmatrix},$$

$\Omega^{(2)'} = diag(\omega'_{21}, \omega'_{22}, \omega'_{23})$, $\Omega^{(3)'} = diag(\omega'_{31}, \omega'_{32}, \omega'_{33})$ *such that* $S = (I - \Lambda')^{-T}\Omega^{(2)'}(I - \Lambda)^{-1}$ *and* $T = \Omega^{(3)'} \bullet (I - \Lambda')^{-1} \bullet (I - \Lambda')^{-1} \bullet (I - \Lambda')^{-1}$*. Taking inverses,*

$$S^{-1} = \begin{pmatrix} \frac{11}{2} & -\frac{3}{2} & 0 \\ -\frac{3}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} \frac{1}{\omega'_{21}} + \frac{(\lambda'_{13})^2}{\omega'_{23}} & -\frac{\lambda'_{21}}{\omega'_{21}} & -\frac{\lambda'_{13}}{\omega'_{23}} \\ -\frac{\lambda'_{21}}{\omega'_{21}} & \frac{(\lambda'_{21})^2}{\omega'_{21}} + \frac{1}{\omega'_{22}} & -\frac{\lambda'_{32}}{\omega'_{22}} \\ -\frac{\lambda'_{13}}{\omega'_{23}} & -\frac{\lambda'_{32}}{\omega'_{22}} & \frac{(\lambda'_{32})^2}{\omega'_{22}} + \frac{1}{\omega'_{23}} \end{pmatrix}.$$

This leads to the solution

$$\lambda'_{21} = \frac{3}{11}, \lambda'_{13} = 0, \lambda'_{32} = 0, \omega'_{21} = \frac{2}{11}, \omega'_{22} = 11, \omega'_{23} = 1.$$

Equating $T = \Omega^{(3)'} \bullet (I - \Lambda')^{-1} \bullet (I - \Lambda')^{-1} \bullet (I - \Lambda')^{-1}$*, we get*

$$\begin{pmatrix} 1 & 3 & 0 \\ 3 & 9 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} \omega'_{31} + \frac{27\omega'_{32}}{1331} & \frac{9\omega'_{32}}{121} & 0 \\ \frac{9\omega'_{32}}{121} & \frac{3\omega'_{32}}{11} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 3 & 9 & 0 \\ 9 & 29 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{9\omega'_{32}}{121} & \frac{3\omega'_{32}}{11} & 0 \\ \frac{3\omega'_{32}}{11} & \omega'_{32} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \omega'_{33} \end{pmatrix}.$$

This set of linear equations in $\omega'_{31}, \omega'_{32}, \omega'_{33}$ *does not have a solution. For example, we have* $3 = \frac{9\omega'_{32}}{121}, 9 = \frac{3\omega'_{32}}{11}$*, which has no solution. Therefore, we cannot transform the equations so that* $(S, T) \in \mathcal{M}^{(2,3)}(G')$*.*

4 DISCOVERY ALGORITHM FOR CYCLE-DISJOINT GRAPHS

This section presents the concepts and results essential for our discovery algorithm. We begin by formally defining the considered graphs.

Definition 4.1. *A directed graph* G *is* cycle-disjoint *if each one of its nodes belongs to at most one cycle in* G*.*

For example, the graphs in Figure 1 are cycle-disjoint, while those in Figure 2 are not. Note that a cycle-disjoint graph G has all its strong components equal to directed cycles (without any additional edges in the components).

The general Theorem 3.5 allows us to particularly easily characterize all graphs that are distribution equivalent to a graph with disjoint cycles. This can be done by reversing the orientation of any of the disjoint cycles and adjusting the incoming edges.

Our discovery algorithm has two distinct steps:

Step 1. Discover the strong components of the graph G, their topological ordering, and the edges and edge weights within each component.

Step 2. Discover the edges and edge weights between different components.

Below, we prove that these steps can be implemented via algebraic constraints among the second and third moments.

Remark 4.2. Causal discovery algorithms that process algebraic relations among moments can be practically implemented using sample moments calculated from data. Comparing sample quantities to thresholds, one may decide algorithmically whether an algebraic constraint of interest holds. A suitable choice of a threshold renders such algorithms consistent because sample moments consistently estimate true moments when the available sample size tends to infinity.

In the rest of this section we assume that $G = (V, E)$ is a cycle-disjoint graph and that $(S, T) \in \mathcal{M}^{2,3}(G)$ are the second- and third-order moments of a random vector following the LSEM given by G. We will use further notation as follows. In graph $G = (V, E)$, the set of ancestors of a subset $C \subseteq V$, denoted by $\mathrm{anc}(C)$, is the set of nodes $u \in V$ with a directed path (of length at least 0) from u to

some $v \in C$. We say that C is *ancestral* if $\mathrm{anc}(C) = C$. Finally, a node $v \in V$ is a *root (node)* if $\{v\}$ is ancestral and a cycle C is a *root cycle* if it is ancestral.

All complete proofs of the results that follow are provided in the supplementary material A.

4.1 DISCOVERING THE STRONG COMPONENTS

For two vertices $u, v \in V$, we define the determinants

$$d_{uv}^{2\times 2} = \det \begin{pmatrix} s_{uu} & s_{uv} \\ t_{uuu} & t_{uuv} \end{pmatrix},$$
$$d_{uv}^{3\times 3} = \det \begin{pmatrix} s_{uu} & s_{uv} & s_{vv} \\ t_{uuu} & t_{uuv} & t_{uvv} \\ t_{uuv} & t_{uvv} & t_{vvv} \end{pmatrix}.$$

Note that $d_{uv}^{3\times 3} = d_{vu}^{3\times 3}$, but $d_{uv}^{2\times 2} \neq d_{vu}^{2\times 2}$ in general.

Theorem 4.3. *The determinants $d_{uv}^{2\times 2}$ and $d_{uv}^{3\times 3}$ satisfy the following relationships.*

(a) $d_{uv}^{2\times 2} = 0$ identically if and only if u and v have no common ancestors and v is not an ancestor of u;

(b) $d_{uv}^{3\times 3} = 0$ identically if and only if there exists no simple 2-trek $(P_u, P_v) \in \mathcal{T}(u, v)$ with non-empty sides.

By "identically" we mean for any choice of $\Lambda \in \mathbb{R}^E$ and distribution on ε.

We remark that conditions (a) and (b) can be restated using trek-separation [Sullivant et al., 2010]; cf. Lemma A.2.

The previous theorem allows us to identify the root nodes of the graph (Corollary 4.4) as well as a collection of cycles containing all root cycles (Corollary 4.5).

Corollary 4.4. *The determinant $d_{ru}^{2\times 2} = 0$ identically for all $u \in V$ if and only if r is a root of G.*

Corollary 4.5. *Set $\mathcal{C}$ to be the collection of all maximal $C \subseteq V$ such that for all $u, v \in C$, we have $d_{uv}^{3\times 3} = 0$ identically but $d_{uv}^{2\times 2} \neq 0$. Then, $\mathcal{C}$ contains all root cycles.*

To precisely identify the root cycles in the set of candidate root cycles $\mathcal{C}$, we use Proposition 4.7 and Lemma 4.8. To state them, we first need a definition.

Definition 4.6. *We denote by $R_{A,B}$ the set of regression coefficients obtained when regressing the variables in A on the variables in B.*

Proposition 4.7 shows that if we regress all remaining variables on an ancestral set C (e.g., a root cycle), the regression residuals will follow the LSEM corresponding to the graph obtained by removing the vertices C and all their adjacent edges from the graph G, denoted by $G[V \setminus C]$.

Proposition 4.7. *Let $X = (I - \Lambda)^{-T}\varepsilon$ be a random vector whose distribution is in the LSEM given by the graph G. Let $C \subseteq V$ be an ancestral set in G, and define the vector of residuals*

$$X_{V\setminus C.C} := X_{V\setminus C} - R_{V\setminus C,C} X_C.$$

Then the joint distribution of $X_{V\setminus C.C}$ is in the LSEM given by the induced subgraph $G[V \setminus C]$, with

$$X_{V\setminus C.C} = (I - \Lambda_{V\setminus C, V\setminus C})^{-T} \varepsilon_{V\setminus C}.$$

Now, to detect the root cycles within $\mathcal{C}$, we take two candidate root cycles $C, D \in \mathcal{C}$, and regress D on C. If C is indeed a root cycle, then $X_{D.C}$ should be independent of X_C. We check such independence using the third-order moments (assuming existence of skew).

Lemma 4.8. *For $\mathcal{C}$ from Corollary 4.5, and $C, D \in \mathcal{C}$, define the residuals $X_{D.C} = X_D - R_{D,C} X_C$. Then C is a root cycle if and only if the matrix $(\mathbb{E}[X_c^2 (X_{D.C})_d])_{c\in C, d\in D}$ is identically zero for all $D \in \mathcal{C} \setminus C$.*

Therefore, we can recursively identify root nodes and root cycles, regress them away, and continue identifying more root nodes and root cycles until we have gone through all nodes. This process finds not only the strong components, but also a topological ordering among them.

4.2 DISCOVERING ALL EDGES AND EDGE WEIGHTS IN CYCLES

When identifying root cycles as above, we can also identify all edges and edge weights within each such cycle. First, to identify the skeleton of a root cycle, we use the following result, which follows from Drton et al. [2019, Eqn. (6.1)].

Lemma 4.9. *Let C be a root cycle. Two vertices u and v of C are not adjacent if and only if the (i, j)-entry of the inverse of $S_{C,C}$ is identically zero. Equivalently, i and j are not adjacent if and only if $\det S_{C\setminus j, C\setminus i} = 0$ identically.*

Once we have found the skeleton of our root cycle, we choose any one of the two possible orientations. According to Theorem 3.5, each of those is possible in a representative of the equivalence class. We then proceed to find the edge weights on the cycle. The following lemma applies in the case of cycles of length 3 or more.

Lemma 4.10. *Let u, v, w be three consecutive nodes in a root cycle C and λ_{uv} the corresponding edge weight in edge*

$u \to v$. *Then, the matrices*

$$A_{uv}^{(2)} = \begin{pmatrix} 1 & \lambda_{uv} & \lambda_{uv}^2 \\ s_{uu} & s_{uv} & s_{vv} \\ t_{uuu} & t_{uuv} & t_{uvv} \\ t_{uuv} & t_{uvv} & t_{vvv} \end{pmatrix} \text{ and}$$

$$A_{uvw}^{(3)} = \begin{pmatrix} 1 & \lambda_{uv} & \lambda_{uv}^2 & \lambda_{uv}^2 & \lambda_{uv}^3 \\ s_{uu} & s_{uv} & s_{vv} & s_{uw} & s_{vw} \\ t_{uuu} & t_{uuv} & t_{uvv} & t_{uuw} & t_{uvw} \\ t_{uuv} & t_{uvv} & t_{vvv} & t_{uvw} & t_{vvw} \end{pmatrix}$$

have rank at most 2 *and* 3*, respectively. For generic choices of* Λ*,* $\Omega^{(2)}$*, and* $\Omega^{(3)}$*, the respective ranks equal 2 and 3.*

This allows us to find simple linear (for cycles of length 3 or more) or quadratic (for cycles of length 2) equations in the edge weights λ_{uv} along the cycle.

Theorem 4.11. *Let* C *be a root cycle in* G*. Then*

(i) If the cycle length is at least 3, and $u, v, w \in C$ *are three consecutive nodes on* C*, then there is a linear equation in* λ_{uv}*,*

$$p(s,t)\lambda_{uv} = q(s,t), \tag{9}$$

where

$$\begin{aligned} p(s,t) &= s_{uu}(t_{uvw}^2 - t_{uuw}t_{vvw}) + s_{uw}(t_{uuu}t_{vvw} \\ &\quad - t_{uuv}t_{uvw}) + s_{vw}(t_{uuv}t_{uuw} - t_{uuu}t_{uvw}), \\ q(s,t) &= -s_{uv}(t_{uvw}^2 - t_{uuw}t_{vvw}) + s_{uw}(t_{uvv}t_{uvw} \\ &\quad - t_{uuv}t_{vvw}) + s_{vw}(t_{uuv}t_{uvw} - t_{uuw}t_{uvv}). \end{aligned}$$

(ii) If the cycle length is 2 with $u, v \in C$*, then there are two solutions for the edge coefficient* λ_{uv} *given by the quadratic equation*

$$\begin{aligned} &(s_{uu}t_{uuv} - s_{uv}t_{uuu})\lambda_{uv}^2 + (s_{vv}t_{uuu} \\ &- s_{uu}t_{uvv})\lambda_{uv} + s_{uv}t_{uvv} - s_{vv}t_{uuv} = 0. \end{aligned} \tag{10}$$

Note that in the case of cycles of length at least 3, these equations do not depend on the length of the cycle since they only use the moments of three of the vertices.

4.3 DISCOVERING EDGES AND EDGE WEIGHTS BETWEEN STRONG COMPONENTS

Once we have found all strong components, the edges and edge weights within each components, and a topological ordering among the strong components, we proceed to find the edges and edge weights between different strong components. This is done via a simple regression, which uses the topological ordering, and simply regresses each cycle on all cycles before it in topological ordering. The edge weights are then equal to the regression coefficients adjusted by the cycle effect, as shown in the lemma below.

Lemma 4.12. *Let* D *be a cycle in* G *and* $C \subset V$ *the vertices of all cycles that appear before* D *in topological ordering. Then the parameters* λ_{cd} *with* $c \in C$ *and* $d \in D$ *can be calculated using the regression coefficient* $R_{D,C}$ *and the edge weights* $\Lambda_{D,D}$ *within the cycle* D *via*

$$\Lambda_{C,D} = R_{DC}^T(I - \Lambda_{D,D}).$$

PROOF. The result follows directly since $R_{D,C} = (I - \Lambda_{D,D})^{-T}(\Lambda_{C,D})^T$; see Lemma A.4 for more details. □

4.4 ALGORITHM

Combining the results from the previous subsections, we present our full algorithm in Algorithm 1. The implementation details can be found in the supplementary material A.8.1 and the code is available in the repository: github.com/ysamwang/disjointCycles.

Remark 4.2 discusses consistency of the proposed procedure when a given threshold allows one to determine whether quantities are zero or nonzero. In practice, this threshold is not known a priori, so our procedure instead uses a hypothesis test to certify whether each quantity of interest is zero or nonzero. Thus, α, the level for each hypothesis test is a tuning parameter we must specify. In addition, because we perform a large number of tests, we find it empirically advantageous to use either the Bonferroni-Holm [Holm, 1979] or Benjamini-Hochberg [Benjamini and Hochberg, 1995] procedures to control false positives at various stages of the algorithm. Additional details are given in the supplement.

Based on our previous results, we obtain the following.

Corollary 4.13. *Let* G *be a cycle-disjoint graph, and let* $(S,T) \in \mathcal{M}^{2,3}(G)$ *be the second- and third-order moments of a random vector following the LSEM based on* G*. Assume the coefficients* Λ *and the moments* $\Omega^{(2)}$*,* $\Omega^{(3)}$ *of the noise vector* ε *are generic. Then Algorithm 1 recovers a graph* G' *that is distribution equivalent to* G*, as well as appropriate coefficients* Λ' *for* G'*. Applying Theorem 3.5, all graphs distribution equivalent to* G *may be constructed.*

Note that after having recovered G and Λ, we are also able to recover $\Omega^{(2)}$ and $\Omega^{(3)}$.

The genericity assumption needed when applying our algorithm to a particular distribution is that the distribution behaves like a typical member of the model with regards to the tested constraints. Given the polynomial nature of the constraints, only a measure zero set of parameters lead to distributions that violate such genericity.

5 NUMERICAL EXPERIMENTS

Simulation setup We now consider the empirical performance of the proposed procedure. For each replication

Algorithm 1 Causal Graph Discovery

Require: Random vector $X = (X_i)_i$ on n components with moments $S = \mathbb{E}[X_i X_j]$ and $T = \mathbb{E}[X_i X_j X_k]$

Ensure: Causal DG $G = (V, E)$ and edge weights Λ.

Part 1. *Learning strong components, their order, and edges and edge weights within components*

1: Until the graph is nonempty, compute all $d_{uv}^{2\times 2}$ and $d_{uv}^{3\times 3}$.
2: Set $\mathcal{C}$ to be the collection of all vertices $r \in V$ such that such that $d_{ur}^{2\times 2} = 0$ for all $u \in V$, $\mathcal{C}$ is the set of all root nodes (Corollary 4.4). Record $\mathcal{C}$.
3: If $\mathcal{C} \neq \emptyset$, regress $V \setminus \mathcal{C}$ on $\mathcal{C}$ to obtain samples from graph without $\mathcal{C}$ (Proposition 4.7); go back to Step 1.
4: Else, set $\mathcal{C}$ to be the collection of maximal $C \subseteq V$ such that $d_{uv}^{3\times 3} = 0$ and $d_{uv}^{2\times 2} \neq 0$ for all $u, v \in C$
5: Prune $\mathcal{C}$ to obtain $\mathcal{C}'$, all root cycles (Lemma 4.8). Record $\mathcal{C}'$.
6: Compute edges and edge weights along each cycle in $\mathcal{C}'$ (Lemma 4.9 and Theorem 4.11).
7: Regress $V \setminus \mathcal{C}'$ on $\mathcal{C}'$ to obtain samples from graph without $\mathcal{C}'$ (Proposition 4.7); go back to Step 1.

Part 2. *Learning edges and weights between cycles*

8: Working backwards in topological ordering, regress each cycle on all possible parents to find edges between cycles (Lemma 4.12).

we construct a random graph with $|V| = p$. We create $p/3$ disjoint cycles $C_1, C_2, \ldots, C_{p/3}$ in a total ordering where $|C_k| = 3$ for all k. We first include an edge between 1 node in cycle C_k and 1 node in cycle C_{k+1} so that $\mathrm{anc}(C_k) = \cup_{j<k} C_j$. Then for all for $j < k$, and each $u \in C_j$ and $v \in C_k$, we add the edge $u \to v$ with probability $1/2$. The linear coefficients corresponding to each edge in the graph are drawn uniformly from $-(.8, .5) \cup (.5, .8)$. For the errors, we consider two distributions: a mixture of normals with weight .9 on $N(-2, .1^2)$ and weight .1 on $N(2, .1^2)$ and a Gamma(1,1) distribution. We center the distributions, and we scale each ε_v so that the standard deviation is uniformly in $(.8, 1)$. Across the various settings, we let $n = 10000, 25000, 50000, 100000$.

ICA based procedure We first show that the ICA based procedure of Lacerda et al. [2008] suffers computationally when p is not small. Specifically, when the demixing matrix is estimated imprecisely, the procedure requires a search over an intractable number of permutations. To examine this concretely, we let $p = 18, 21, 24, 27$ and record the proportion of times the ICA based procedure returns an estimate within 24 hours. With gamma errors and $n = 100000$, the ICA based procedure—as implemented in the rpy-tetrad package [Ramsey and Andrews, 2023]—returns an estimate in 14 out of 25 replicates when $p = 18$ and 10 out of 25 replicates when $p = 21$. The performance is worse when n is smaller because the demixing matrix is estimated less precisely. When $p = 24, 27$ the procedure fails to return an estimate in all 25 replicates; additional details are given in the appendix.

Proposed procedure To demonstrate that our proposed procedure can handle larger problems, we apply it to problems where $|V| = 30, 45, 60$. In the appendix, we also show results when the graph consists of cycles of size 5. We assess the performance of our proposed procedure through two measures. First, we consider the proportion of times that Part 1 of Algorithm 1 correctly identifies the cycles and their ordering. This ordering is invariant across graphs in the same equivalence class. We also count the number of pairs $u, v \in V \times V$ which are decided correctly; i.e., $u \to v$ vs. $v \to u$ vs. no edge. There may be multiple graphs in both the equivalence class of the ground truth and the equivalence class of the estimated graph. However, for the ground truth, there is a unique graph for which all cycles are stable; i.e, the product of the edges in the graph are less than 1 in absolute value. By construction, this is the graph we used to generate the data. From the equivalence class of the estimated graph, we pick a unique graph in the following way. To orient the skeleton, we estimate the edges for both orientations and select the orientation with the smaller pathweight. When the true and estimated graphs are in the same equivalence class, this should (up to sampling error when estimating the edgeweights) yield the same unique graph. However, when the true and estimated graphs are in different equivalence classes, this may not be the pair of graphs which maximize the number of correct pairs. These measures are shown in Figure 4. For each type of graph and error distribution, we draw 100 replications; for each setting, we show the combination of α and multiple testing correction which yields the best average performance.

The procedure performs better when the errors are mixtures of normals, which has lighter tails, as opposed to gamma. In addition, we see that as p increases performance generally decreases. Nonetheless, even when $p = 60$, nearly 80% of edges are correctly oriented at the largest sample size when the errors are mixtures of normals. In Figure 5, we also show the computational time in seconds. We see that the computation time increases with both n and p. However, even at the largest sample size and $p = 60$, the average time is still under 1.25 hrs per replication.

6 CONCLUSION

This paper studied causal discovery for linear non-Gaussian models that feature feedback loops. Restricting to the setting of cycle-disjoint graphs, we derived algebraic constraints on moments that reveal the strong components of the graph, which then facilitates consistent causal discovery.

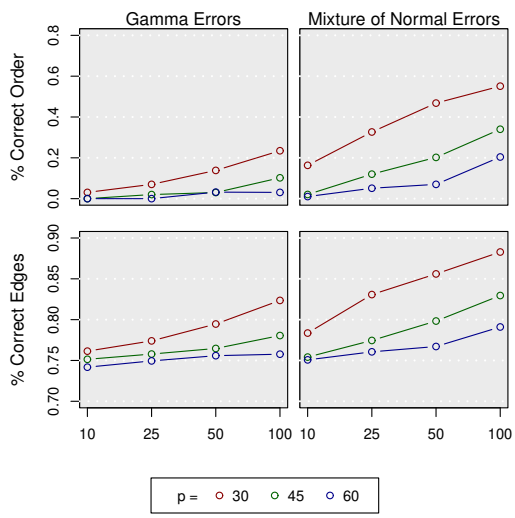

Figure 4: Horizontal axis shows sample size in thousands. Top: Proportion of times the correct ordering is recovered. Bottom: Proportion of edges that are correctly identified. Left: Errors are drawn from Gamma. Right: Errors are drawn from mixture of normals.

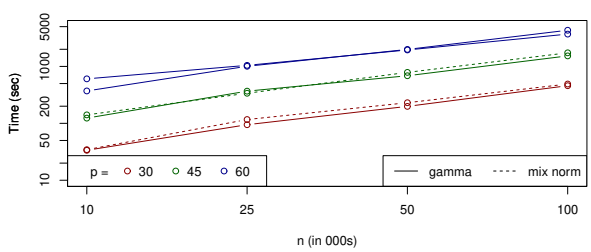

Figure 5: Average time (in seconds) required per replication.

As shown in our experiments, moment-based methods are state-of-the-art for higher-dimensional problems, in which more established ICA-based methods become prohibitively expensive in terms of computation time and memory. From a theoretical perspective, moment-based algorithms also have the appeal of straightforward consistency guarantees.

Our work proves that cycle-disjoint graphs can be learned computationally efficiently. More algebraic insights are needed in order to design moment-based algorithms that could consistently learn non-cycle-disjoint cyclic graphs.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 883818).

References

Jeffrey Adams, Niels Hansen, and Kun Zhang. Identification of partially observed linear causal models: Graphical conditions for the non-Gaussian and heterogeneous cases. In M Ranzato, A Beygelzimer, Y Dauphin, P S Liang, and J Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22822–22833. Curran Associates, Inc., 2021.

Carlos Améndola, Mathias Drton, Alexandros Grosdos, Roser Homs, and Elina Robeva. Third-order moment varieties of linear Non-Gaussian graphical models. *Information and Inference: A Journal of the IMA*, 12(3): 1405–1436, 2023.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 57(1):289–300, 1995.

Ruichu Cai, Zhiyi Huang, Wei Chen, Zhifeng Hao, and Kun Zhang. Causal discovery with latent confounders based on higher-order cumulants. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Haoyue Dai, Ignavier Ng, Yujia Zheng, Zhengqing Gao, and Kun Zhang. Local causal discovery with linear non-Gaussian cyclic models. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 154–162. PMLR, 2024.

Georges Darmois. Analyse générale des liaisons stochastiques: etude particulière de l'analyse factorielle linéaire. *Review of the International Statistical Institute*, 21(1/2): 2–8, 1953.

Mathias Drton, Christopher Fox, Andreas Käufl, and Guillaume Pouliot. The maximum likelihood threshold of a path diagram. *Ann. Statist.*, 47(3):1536–1553, 2019.

AmirEmad Ghassami, Alan Yang, Negar Kiyavash, and Kun Zhang. Characterizing distribution equivalence and structure learning for cyclic and acyclic directed graphs. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3494–3504. PMLR, 2020.

James B Grace, T Michael Anderson, Eric W Seabloom, Elizabeth T Borer, Peter B Adler, W Stanley Harpole, Yann Hautier, Helmut Hillebrand, Eric M Lind, Meelis Pärtel, et al. Integrative modelling reveals mechanisms linking productivity and plant species richness. *Nature*, 529(7586):390–393, 2016.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scand. J. Statist.*, 6(2):65–70, 1979.

Yoshinobu Kawahara, Kenneth Bollen, Shohei Shimizu, and Takashi Washio. Grouplingam: Linear non-Gaussian acyclic models for sets of variables. *CoRR*, abs/1006.5041, 2010.

Gustavo Lacerda, Peter Spirtes, Joseph D. Ramsey, and Patrik O. Hoyer. Discovering cyclic causal models by independent components analysis. In David A. McAllester and Petri Myllymäki, editors, *UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, Helsinki, Finland, July 9-12, 2008*, pages 366–374. AUAI Press, 2008.

Yiheng Liu, Elina Robeva, and Huanqing Wang. Learning linear non-Gaussian graphical models with multidirected edges. *J. Causal Inference*, 9(1):250–263, 2021.

Marloes Maathuis, Mathias Drton, Steffen Lauritzen, and Martin Wainwright. *Handbook of graphical models*. CRC Press, 2018.

Judea Pearl. Linear models: a useful "microscope" for causal analysis. *J. Causal Inference*, 1(1):155–169, 2013.

Joseph Ramsey and Bryan Andrews. Py-tetrad and rpy-tetrad: A new python interface with r support for tetrad causal search. In *Causal Analysis Workshop Series*, pages 40–51. PMLR, 2023.

Thomas Richardson. A discovery algorithm for directed cyclic graphs. In Eric Horvitz and Finn Verner Jensen, editors, *UAI '96: Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence, Reed College, Portland, Oregon, USA, August 1-4, 1996*, pages 454–461. Morgan Kaufmann, 1996a.

Thomas Richardson. A polynomial-time algorithm for deciding equivalence of directed cyclic graphical models. In Eric Horvitz and Finn Verner Jensen, editors, *UAI '96: Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence, Reed College, Portland, Oregon, USA, August 1-4, 1996*, pages 462–469. Morgan Kaufmann, 1996b.

Elina Robeva and Jean-Baptiste Seby. Multi-trek separation in linear structural equation models. *SIAM Journal on Applied Algebra and Geometry*, 5(2):278–303, 2021.

Saber Salehkaleybar, AmirEmad Ghassami, Negar Kiyavash, and Kun Zhang. Learning linear non-Gaussian causal models in the presence of latent variables. *Journal of Machine Learning Research*, 21(39):1–24, 2020.

Daniela Schkoda, Elina Robeva, and Mathias Drton. Causal discovery of linear non-Gaussian causal models with unobserved confounding. *arXiv:2408.04907*, 2024.

Pardis Semnani and Elina Robeva. Causal structure learning in directed, possibly cyclic, graphical models. *Journal of Causal Inference*, 13(1), 2025.

Shohei Shimizu. *Statistical causal discovery: LiNGAM approach*. SpringerBriefs in Statistics. Springer Japan, Tokyo, 2022. JSS Research Series in Statistics.

Shohei Shimizu, Patric. O. Hoyer, Aapo Hyvärinen, and Antti" Kerminen. A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learninig Research*, 7:2003–2030, 2006.

Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O. Hoyer, and Kenneth Bollen. DirectLiNGAM: a direct method for learning a linear non-Gaussian structural equation model. *Journal of Machine Learning Research*, 12(33):1225–1248, 2011.

V. P. Skitovich. On a property of the normal distribution. *Doklady Akademii Nauk SSSR (in Russian)*, 89:217–219, 1953.

Peter Spirtes, Thomas Richardson, Christopher Meek, Richard Scheines, and Clark Glymour. Using path diagrams as a structural equation modeling tool. *Sociological Methods & Research*, 27(2):182–225, 1998.

Seth Sullivant, Kelli Talaska, and Jan Draisma. Trek separation for Gaussian graphical models. *Annals of Statistics*, 38(3):1665–1685, 2010.

Daniele Tramontano, Anthea Monod, and Mathias Drton. Learning linear non-Gaussian polytree models. In James Cussens and Kun Zhang, editors, *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, volume 180 of *Proceedings of Machine Learning Research*, pages 1960–1969. PMLR, 01–05 Aug 2022.

Y. Samuel Wang and Mathias Drton. High-dimensional causal discovery under Non-Gaussianity. *Biometrika*, 107(1):41–59, 2019.

Y. Samuel Wang and Mathias Drton. Causal discovery with unobserved confounding and non-Gaussian data. *Journal of Machine Learning Research*, 24(271):1–61, 2023.

Causal Discovery for Linear Non-Gaussian Causal Models with Disjoint Cycles (Supplementary Material)

Mathias Drton[1] **Marina Garrote-López**[2] **Niko Nikov**[3] **Elina Robeva**[4] **Y. Samuel Wang**[5]

[1]Technical University of Munich and Munich Center for Machine Learning, Munich, Germany
[2]KTH Royal Institute of Technology, Stockholm, Sweden
[3]University of Edinburgh, Edinburgh, UK
[4]University of British Columbia, Vancouver, BC, Canada
[5]Cornell University, Ithaca, NY, USA

A SUPPLEMENT TO SECTION 4

A.1 THE TREK RULE AND TREK SEPARATION

As introduced in Section 2, we use the trek rule to express the entries of S and T in terms of the entries of Λ, $\Omega^{(2)}$, and $\Omega^{(3)}$:

$$s_{ij} = \sum_{(P_i,P_j)\in\mathcal{T}(i,j)} \lambda^{P_i}\lambda^{P_j}\omega^{(2)}_{\mathrm{top}(P_i,P_j)}, \quad \text{and} \quad t_{ijk} = \sum_{(P_i,P_j,P_k)\in\mathcal{T}(i,j,k)} \lambda^{P_i}\lambda^{P_j}\lambda^{P_k}\omega^{(3)}_{\mathrm{top}(P_i,P_j,P_k)},$$

where $\lambda^P = \lambda_{u_1u_2}\lambda_{u_2u_3}\dots\lambda_{u_{r-1}u_r}$ for a path $P = (u_1,\dots,u_r)$. We will use $\mathcal{P}(i,j)$ to denote the set of $i \to j$-paths in G, and $\mathcal{P}^*(i,j) \subseteq \mathcal{P}(i,j)$ to denote those paths that have no repeating vertices. Moreover, the following notation

$$\lambda^{i\to j} = \sum_{P\in\mathcal{P}(i,j)} \lambda^P \quad \text{and} \quad \lambda^{i\to j}_* = \sum_{P\in\mathcal{P}^*(i,j)} \lambda^P$$

will appear in various factorizations of the s_{ij} and t_{ijk}.

We further recall the definition of t-separation from Sullivant et al. [2010].

Definition A.1. *Let $G = (V,E)$ be a directed graph and A, B, C, and D subsets of V. A pair (C,D)* trek-separates *(or t-separates) A from B if for every trek $(P_1,P_2) \in \mathcal{T}(a,b)$ between any $a \in A$ and $b \in B$, either P_1 contains a vertex in C or P_2 contains a vertex in D.*

A.2 TREK-SEPARATION DESCRIPTION OF THEOREM 4.3

Conditions (a) and (b) of Theorem 4.3 can be expressed using t-separation as follows.

Lemma A.2. *Let $G = (V,E)$ be a cycle-disjoint graph. Two vertices $u,v \in V$ have no common ancestor and v is not an ancestor of u if and only if $(\emptyset,u)$ t-separates u from v. Similarly, there exists no simple trek $(P_u,P_v) \in \mathcal{T}(u,v)$ with non-empty sides if and only if (v,u) t-separates u from v.*

PROOF. The nodes u and v share no common ancestor and v is not an ancestor of u if and only if there exists no simple trek $(P_u,P_v) \in \mathcal{T}(u,v)$ with non-empty paths P_u and P_v and there exists no $v \to u$-path. This is equivalent to $(\emptyset,u)$ t-separating u from v

Similarly, there exists no simple trek $(P_u,P_v) \in \mathcal{T}(u,v)$ with non-empty P_u and P_v if and only if the only possible treks between u and v are either a directed path from u to v or a directed path from v to u. This is equivalent to (v,u) t-separating u from v.

A.3 PROOF OF THEOREM 4.3

Before proving Theorem 4.3 we need the following lemma.

Lemma A.3. *G is a cycle-disjoint graph, and $(S,T) \in \mathcal{M}^{(2,3)}(G)$. Let C be a cycle in G and u, v two nodes in C such that (v,u) t-separates u from v. Then,* $d_{uv}^{3\times 3} = \det \begin{pmatrix} s_{uu} & s_{uv} & s_{vv} \\ t_{uuu} & t_{uuv} & t_{uvv} \\ t_{uuv} & t_{uvv} & t_{vvv} \end{pmatrix} = 0.$

PROOF. By the trek rule (7), the entries of the second and third-order moments S and T, can be written as

$$\begin{aligned} s_{ij} &= a_0^{(ij)}\omega_0^{(2)} + a_1^{(ij)}\omega_1^{(2)} + \cdots + a_n^{(ij)}\omega_n^{(2)}, \\ t_{ijk} &= b_0^{(ijk)}\omega_0^{(3)} + b_1^{(ijk)}\omega_1^{(3)} + \cdots + b_n^{(ijk)}\omega_n^{(3)} \end{aligned} \tag{11}$$

where

$$a_\ell^{(ij)} = \sum_{\substack{(P_1,P_2)\in\mathcal{T}(i,j) \\ \mathrm{top}(P_1,P_2)=\ell}} \lambda^{P_1}\lambda^{P_2} = \lambda^{\ell\to i}\lambda^{\ell\to j} \quad \text{and} \quad b_\ell^{(ijk)} = \sum_{\substack{(P_1,P_2,P_3)\in\mathcal{T}(i,j,k) \\ \mathrm{top}(P_1,P_2,P_3)=\ell}} \lambda^{P_1}\lambda^{P_2}\lambda^{P_3} = \lambda^{\ell\to i}\lambda^{\ell\to j}\lambda^{\ell\to k}. \tag{12}$$

We define the following sets of nodes for any two vertices $u, v \in C$

$$\begin{aligned} A_u &:= (\mathrm{anc}(u) \setminus \mathrm{anc}(v)) \cup \{w \in \mathrm{anc}(u) \cap \mathrm{anc}(v) \mid u \in P \text{ for all } P \in \mathcal{P}^*(w,v)\} \text{ and} \\ A_v &:= (\mathrm{anc}(v) \setminus \mathrm{anc}(u)) \cup \{w \in \mathrm{anc}(u) \cap \mathrm{anc}(v) \mid v \in P \text{ for all } P \in \mathcal{P}^*(w,u)\}. \end{aligned} \tag{13}$$

Since (v,u) t-separates u from v, there exists no simple trek $(P_u, P_v) \in \mathcal{T}(u,v)$ with non-empty paths P_u and P_v. Consequently, $A_u \cup A_v = \mathrm{anc}(u) \cup \mathrm{anc}(v)$ and $A_u \cap A_v = \emptyset$. Therefore,

$$\lambda^{\ell\to u} = \begin{cases} \lambda^{\ell\to u} & \text{if } \ell \in A_u \\ \lambda^{\ell\to v}\lambda_*^{v\to u} & \text{if } \ell \in A_v \end{cases} \text{ and } \lambda^{\ell\to v} = \begin{cases} \lambda^{\ell\to u}\lambda_*^{u\to v} & \text{if } \ell \in A_u \\ \lambda^{\ell\to v} & \text{if } \ell \in A_v \end{cases}$$

and then,

$$\begin{aligned} s_{uu} &= \sum_{\ell\in A_u} a_\ell^{(uu)}\omega_\ell^{(2)} + \sum_{\ell\in A_v} a_\ell^{(uu)}\omega_\ell^{(2)} = \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \omega_\ell^{(2)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to u}\right)^2 \omega_\ell^{(2)} = \\ &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \omega_\ell^{(2)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^2 \left(\lambda_*^{v\to u}\right)^2 \omega_\ell^{(2)}, \\ s_{uv} &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \lambda_*^{u\to v}\omega_\ell^{(2)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^2 \lambda_*^{v\to u}\omega_\ell^{(2)} \text{ and} \\ s_{vv} &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \left(\lambda_*^{u\to v}\right)^2 \omega_\ell^{(2)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^2 \omega_\ell^{(2)}. \end{aligned}$$

Similarly,

$$\begin{aligned} t_{uuu} &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^3 \omega_\ell^{(3)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^3 \left(\lambda_*^{v\to u}\right)^3 \omega_\ell^{(3)}, \\ t_{uuv} &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^3 \lambda_*^{u\to v}\omega_\ell^{(3)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^3 \left(\lambda_*^{v\to u}\right)^2 \omega_\ell^{(3)}, \\ t_{uvv} &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^3 \left(\lambda_*^{u\to v}\right)^2 \omega_\ell^{(3)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^3 \lambda_*^{v\to u}\omega_\ell^{(3)} \text{ and} \\ t_{vvv} &= \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^3 \left(\lambda_*^{u\to v}\right)^3 \omega_\ell^{(3)} + \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^3 \omega_\ell^{(3)}. \end{aligned}$$

For brevity, we define $L_u^{(2)} := \sum_{\ell \in A_u} \left(\lambda^{\ell \to u}\right)^2 \omega_\ell^{(2)}$, $L_v^{(2)} := \sum_{\ell \in A_v} \left(\lambda^{\ell \to v}\right)^2 \omega_\ell^{(2)}$, $L_u^{(3)} := \sum_{\ell \in A_u} \left(\lambda^{\ell \to u}\right)^3 \omega_\ell^{(3)}$ and $L_v^{(3)} := \sum_{\ell \in A_v} \left(\lambda^{\ell \to v}\right)^3 \omega_\ell^{(3)}$. Then the matrix

$$\begin{pmatrix} s_{uu} & s_{uv} & s_{vv} \\ t_{uuu} & t_{uuv} & t_{uvv} \\ t_{uuv} & t_{uvv} & t_{vvv} \end{pmatrix} = \begin{pmatrix} L_u^{(2)} & \lambda_*^{u \to v} L_u^{(2)} & (\lambda_*^{u \to v})^2 L_u^{(2)} \\ L_u^{(3)} & \lambda_*^{u \to v} L_u^{(3)} & (\lambda_*^{u \to v})^2 L_u^{(3)} \\ \lambda_*^{u \to v} L_u^{(3)} & (\lambda_*^{u \to v})^2 L_u^{(3)} & (\lambda_*^{u \to v})^3 L_u^{(3)} \end{pmatrix} + \tag{14}$$

$$\begin{pmatrix} (\lambda_*^{v \to u})^2 L_v^{(2)} & \lambda_*^{v \to u} L_v^{(2)} & L_v^{(2)} \\ (\lambda_*^{v \to u})^3 L_v^{(2)} & (\lambda_*^{v \to u})^2 L_v^{(2)} & \lambda_*^{v \to u} L_v^{(2)} \\ (\lambda_*^{v \to u})^2 L_v^{(3)} & \lambda_*^{v \to u} L_v^{(3)} & L_v^{(3)} \end{pmatrix} \tag{15}$$

has rank at most 2 since it is the sum of two rank 1 matrices. Therefore $d_{uv}^{3\times 3} = 0$.

We now proceed with the proof of Theorem 4.3.

PROOF OF THEOREM 4.3. We begin by proving the converse of (b): assume there is no trek simple (P_u, P_v) with non-empty sides. Without loss of generality, assume that there is a path from u to v. If there also exists a path from v to u then u and v belong to a cycle and then $d_{uv}^{3\times 3} = 0$ by Lemma A.3.

Therefore, we assume there is no path from v to u. In this case,

$$\lambda^{\ell \to v} = \begin{cases} \lambda^{\ell \to u} \lambda^{u \to v} & \text{if } \ell \in \mathrm{anc}(u) \\ \lambda^{\ell \to v} & \text{if } \ell \notin \mathrm{anc}(u). \end{cases}$$

Following the same notation as in (11) and (12) we have that

$$a_\ell^{(uu)} = \left(\lambda^{\ell \to u}\right)^2, \quad a_\ell^{(uv)} = \left(\lambda^{\ell \to u}\right)^2 \lambda^{u \to v} = \lambda^{u \to v} a_\ell^{(uu)} \text{ and}$$
$$a_\ell^{(vv)} = \left(\lambda^{\ell \to v}\right)^2 = \begin{cases} (\lambda^{u \to v})^2 a_\ell^{(uu)} & \text{if } \ell \in \mathrm{anc}(u) \\ \left(\lambda^{\ell \to v}\right)^2 & \text{if } \ell \notin \mathrm{anc}(u). \end{cases}$$

Therefore,

$$s_{uv} = \lambda^{u \to v} s_{uu}, \quad \text{and} \quad s_{vv} = (\lambda^{u \to v})^2 s_{uu} + \sum_{\substack{\ell \in V \\ \ell \notin \mathrm{anc}(u)}} \left(\lambda^{\ell \to v}\right)^2 \omega_\ell^{(2)}.$$

Similarly, it is straightforward to check that

$$t_{uuv} = \lambda^{u \to v} t_{uuu}, \quad t_{uvv} = (\lambda^{u \to v})^2 t_{uuu}, \quad \text{and} \quad t_{vvv} = (\lambda^{u \to v})^3 t_{uuu} + \sum_{\substack{\ell \in V \\ \ell \notin \mathrm{anc}(u)}} \left(\lambda^{\ell \to v}\right)^3 \omega_\ell^{(3)}.$$

Therefore, the matrix

$$\begin{pmatrix} s_{uu} & s_{uv} & s_{vv} \\ t_{uuu} & t_{uuv} & t_{uvv} \\ t_{uuv} & t_{uvv} & t_{vvv} \end{pmatrix} = \begin{pmatrix} s_{uu} & \lambda^{u \to v} s_{uu} & (\lambda^{u \to v})^2 s_{uu} \\ t_{uuu} & \lambda^{u \to v} t_{uuu} & (\lambda^{u \to v})^2 t_{uuu} \\ t_{uuv} & \lambda^{u \to v} t_{uuv} & (\lambda^{u \to v})^2 t_{uuv} \end{pmatrix} + \begin{pmatrix} 0 & 0 & \sum_{\ell \in V, \ell \notin \mathrm{anc}(u)} \left(\lambda^{\ell \to v}\right)^2 \omega_\ell^{(2)} \\ 0 & 0 & 0 \\ 0 & 0 & \sum_{\ell \in V, \ell \notin \mathrm{anc}(u)} \left(\lambda^{\ell \to v}\right)^3 \omega_\ell^{(3)} \end{pmatrix}$$

has rank 2 as it is the sum of two rank 1 matrices, and consequently $d_{uv}^{3\times 3} = 0$.

The converse of (a) follows directly. Given that there exists no $v \to u$-path and u and v share no common ancestor, the 2×2 submatrix

$$\begin{pmatrix} s_{uu} & s_{uv} \\ t_{uuu} & t_{uuv} \end{pmatrix} = \begin{pmatrix} s_{uu} & \lambda^{u \to v} s_{uu} \\ t_{uuu} & \lambda^{u \to v} t_{uuu} \end{pmatrix}$$

has rank 1. Therefore $d_{uv}^{2\times 2} = 0$.

We will now prove the direct implications, beginning with (b) and then proceeding to (a). First, assume there exists a simple trek $\tau = (P_1, P_2) \in \mathcal{T}(u, v)$ with $\mathrm{top}(\tau) = \ell$ such that P_1 and P_2 are non-empty Furthermore, assume that P_1

and P_2 are the shortest disjoint paths from ℓ to u and v, respectively, containing only distinct vertices. Let P_1 consist of vertices $\ell = u_0, u_1, \ldots, u_k = u$ with edges $u_i \to u_{i+1}$, $i = 1, \ldots, k-1$ and P_2 of vertices $\ell = v_0, v_1, \ldots, v_s = v$ with edges $v_i \to v_{i+1}$, $i = 1, \ldots, s-1$. Set $\lambda_{v_0,v_1} = \lambda_{u_0,u_1} = 2$, $\lambda_{u_i,u_{i+1}} = \lambda_{v_j,v_{j+1}} = 1$ for $i = 1, \ldots, k-1$ and $j = 1, \ldots, s-1$ and set all other λ's to 0. Also, set $\omega^{(2)}_{u_i} = \omega^{(3)}_{u_i} = 1$ and $\omega^{(2)}_{v_j} = \omega^{(3)}_{v_j} = 1$ for $i = 0, \ldots, k$ and $j = 0, \ldots, s$. Moreover, define $S^{(2)}_u := \sum_{i=1}^k \left(\lambda^{u_i \to u}\right)^2 \omega^{(2)}_{u_i}$, $S^{(3)}_u := \sum_{i=1}^k \left(\lambda^{u_i \to u}\right)^3 \omega^{(3)}_{u_i}$, $S^{(2)}_v := \sum_{i=1}^k \left(\lambda^{v_i \to v}\right)^2 \omega^{(2)}_{v_i}$ and $S^{(3)}_v := \sum_{i=1}^k \left(\lambda^{v_i \to v}\right)^3 \omega^{(3)}_{v_i}$, then

$$
\begin{aligned}
d^{3\times 3}_{uv} &= \det \begin{pmatrix} \left(\lambda^{\ell\to u}\right)^2 \omega^{(2)}_\ell + S^{(2)}_u & \lambda^{\ell\to u}\lambda^{\ell\to v}\omega^{(2)}_\ell & \left(\lambda^{\ell\to v}\right)^2 \omega^{(2)}_\ell + S^{(2)}_v \\ \left(\lambda^{\ell\to u}\right)^3 \omega^{(3)}_\ell + S^{(3)}_u & \left(\lambda^{\ell\to u}\right)^2 \lambda^{\ell\to v}\omega^{(3)}_\ell & \lambda^{\ell\to u}\left(\lambda^{\ell\to v}\right)^2 \omega^{(3)}_\ell \\ \left(\lambda^{\ell\to u}\right)^2 \lambda^{\ell\to v}\omega^{(3)}_\ell & \lambda^{\ell\to u}\left(\lambda^{\ell\to v}\right)^2 \omega^{(3)}_\ell & \left(\lambda^{\ell\to v}\right)^3 \omega^{(3)}_\ell + S^{(3)}_v \end{pmatrix} \\
&= \det \begin{pmatrix} 2^2+k & 2^2 & 2^2+k \\ 2^3+k & 2^3 & 2^3 \\ 2^3 & 2^3 & 2^3+k \end{pmatrix} = 12k^2 > 0.
\end{aligned}
$$

Moreover,

$$
d^{2\times 2}_{uv} = \det \begin{pmatrix} 2^2+k & 2^2 \\ 2^3+k & 2^3 \end{pmatrix} = 2^2 k > 0.
$$

Therefore, $d^{2\times 2}_{uv}$ and $d^{3\times 3}_{uv}$ are not zero for all values of λ and ω, leading to a contradiction. Thus, there exist no simple trek $\tau = (P_1, P_2) \in \mathcal{T}(u, v)$ with $\mathrm{top}(\tau) = \ell$ such that P_1 and P_2 are non/empty.

Alternatively, if there is a $v \to u$ path, then P_1 consist of vertices $v = u_0, u_1, \ldots, u_k = u$ with edges $u_i \to u_{i+1}$, $i = 1, \ldots, k-1$ and P_2 is just the vertex v. Using the same parameters and notation as above, we also have that

$$
d^{2\times 2}_{uv} = \det \begin{pmatrix} \left(\lambda^{v\to u}\right)^2 \omega^{(2)}_v + S^{(2)}_u & \lambda^{v\to u}\lambda^{v\to v}\omega^{(2)}_v \\ \left(\lambda^{v\to u}\right)^3 \omega^{(3)}_v + S^{(3)}_u & \left(\lambda^{v\to u}\right)^2 \lambda^{v\to v}\omega^{(3)}_v \end{pmatrix} = \det \begin{pmatrix} 2^2+k & 2^2 \\ 2^3+k & 2^3 \end{pmatrix} = 2^2 k > 0,
$$

which concludes the proof.

A.4 PROOF OF PROPOSITION 4.7

The following lemma, necessary for proving Proposition 4.7, provides a closed-form expression for regression coefficients when a subset of nodes is regressed onto an ancestral set that contains all of its ancestors.

Lemma A.4. *Let $G = (V, E)$ be a simple cyclic graph with disjoint subsets of nodes $C, D \subseteq V$ such that C is an ancestral set with $\mathrm{anc}(D) \subseteq C \cup D$. Let $R_{D,C} = S_{D,C} S^{-1}_{C,C}$ be the regressing coefficients when D is regressed onto C. Then*

$$
R_{D,C} = (I - \Lambda_{D,D})^{-T} (\Lambda_{C,D})^T.
$$

PROOF.

Since C is ancestral, C and D are disjoints and $\mathrm{anc}(D) \subseteq C \cup D$, there are no edges from D to C nor from $V \setminus (C \cup D)$ to C or D. Therefore, the matrix Λ can be written in block form as

$$
\Lambda = \begin{pmatrix} \Lambda_{C,C} & \Lambda_{C,D} & \Lambda_{C,V\setminus(C\cup D)} \\ 0 & \Lambda_{D,D} & \Lambda_{D,V\setminus(C\cup D)} \\ 0 & 0 & \Lambda_{V\setminus(C\cup D),V\setminus(C\cup D)} \end{pmatrix},
$$

where $V \setminus (C \cup D)$ can be empty. Hence, the corresponding path matrix $(I - \Lambda)^{-1}$ has the form

$$
(I - \Lambda)^{-1} = \begin{pmatrix} (I - \Lambda_{C,C})^{-1} & M_1 & M_2 \\ 0 & (I - \Lambda_{D,D})^{-1} & M_3 \\ 0 & 0 & (I - \Lambda_{V\setminus(C\cup D),V\setminus(C\cup D)})^{-1} \end{pmatrix}.
$$

The regressing coefficients when D is regressed onto C are defined as $R_{D,C} = S_{D,C}S_{C,C}^{-1}$, where

$$S_{D,C} = \left((I-\Lambda)^{-T}\right)_{D,V} \Omega^{(2)} \left((I-\Lambda)^{-1}\right)_{V,C} = \begin{pmatrix} M_1^T & (I-\Lambda_{D,D})^{-T} & 0 \end{pmatrix} \Omega^{(2)} \begin{pmatrix} (I-\Lambda_{C,C})^{-1} \\ 0 \\ 0 \end{pmatrix}$$
$$= M_1^T \Omega^{(2)}_{C,C} (I-\Lambda_{C,C})^{-1},$$

and

$$S_{C,C} = \left((I-\Lambda)^{-T}\right)_{C,V} \Omega^{(2)} \left((I-\Lambda)^{-1}\right)_{V,C} = \begin{pmatrix} (I-\Lambda_{C,C})^{-T} & 0 & 0 \end{pmatrix} \Omega^{(2)} \begin{pmatrix} (I-\Lambda_{C,C})^{-1} \\ 0 \\ 0 \end{pmatrix}$$
$$= (I-\Lambda_{C,C})^{-T} \Omega^{(2)}_{C,C} (I-\Lambda_{C,C})^{-1}.$$

Therefore, we obtain that

$$R_{D,C} = M_1^T (I-\Lambda_{C,C})^T.$$

Our claim that $R_{D,C} = (I-\Lambda_{D,D})^{-T}(\Lambda_{C,D})^T$ now follows because

$$M_1^T = (I-\Lambda_{D,D})^{-T}(\Lambda_{C,D})^T(I-\Lambda_{C,C})^{-T},$$

which can be deduced from

$$0 = \left[(I-\Lambda)(I-\Lambda)^{-1}\right]_{C,D} = (I-\Lambda_{C,C})M_1 - \Lambda_{C,D}(I-\Lambda_{D,D})^{-1}.$$

PROOF OF PROPOSITION 4.7. Observe that Lemma A.4 applies here, thus, $R_{V\setminus C,C} = (I-\Lambda_{V\setminus C,V\setminus C})^{-T}(\Lambda_{C,V\setminus C})^T$. Therefore,

$$X_{V\setminus C.C} = X_{V\setminus C} - R_{V\setminus C,C}X_C = X_{V\setminus C} - (I-\Lambda_{V\setminus C,V\setminus C})^{-T}(\Lambda_{C,V\setminus C})^T X_C.$$

Since there are no edges from $V \setminus C$ to C,

$$\begin{pmatrix} \varepsilon_C \\ \varepsilon_{V\setminus C} \end{pmatrix} = (I-\Lambda^T)X = \begin{pmatrix} I-\Lambda_{C,C}^T & 0 \\ -\Lambda_{C,V\setminus C}^T & I-\Lambda_{V\setminus C,V\setminus C}^T \end{pmatrix} \begin{pmatrix} X_C \\ X_{V\setminus C} \end{pmatrix}$$
$$= \begin{pmatrix} (I-\Lambda_{C,C}^T)X_C \\ -\Lambda_{C,V\setminus C}^T X_C + X_{V\setminus C} - \Lambda_{V\setminus C,V\setminus C}^T X_{V\setminus C} \end{pmatrix}.$$

Hence,

$$X_{V\setminus C.C} = X_{V\setminus C} - (I-\Lambda_{V\setminus C,V\setminus C})^{-T}(\Lambda_{C,V\setminus C})^T X_C$$
$$= (I-\Lambda_{V\setminus C,V\setminus C})^{-T}\varepsilon_{V\setminus C}.$$

A.5 PROOF OF LEMMA 4.8

PROOF. Let $\mathcal{C}$ be the set containing all root cycles in G constructed as described in Corollary 4.5. Let $C, D \in \mathcal{C}$ be two distinct sets. If C is a root cycle, then it is an ancestral set, and thus $X_C = (I-\Lambda_{C,C})^{-T}\varepsilon_C$. By Proposition 4.7, we have that $X_{D.C} = (X_{V\setminus C.C})_D = ((I-\Lambda_{V\setminus C,V\setminus C})^{-T}\varepsilon_{V\setminus C})_D$ depends only on $\varepsilon_{V\setminus C}$. This implies that X_C and $X_{D.C}$ are independent. Then, for all $c \in C$ and $d \in D$, we have that

$$0 = \mathbb{E}[X_c^2 (X_{D.C})_d] = \mathbb{E}[X_c^2 (X_d - R_{d,C}X_C)].$$

Conversely, suppose C is not a root cycle. Then there must be a cycle $D \in \mathcal{C}$ such that there is a directed path from a node $d \in D$ to a node $c \in C$. Take the shortest possible path from d to c so that no interior node of the path is in C.

Specialize all weights λ_{uv} to zero except for those on the path. Then $(R_{D,C})_{dc} = \frac{s_{dc}}{s_{cc}}$ and

$$\mathbb{E}[X_c^2 (X_{D.C})_d] = \mathbb{E}[X_c^2 (X_d - R_{d,c}X_c)] = \mathbb{E}[X_c^2 X_d] - R_{d,C}\mathbb{E}[X_c^3] = t_{ccd} - \frac{s_{dc}}{s_{cc}} t_{ccc} = 0$$

if and only if $s_{cc}t_{ccd} - s_{dc}t_{ccc} = 0$ or equivalently, $d_{cd}^{2\times 2} = 0$. However, by Theorem 4.3(a), $d_{cd}^{2\times 2}$ is not identically zero if d is an ancestor of c.

A.6 PROOF OF LEMMA 4.10

PROOF. Let C be a root cycle and let $u,\ v,\ w \in C$ be three consecutive nodes. Since C is a root cycle and there is an edge $u \to v$, the sets A_u and A_v defined in (13) are $A_u = C \setminus v$ and $A_v = \{v\}$. Therefore, following a similar construction as in Lemma A.3 we obtain

$$A^{(2)}_{uv} = \begin{pmatrix} 1 & \lambda_{uv} & \lambda^2_{uv} \\ s_{uu} & s_{uv} & s_{vv} \\ t_{uuu} & t_{uuv} & t_{uvv} \\ t_{uuv} & t_{uvv} & t_{vvv} \end{pmatrix} = \begin{pmatrix} 1 & \lambda_{uv} & \lambda^2_{uv} \\ L^{(2)}_u & \lambda_{uv}L^{(2)}_u & \lambda^2_{uv}L^{(2)}_u \\ L^{(3)}_u & \lambda_{uv}L^{(3)}_u & \lambda^2_{uv}L^{(3)}_u \\ \lambda_{uv}L^{(3)}_u & \lambda^2_{uv}L^{(3)}_u & \lambda^3_{uv}L^{(3)}_u \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ (\lambda^{v\to u}_*)^2 L^{(2)}_v & \lambda^{v\to u}_* L^{(2)}_v & L^{(2)}_v \\ (\lambda^{v\to u}_*)^3 L^{(2)}_v & (\lambda^{v\to u}_*)^2 L^{(2)}_v & \lambda^{v\to u}_* L^{(2)}_v \\ (\lambda^{v\to u}_*)^2 L^{(3)}_v & \lambda^{v\to u}_* L^{(3)}_v & L^{(3)}_v \end{pmatrix}$$

where $L^{(2)}_u := \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \omega^{(2)}_\ell$, $L^{(2)}_v := \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^2 \omega^{(2)}_\ell$, $L^{(3)}_u := \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^3 \omega^{(3)}_\ell$ and $L^{(3)}_v := \sum_{\ell\in A_v} \left(\lambda^{\ell\to v}\right)^3 \omega^{(3)}_\ell$. Consequently, $A^{(2)}_{uv}$ has rank at most 2 since it is the sum of two rank 1 matrices.

Moreover, since $\lambda^{v\to w} = \lambda^{v\to v}\lambda_{vw}$ we have

$$s_{uw} = \sum_{\ell\in A_u} \lambda^{\ell\to u}\lambda^{\ell\to w}\omega^{(2)}_\ell + \lambda^{v\to u}\lambda^{v\to w}\omega^{(2)}_v = L^{(2)}_{uw\setminus v} + \lambda^{v\to u}_*\lambda_{vw}L^{(2)}_v$$
$$s_{vw} = \sum_{\ell\in A_u} (\lambda^{\ell\to u}\lambda_{uv})\lambda^{\ell\to w}\omega^{(2)}_\ell + \lambda^{v\to v}\lambda^{v\to w}\omega^{(2)}_v = \lambda_{uv}L^{(2)}_{uw\setminus v} + \lambda_{vw}L^{(2)}_v$$

where $L^{(2)}_{uw} := \sum_{\ell\in A_u} \lambda^{\ell\to u}\lambda^{\ell\to w}\omega^{(2)}_\ell$. Defining $L^{(3)}_{uuw} := \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \lambda^{\ell\to w}\omega^{(2)}_\ell$ we also have that

$$t_{uuw} = \sum_{\ell\in A_u} \left(\lambda^{\ell\to u}\right)^2 \lambda^{\ell\to w}\omega^{(3)}_\ell + (\lambda^{v\to u})^2 \lambda^{v\to w}\omega^{(3)}_v = L^{(3)}_{uuw} + (\lambda^{v\to u}_*)^2\lambda_{vw}L^{(3)}_v$$
$$t_{uvw} = \sum_{\ell\in A_u} \lambda^{\ell\to u}\lambda^{\ell\to v}\lambda^{\ell\to w}\omega^{(3)}_\ell + \lambda^{v\to u}\lambda^{v\to v}\lambda^{v\to w}\omega^{(3)}_v = \lambda_{uv}L^{(3)}_{uuw} + \lambda^{v\to u}_*\lambda_{vw}L^{(3)}_v$$
$$t_{vvw} = \sum_{\ell\in A_u} \left(\lambda^{\ell\to v}\right)^2 \lambda^{\ell\to w}\omega^{(3)}_\ell + (\lambda^{v\to v})^2 \lambda^{v\to w}\omega^{(3)}_v = \lambda^2_{uv}L^{(3)}_{uuw} + \lambda_{vw}L^{(3)}_v.$$

Therefore,

$$A^{(3)}_{uv} = \begin{pmatrix} 1 & \lambda_{uv} & \lambda^2_{uv} & 0 & 0 \\ L^{(2)}_u & \lambda_{uv}L^{(2)}_u & \lambda^2_{uv}L^{(2)}_u & 0 & 0 \\ L^{(3)}_u & \lambda_{uv}L^{(3)}_u & \lambda^2_{uv}L^{(3)}_u & 0 & 0 \\ \lambda_{uv}L^{(3)}_u & \lambda^2_{uv}L^{(3)}_u & \lambda^3_{uv}L^{(3)}_u & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & \lambda^2_{uv} & \lambda^3_{uv} \\ 0 & 0 & 0 & L^{(2)}_{uw} & \lambda_{uv}L^{(2)}_{uw} \\ 0 & 0 & 0 & L^{(3)}_{uuw} & \lambda_{uv}L^{(3)}_{uuw} \\ 0 & 0 & 0 & \lambda_{uv}L^{(3)}_{uuw} & \lambda^2_{uv}L^{(3)}_{uuw} \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ (\lambda^{v\to u}_*)^2L^{(2)}_v & \lambda^{v\to u}_*L^{(2)}_v & L^{(2)}_v & \lambda^{v\to u}_*\lambda_{vw}L^{(2)}_v & \lambda_{vw}L^{(2)}_v \\ (\lambda^{v\to u}_*)^3L^{(3)}_v & (\lambda^{v\to u}_*)^2L^{(3)}_v & \lambda^{v\to u}_*L^{(3)}_v & (\lambda^{v\to u}_*)^2\lambda_{vw}L^{(3)}_v & \lambda^{v\to u}_*\lambda_{vw}L^{(3)}_v \\ (\lambda^{v\to u}_*)^2L^{(3)}_v & \lambda^{v\to u}_*L^{(3)}_v & L^{(3)}_v & \lambda^{v\to u}_*\lambda_{vw}L^{(3)}_v & \lambda_{vw}L^{(3)}_v \end{pmatrix}. \tag{16}$$

Consequently, $A^{(3)}_{uvw}$ has rank at most 3 since it is the sum of three rank 1 matrices.

Finally, the matrices $A^{(2)}_{uv}$ and $A^{(3)}_{uvw}$ have ranks 2 and 3 respectively, for generic choices of the parameters, provided their determinants are not identically zero. To establish this, it suffices to demonstrate a single set of parameters for which their determinants are nonzero. Take $\lambda_{uv} = 2$ and $\lambda_{ij} = 1$ for all other edge $i \to j$ in C. Also, set $\omega^{(2)}_i = \omega^{(3)}_i = 1$ for all $i \in C$. Straightforward computations show that, for this choice of parameters, we have

$$L^{(2)}_u = k-1, \quad L^{(2)}_u = -k+1, \quad L^{(2)}_{uw} = 2k-3, \quad L^{(2)}_{uuw} = -2k+3,$$

$$L_v^{(2)} = 1, \quad L_v^{(3)} = -1, \quad \frac{\lambda_*^{v\to v}}{\lambda_{uv}} = 1.$$

Then, the matrix

$$A_{uv}^{(2)} = \begin{pmatrix} 1 & 2 & 4 \\ k & 2k-1 & 4k-3 \\ -k & -2k+1 & -4k+3 \\ -2k+1 & -4k+3 & -8k+7 \end{pmatrix}$$

has rank 2 since all its 3×3 minors vanish and $\det \begin{pmatrix} 1 & 2 \\ k & 2k-1 \end{pmatrix} = -1 \neq 0$. Similarly,

$$A_{uv}^{(2)} = \begin{pmatrix} 1 & 2 & 4 & 4 & 8 \\ k & 2k-1 & 4k-3 & 2k-2 & 4k-5 \\ -k & -2k+1 & -4k+3 & -2k+2 & -4k+5 \\ -2k+1 & -4k+3 & -8k+7 & -4k+5 & -8k+11 \end{pmatrix}$$

has rank 3 since all its 4×4 minors vanish and $\det \begin{pmatrix} 1 & 2 & 4 \\ k & 2k-1 & 2k-2 \\ -2k+1 & -4k+3 & -4k+5 \end{pmatrix} = -2k+1 \neq 0$ for $k \geq 1$.

A.7 PROOF OF THEOREM 4.11

PROOF. The proof of (i) follows from the decomposition of $A_{uv}^{(3)}$ given in (16).

Consider the submatrix A of $A_{uv}^{(3)}$ formed by taking the last three rows, that is

$$A := \begin{pmatrix} s_{uu} & s_{uv} & s_{vv} & s_{uw} & s_{vw} \\ t_{uuu} & t_{uuv} & t_{uvv} & t_{uuw} & t_{uvw} \\ t_{uuv} & t_{uvv} & t_{vvv} & t_{uvw} & t_{vvw} \end{pmatrix}.$$

Let $A = M + N + O$ be the decomposition of A as in(16). Thus, M, N and O are rank-1 matrices formed by the last three rows of the corresponding matrices in (16), taken in the same order. Denote by A_i the i-th column of A (and similarly for M, N and O). Define $q(s,t) := \det(A_2\ A_4\ A_5)$ and $p(s,t) := \det(A_1\ A_4\ A_5)$. Then

$$\begin{aligned} q(s,t) &= \det(A_2\ A_4\ A_5) = \det(M_2 + O_2 \quad N_4 + O_4 \quad N_5 + O_5) \\ &= \det(M_2 \quad N_4 + O_4 \quad N_5 + O_5) + \det(O_2 \quad N_4 + O_4 \quad N_5 + O_5) = \det(M_2 \quad N_4 + O_4 \quad N_5 + O_5), \end{aligned} \tag{17}$$

where the second to last equality follows from the multilinearity of the determinant with respect to its columns, and the last equality follows because $\det(O_2 \quad N_4 + O_4 \quad N_5 + O_5)$ is zero as it is the determinant of the sum of two rank-1 matrices.

Similarly,

$$p(s,t) = \det(A_1\ A_4\ A_5) = \det(M_1 \quad N_4{+}O_4 \quad N_5{+}O_5){+}\det(O_1 \quad N_4{+}O_4 \quad N_5{+}O_5) = \det(M_1 \quad N_4{+}O_4 \quad N_5{+}O_5)$$

since $\det(O_1 \quad N_4 + O_4 \quad N_5 + O_5)$ is zero as it also is the determinant of the sum of two rank-1 matrices.

Finally, since $M_2 = \lambda_{uv} M_1$ we have

$$q(s,t) = \det(M_2 \quad N_4 + O_4 \quad N_5 + O_5) = \lambda_{uv} \det(M_1 \quad N_4 + O_4 \quad N_5 + O_5) = \lambda_{uv} p(s,t).$$

Straightforward computations show that the polynomials $q(s,t) = \det(A_2\ A_4\ A_5)$ and $p(s,t) = \det(A_1\ A_4\ A_5)$ correspond to the polynomials in the statement of Theorem Theorem 4.11.

To prove (ii), assume C has only 2 nodes u, v. The quadratic equation (10) on λ_{uv} follows from the fact that the 3 minors of $A_{uv}^{(2)}$ vanish. Specifically, taking the determinant of the first three rows of $A_{uv}^{(2)}$ yields

$$(s_{uu}t_{uuv} - s_{uv}t_{uuu})\lambda_{uv}^2 + (s_{vv}t_{uuu} - s_{uu}t_{uvv})\lambda_{uv} + s_{uv}t_{uvv} - s_{vv}t_{uuv}.$$

Furthermore, the coefficient of λ_{uv}^2, $s_{uu}t_{uuv} - s_{uv}t_{uuu}$, is nonzero (see Equation (14)).

A.8 CAUSAL GRAPH DISCOVERY ALGORITHM

A.8.1 Implementation details

We now give details of the implementation of Algorithm 1.

- Line 2: To certify if r is a root we test $H_0 : d_{ur}^{2x2} = 0 \quad \forall u \neq r$. Specifically, for each $u \neq r$ we calculate a p-value for $H_0 : d_{ur}^{2x2} = 0$ which is calibrated using asymptotic distribution of $\hat{d}_{ur}^{2x2}$ derived via the delta method. We then control for multiple testing across the $p(p-1)$ individual tests.
- Line 4: To find all maximal sets C such that $d_{uv}^{3\times3} = 0, d_{uv}^{2x2} \neq 0, d_{vu}^{2x2} \neq 0)$ for all $u, v \in C$ we do the following. To test if $d_{uv}^{3\times3} = 0$ we calculate a p-value for $H_0 : d_{uv}^{3\times3} = 0$ which is calibrated using the asymptotic normal distribution of $\hat{d}_{ur}^{3\times3}$ using the delta method. We apply a multiple testing correction to these $p(p-1)/2$ tests. To test if both d_{uv}^{2x2} and d_{vu}^{2x2} we take the maximum previously calculated p-value (after adjusting for multiple testing) of the two tests $H_0 : d_{vu}^{2x2} = 0$ and $H_0 : d_{uv}^{2x2} = 0$. We construct an undirected graph with an edge between each pair in the set $(u, v : d_{uv}^{3\times3} = 0, d_{uv}^{2x2} \neq 0, d_{vu}^{2x2} \neq 0)$. We then let $\mathcal{C}$ be the set of maximal cliques in that graph.
- In practice, the set
$$C^\star = (\{u, v\} : d_{uv}^{3\times3} = 0, d_{uv}^{2x2} \neq 0, d_{vu}^{2x2} \neq 0)$$
may be empty given a fixed hypothesis testing level. When this is the case, we ensure that $C^\star$ is non-empty with the following two steps. First, if the set $C^2 = (u, v : d_{uv}^{2x2} \neq 0, d_{vu}^{2x2} \neq 0)$ is empty for our initial hypothesis testing level, we set it to
$$C^2 = (u, v) = \arg\min_{v',v''} \max(\pi^2_{v',v''}, \pi^2_{v'',v'})$$
where $\pi^2_{v',v''}$ is the p-value when testing if $d^{2\times2}_{v',v''} = 0$. Once C^2 is non-empty, we then set
$$C^3 = (\{u, v\} : d_{uv}^{3\times3} = 0).$$
If $C^2 \cap C^3 = \emptyset$, we then lower the hypothesis testing level until it is non empty; i.e.,
$$C^3 = (u, v) = \arg\max_{(v',v'')\in C^2} \pi^3_{v',v''}$$
where $\pi^3_{v'',v'}$ is the p-value for testing if $d^{3\times3}_{v',v''} = 0$. This ensures that $C^\star = C^2 \cap C^3$ is non-empty.
- Line 5: To prune $\mathcal{C}$ to obtain $\mathcal{C}'$ all root cycles we do the following. For each maximal clique $C_1 \in \mathcal{C}$ and let $C_2 = (v : v \in C$ for some $C \in \mathcal{C}) \setminus C_1$. Letting $X_{C_2.C_1}$ be the residuals when regressing X_{C_2} onto X_{C_1}, we then test if $E(X_{C_2.C_1} X^2_{C_1})$ is the zero matrix using empirical likelihood. We then apply a multiple testing procedure to each of the $|\mathcal{C}|$ tests. In practice, all $C \in \mathcal{C}$ may be rejected, in which case we simply set the root cycle to be the union of all maximal cliques. Alternatively, we could pick the maximal clique with the largest p-value, but that ends up doing worse empirically. If, multiple maximal cliques in $C \in \mathcal{C}$ are not rejected which happen to have non-empty intersection, we combine them into a larger single cycle.
- The output of Part 1 is a topological layer where each layer is either a set of singleton nodes or a set of cycles. For each layer with cycles, we construct an undirected cycle for the set D using the greedy approach described in Algorithm 2.

Algorithm 2 Find Cycle

Require: $\hat{S}$: the sample correlation of nodes in a p-cycle
1: Start with an empty undirected graph $G = \{V = [p], E = \emptyset\}$
2: **for** $k = 1 \ldots p-1$ **do**
3: $\quad E_{\text{valid}} = \{(u, v) : u$ and v are not connected in G and have degree less than 2$\}$
4: $\quad$ Add the edge $\arg\max_{(u,v)\in E_{valid}} |(\hat{S})^{-1}_{u,v}|$ to E
5: **end for**
6: Add the edge (u, v) to G where u and v are the only remaining nodes with degree 1
7: Estimate the path weights when orienting the cycle in both directions
8: Select the orientation with the smaller absolute value of the product of edgeweights

- Once we have established the undirected cycle, we have two sets of edgeweights for each edge in the cycle (via Theorem 4.13) corresponding to the two possible orientations of the cycle. We estimate both directions, then pick the cycle orientation which has the smaller product of the edge weights.
- Line 8: Once we have oriented the cycle and estimated the edgeweights of the cycle $\hat{\Lambda}_{D,D}$, we select parents for each node in the cycle through the following procedure:
  - Let C denote all nodes which are in a layer preceeding D. We estimate the edgeweights from $C \to D$ using

    $$\hat{\Lambda}_{C,D} = \hat{R}_{CD}(I - \hat{\Lambda}_{D,D})$$

    where $\hat{R}_{CD}$ is the matrix of least squares coefficients when regressing D onto C.
  - Then for each $d \in D$ and $c \in C$, we test the existence of edge $c \to d$ by the following. Let $d' \in D$ be the parent of d in the cycle. Then, we estimate the residuals of Y_d when setting the coefficient $\lambda_{cd} = 0$ by the following:

    $$\hat{\epsilon}_{d,-c} = Y_d - \sum_{c' \neq c} \hat{\lambda}_{cd} Y_{c'} - \hat{\lambda}_{d',d} Y_{d'}.$$

    We see empirically that failing to account for the estimation of the nuissance parameters $\hat{\lambda}_{C \setminus c,d}$ and $\lambda_{d',d}$ and naively testing $E(\hat{\epsilon}_{d,-c} Y_c^2) = 0$ using empirical likelihood does not maintain nominal size. Thus, we instead take the following $|C| + 1$ estimating equations.

    $$\begin{aligned} m_1(X) &= \hat{\epsilon}_{d,-c} Y_c^2 \\ m_k(X) &= \hat{\epsilon}_{d,-c} Y_{c'}^2 \text{ for } c' \in C \text{ and } 1 < k \leq |C| \\ m_{|C|+1}(X) &= \hat{\epsilon}_{d,-c}^2 Y_c \end{aligned} \tag{18}$$

    Letting $A \in \mathbb{R}^{|C|+1 \times |C|+a}$ Jacobian of the estimating equations with respect the coefficients. Without loss of generality, we will assume that the potential parent of interest c is first in C. Then, we have

    $$g(X) = m_1(X) - A_{1,-1} A_{-1,-1}^{-1} m_{2:(|C|+1)}(X).$$

    Using empirical likelihood to test if $g(X) = 0$ for each $(c, d) \in V \times V$ such that c is in a topological layer preceding d yields a p-value $\pi_{c,d}$. To estimate the final edge set, we add edge $c \to d$ if $\pi_{c,d}$ after being adjusted for multiple testing exceeds a threshold.

A.8.2 Limitations and Future Directions

It is valuable to consider the behavior of the algorithm when its underlying assumptions are violated. This issue is not unique to our method; for instance, DAG-based algorithms like DirectLiNGAM [Shimizu et al., 2011] also face challenges when their acyclicity assumption is violated by the presence of cycles or confounding.

In our case, a key limitation arises when the graph structure contains overlapping cycles. In such cases, our method can correctly infer the causal structure up to the point where it encounters a root component that is a strongly connected component but is not a simple cycle (i.e., it contains two or more intersecting cycles). At this juncture, two strategies are possible. The first is a conservative approach where the algorithm reports that no further simple cycles can be identified and terminates, providing no additional details about the subgraph induced by the remaining nodes. The second is a more aggressive strategy that attempts to find a "best-fitting simple cycle model" by, for example, identifying a cycle that minimizes the violation of the required polynomial equalities.

Our current implementation follows the conservative strategy. When no simple root cycle is found, our algebraic conditions fail the statistical tests, and the algorithm correctly indicates the absence of a simple cycle before halting. While the more aggressive strategy is common in implementations of related algorithms like DirectLiNGAM, we have not yet implemented it for our method. Developing and evaluating such a strategy presents an interesting and potentially fruitful direction for future work.

B ADDITIONAL NUMERICAL EXPERIMENTS

In this section we first present additional numerical experiments. Specifically, we generate data in the same way as described in Section 5. However, we now create graphs with $p/5$ disjoint cycles where each cycle is of size 5. In general, the procedure

performs better when the graphs have 3-cycles when compared to the setting with 5-cycles. Similar to the setting where the graphs consisted of 3-cycles, the procedure performs better when the errors are a mixture of normals when compared to the gamma errors.

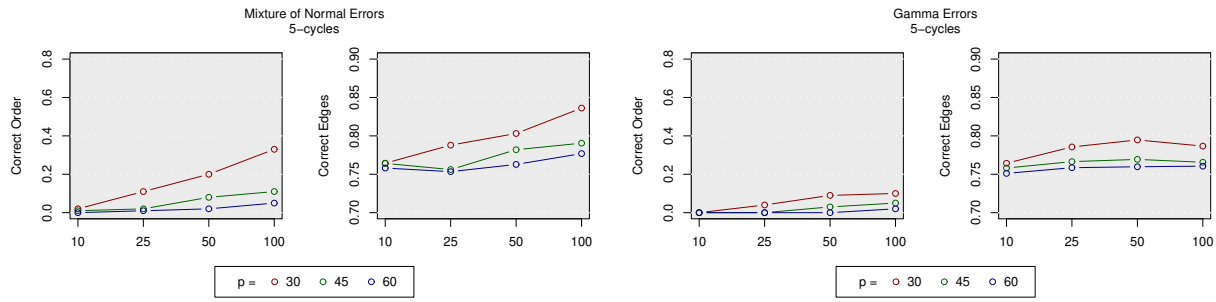

Figure 6: Results when each graph consists of 5-cycles.

In Table 1, we give details for the ICA based procedure of Lacerda et al. [2008]. Specifically, we generate random graphs with 3-cycles and data as described in Section 5. We use the implementation from the rpy-tetrad [Ramsey and Andrews, 2023] and run the experiments on a cluster with 24GB of memory allocated to each job. Table 1 shows the number of replicates (out of 25 for each setting) where the ICA based procedure returns an estimate within 24 hours. In general, we see that the computational performance improves as n grows larger and that performance is better for the mixture of normals when compared to the gamma errors. This is because when the demixing matrix is estimated more precisely fewer permutations must be considered in the post-processing procedures. We note that in each instance where an estimated graph is returned, the true graph is recovered exactly.

Table 1: The number of replicates where the ICA approach returns an estimated graph within 24 hours.

		Completed replicates out of 25	
p	n (in 000s)	Mixture of normals	Gamma
18	10	16	1
	25	21	4
	500	24	9
	100	24	14
21	10	4	0
	25	9	0
	50	16	1
	100	22	10
24	10	4	0
	25	9	0
	50	1	0
	100	1	0
27	10	0	0
	25	0	0
	50	0	0
	100	0	0