LEARNING GENE REGULATORY NETWORKS UNDER FEW ROOT CAUSES ASSUMPTION.

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a novel directed acyclic graph (DAG) learning method based on a causal form of Fourier-sparsity. Our ideas connect a theory of causal Fourier analysis with data generated by a structural equation model (SEM). We show that data generated by linear SEMs can be characterized in the Fourier domain as having a dense spectrum of root causes with random coefficients. We then propose the new problem of learning DAGs from data with sparse spectra (Fourier-sparsity) or, equivalently, few root causes. We provide proofs of identifiability in the new setting and, moreover, show that the true DAG is the global minimizer of the L^0-norm of the approximated spectra. Our method is applied to the CausalBench Challenge showing superior performance over the baselines.

1 INTRODUCTION

For causality, Fourier-sparsity translates into the data being generated from a few spectral coefficients or (in a sense being defined) few root causes. To formally analyze this we express the linear SEM equation based on the recently proposed theory of causal Fourier analysis on DAGs (Seifert et al., 2022b;a). We first show that data produced by linear structural equation models (SEMs) in prior work can be viewed as having a dense, random Fourier spectrum of causes and then extend them to include case where the data is sparse in the Fourier domain, i.e., has few root causes. It is worth noting that for prior, classical forms of Fourier transforms, Fourier-sparsity has played a significant role, including for the discrete Fourier transform (DFT) (Hassanieh, 2018), the discrete cosine transform (DCT) where it enables JPEG compression (Wallace, 1991), and the Walsh-Hadamard transform (WHT) for estimating set functions Stobbe & Krause (2012); Amrollahi et al. (2019).

Contributions. Towards this competition we provide the following contributions.

- We analyze linear SEMs in the Fourier domain and show that they yield dense spectra. We then pose the new assumption of data generated from a few root causes.
- For DAG learning from data with few root causes assumption, we propose a novel, linear DAG learning method, called MÖBIUS , based on the minimization of the L^1-norm of the approximated spectrum. We provide theoretical guarantees for our method.
- We evaluate our method on the CausalBench dataset and show that MÖBIUS offers improvement over prior DAG learning methods.

2 MOTIVATION

Consider a DAG $\mathcal{G} = (V, E)$ with $|V| = d$ vertices, E the set of directed edges, and no self-loops. We say j is a parent of i whenever $(j, i) \in E$ while j is an ancestor of i means there is a path from j to i. The vertices are sorted topologically and we set accordingly $V = \{1, 2, ..., d\}$. Further, we assume a weighted adjacency matrix $\mathbf{A} = (a_{ij})_{i,j \in V}$ of the graph, where $a_{ij} = 0$ if there is no edge.

Linear SEM. A data matrix $\mathbf{X} \in \mathbf{R}^{n \times d}$ consisting of n signals (as rows) of dimension d indexed by the DAG $\mathcal{G}$ satisfies a linear SEM (Shimizu et al., 2006; Zheng et al., 2018; Ng et al., 2020) if

$$\mathbf{X} = \mathbf{X}\mathbf{A} + \mathbf{N}. \tag{1}$$

where the matrix $\mathbf{N}$ consists of independent random noise samples. Isolating $\mathbf{X}$ in (1) yields:

$$\mathbf{X} = \mathbf{N}\left(\mathbf{I} - \mathbf{A}\right)^{-1} = \mathbf{N}\left(\mathbf{I} + \mathbf{A} + ... + \mathbf{A}^{(d-1)}\right) = \mathbf{N}\left(\mathbf{I} + \overline{\mathbf{A}}\right) \tag{2}$$

where $\overline{\mathbf{A}} = \mathbf{A} + \mathbf{A}^2 + ... + \mathbf{A}^{d-1}$ denotes the polynomial FW weighted transitive closure of the adjacency matrix. Eq. (2) can be viewed as the close form solution of Eq. (1).

Causal Fourier transform. Eq. (2) connects with the recent causal Fourier analysis framework of Seifert et al. (2022b). Consider the linear equation between the observed signal $\mathbf{s}$ and its causes $\mathbf{c}$

$$\mathbf{s} = \left(\mathbf{I} + \overline{\mathbf{A}}^T\right)\mathbf{c}. \tag{3}$$

Seifert et al. (2022b) argue that $\mathbf{c}$ can be interpreted as a form of spectrum of $\mathbf{s}$. This is done by providing a suitable notion of shift and associated shift-equivariant convolution whose eigenvectors are the columns of $\mathbf{I}+\overline{\mathbf{A}}^T$, following the algebraic theory of constructing Fourier analyses by Püschel & Moura (2006; 2008).

Fourier-sparse data. Combining Eq. (2) and (3) naturally leads to the idea of assuming a more general linear SEM model, which in addition to the random noise term contains an explicit term corresponding to the root causes. The equation of generating data $\mathbf{X} \in \mathbb{R}^{n\times d}$ becomes

$$\mathbf{X} = \left(\mathbf{C} + \mathbf{N}_f\right)\left(\mathbf{I} + \overline{\mathbf{A}}\right) + \mathbf{N}_s. \tag{4}$$

The matrix $\mathbf{C} \in \mathbb{R}^{n\times d}$ represents the root causes and $\mathbf{N}_f, \mathbf{N}_s \in \mathbb{R}^{n\times d}$ the random noises for the frequency and signal domain, respectively. Approximate Fourier-sparsity or few root causes means that only a few coefficients in $\mathbf{C}$ are non-zero and the values of $\mathbf{N}_f, \mathbf{N}_s$ have negligible magnitude.

Example. Consider the first n Fibonacci numbers. The recurrence equation generating the sequence can be viewed as a linear SEM where each term depends on the two predecessors. Unrolling or solving this recurrence shows, equivalently, that all numbers only depend on the first 2. These two are the root causes and yield the setting of Eq. (4) with $\mathbf{C}$ sparse, having only the first two values nonzero, and $\mathbf{N}_f = \mathbf{N}_s = \mathbf{0}$. For general linear SEMs this recurrence-solving yields Eq. (2) from Eq. (1). Doing so yields our novel setting of Eq. (4) which captures the situation where some nodes in the DAG insert spikes of values that then percolate through the DAG as determined by the edge weights. But not exactly as captured by $\mathbf{N}_f$ and not exactly measurable as captured by $\mathbf{N}_s$.

3 Our Method

Theoretical Guarantees. First we show that the novel setting based on the assumption of generation via few root causes is identifiable and then we define a discrete optimization problem that is guaranteed to find the true DAG under the assumption of having enough data.

Theorem 3.1. *Assume data generated via the extended linear SEM Eq. (4). We assume that the spectra* $\mathbf{C}$ *are independent random variables taking uniform values from* $[0, 1]$ *with probability* p*, and are* $= 0$ *with probability* $1 - p$*. Then Eq. (4) translates into a linear SEM with non-Gaussian noise and thus* $\mathbf{A}$ *identifiable due to (Shimizu et al., 2006).*

Given the data $\mathbf{X}$ we propose the following optimization problem to retrieve the DAG structure:

$$\min_{\mathbf{A}\in\mathbb{R}^{d\times d}} \left\|\mathbf{X} - \mathbf{X}\mathbf{A}\right\|_0 \quad \text{s.t.} \quad \mathbf{A} \text{ is acyclic.} \tag{5}$$

Theorem 3.2. *Consider a DAG with weighted adjacency matrix* $\mathbf{A}$*. Suppose that* $\widehat{\mathbf{A}}$ *is the optimal solution of the optimization problem (5) where the number* n *of data rows in* $\mathbf{X}$ *satisfies*

$$n \geq \frac{2^{3d-2}d(d-1)}{(1-\delta)^2 p^k (1-p)^{d-k}} \tag{6}$$

where $k = \lfloor dp \rfloor$ *and*

$$\delta \;\geq \frac{1}{\sqrt{2}} \max\left\{ \frac{1}{p^k(1-p)^{d-k}\binom{d}{k}}\sqrt{\ln\left(\frac{1}{\epsilon}\right)},\; \binom{d}{k}\sqrt{\ln\left(\frac{\binom{d}{k}}{\epsilon}\right)} \right\}. \tag{7}$$

Then with probability $(1-\epsilon)^2$*,* $\mathbf{A}$ *is the global minimizer of (5), namely* $\widehat{\mathbf{A}} = \mathbf{A}$*.*

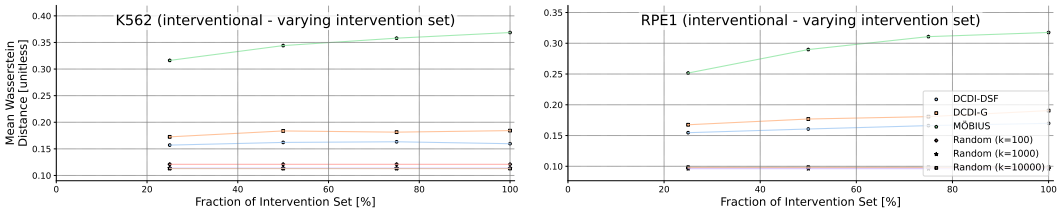


Figure 1: Evaluation of performance on the Datasets K562 and RPE1 (Replogle et al., 2022) based on CausalBench (Chevalley et al., 2022) framework. The Wasserstein distance metric (higher is better) is computed for our method in comparison to some implemented baselines.

MÖBIUS . Our method is formed as the continuous relaxation of the discrete optimization problem (5). We substitute the L^0-norm from (5) with its convex approximation (Ramirez et al., 2013), the L^1-norm. The acyclicity is then captured with the continuous constraint $h(\mathbf{A}) = tr\left(e^{\mathbf{A}\odot\mathbf{A}}\right) - d$ from (Zheng et al., 2018). Finally, we use $R(\mathbf{A}) = \lambda \|\mathbf{A}\|_1$ as the sparsity regularizer for the adjacency matrix and our final continuous optimization problem is formulated as

$$\min_{\mathbf{A}\in\mathbb{R}^{d\times d}} \frac{1}{2n} \|\mathbf{X} - \mathbf{X}\mathbf{A}\|_1 + \lambda\|\mathbf{A}\|_1 \quad \text{s.t.} \quad h(\mathbf{A}) = 0. \tag{8}$$

We call this method MÖBIUS due to the fact that the Fourier transform of (3) in the causal setting coincides with the weighted Möbius transform for posets (Seifert et al., 2022b).

Handling interventions. The gene expression data provided by the CausalBench framework can contain interventions, either for all genes or for a fraction of them. An intervention assigns a value to a gene which is independent to the expression data of its predecessors. Mathematically, the linear SEM adopting the intervention scheme is formulated with the following equation

$$\mathbf{X} = \mathbf{X}\mathbf{A}\mathbf{M} + \mathbf{N}. \tag{9}$$

$\mathbf{M}$ is an intervention mask which has rows with ones, except on the position i where it has 0 when the intervention acts on gene i. In that case the gene is initialized with noise according to Eq. (9) or more generally with some spectral value together with noise as captured by Eq. (4). Given that the positions of the interventions in the dataset are known the optimization problem becomes

$$\min_{\mathbf{A}\in\mathbb{R}^{d\times d}} \frac{1}{2n} \|\mathbf{X} - \mathbf{X}\mathbf{A}\mathbf{M}\|_1 + \lambda\|\mathbf{A}\|_1 \quad \text{s.t.} \quad h(\mathbf{A}) = 0. \tag{10}$$

4 Contest evaluation

Our method appears to work competitively in synthetic data generated with a few root causes and also in the gene regulatory network dataset by Sachs et al. (2005), as shown in the appendix. In Fig. 1 we present our performance on the gene-gene interaction network benchmark provided by Chevalley et al. (2022). Our method seems to perform better than the implemented baselines and also exhibits an upward trend which indicates that it benefits from interventions.

Implementational details. For our method, we construct a PyTorch model consisting of a linear layer which represents the weighted adjacency matrix $\mathbf{A}$. Then given the data $\mathbf{X}$, processed in batches, and the interventional positions masking matrix $\mathbf{M}$ we train our model with Adam optimizer with learning rate $\lambda = 10^{-3}$, to minimize the loss defined by Eq. (10). The final adjacency matrix is thresholded at 0.035 which experimentally showed to result into more than thousands of edges.

5 Conclusion

We presented a new perspective on linear SEMs motivated by a recently proposed causal Fourier analysis for DAGs. Mathematically, this perspective translates (or solves) the recurrence describing the SEM into an invertible linear transformation that takes as input a chosen Fourier spectrum of values, thus called root causes, to produce the data as output. Prior data generation for linear SEMs assumed a dense, random spectrum. In this paper we adapted the novel scenario of data generated from few root causes, to reconstruct the gene-gene interactome. Our assumption seems to performs well in this setting and possibly gives new insights for the data generation of gene expression data.

REFERENCES

Andisheh Amrollahi, Amir Zandieh, Michael Kapralov, and Andreas Krause. Efficiently learning Fourier sparse set functions. *Advances in Neural Information Processing Systems*, 32, 2019.

Mathieu Chevalley, Yusuf Roohani, Arash Mehrjou, Jure Leskovec, and Patrick Schwab. Causalbench: A large-scale benchmark for network inference from single-cell perturbation data. *arXiv preprint arXiv:2210.17283*, 2022.

Yinghua Gao, Li Shen, and Shu-Tao Xia. DAG-GAN: Causal structure learning with generative adversarial nets. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3320–3324. IEEE, 2021.

Haitham Hassanieh. *The Sparse Fourier Transform: Theory and Practice*, volume 19. Association for Computing Machinery and Morgan and Claypool, 2018.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural dag learning. *arXiv preprint arXiv:1906.02226*, 2019.

Ignavier Ng, AmirEmad Ghassami, and Kun Zhang. On the role of sparsity and dag constraints for learning linear dags. *Advances in Neural Information Processing Systems*, 33:17943–17954, 2020.

M. Püschel and J. M. F. Moura. Algebraic signal processing theory: Foundation and 1-D time. *IEEE Trans. Signal Process.*, 56(8):3572–3585, 2008.

Markus Püschel and José MF Moura. Algebraic signal processing theory. *arXiv preprint cs/0612077*, 2006.

Carlos Ramirez, Vladik Kreinovich, and Miguel Argaez. Why ℓ_1 is a good approximation to ℓ_0: A Geometric Explanation. *Journal of Uncertain Systems*, 7(3):203–207, 2013.

Alexander Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag! causal discovery benchmarks may be easy to game. *Advances in Neural Information Processing Systems*, 34:27772–27784, 2021.

Joseph M Replogle, Reuben A Saunders, Angela N Pogson, Jeffrey A Hussmann, Alexander Lenail, Alina Guna, Lauren Mascibroda, Eric J Wagner, Karen Adelman, Gila Lithwick-Yanai, et al. Mapping information-rich genotype-phenotype landscapes with genome-scale perturb-seq. *Cell*, 185(14):2559–2575, 2022.

Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

Bastian Seifert, Chris Wendler, and Markus Püschel. Causal Fourier Analysis on Directed Acyclic Graphs and Posets. *arXiv preprint arXiv:2209.07970*, 2022a.

Bastian Seifert, Chris Wendler, and Markus Püschel. Learning Fourier-Sparse Functions on DAGs. In *ICLR2022 Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022b.

Shohei Shimizu, Patrik O. Hoyer, Aapo Hyvärinen, and Antti Kerminen. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Journal of Machine Learning Research*, 7(72):2003–2030, 2006. URL `http://jmlr.org/papers/v7/shimizu06a.html`.

Peter Stobbe and Andreas Krause. Learning Fourier sparse set functions. In *Artificial Intelligence and Statistics*, pp. 1125–1133. PMLR, 2012.

Gregory K. Wallace. The JPEG Still Picture Compression Standard. *Commun. ACM*, 34(4):30–44, apr 1991. ISSN 0001-0782. doi: 10.1145/103085.103089. URL `https://doi.org/10.1145/103085.103089`.

Yue Yu, Jie Chen, Tian Gao, and Mo Yu. DAG-GNN: DAG structure learning with graph neural networks. In *International Conference on Machine Learning*, pp. 7154–7163. PMLR, 2019.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Table 1: SHD metric (lower is better) for learning DAGs with 40 nodes and 80 edges. Each row is an experiment. The first row is the default, whose settings are in the blue column. In each other row, exactly one default parameter is changed (current value). The last five columns correspond to prior algorithms. The best results are shown bold.

	Hyperparameter	Default value	Current value	Varsortability	MÖBIUS (ours)	GOLEM-EV	sortnregress	NOTEARS	DAG-NoCurl	GES
1.	Default settings			0.94	$\mathbf{0.00 \pm 0.00}$	3.10 ± 2.55	100.30 ± 22.55	3.10 ± 1.87	34.90 ± 7.26	73.30 ± 15.13
2.	Graph type	Erdös-Renyi	Scale-free	0.99	$\mathbf{0.00 \pm 0.00}$	2.30 ± 1.27	52.20 ± 18.24	2.30 ± 1.49	17.00 ± 11.53	61.70 ± 12.04
3.	Edges / Vertices	2	3	0.97	$\mathbf{0.00 \pm 0.00}$	6.30 ± 3.07	158.90 ± 30.91	15.10 ± 8.19	58.00 ± 10.83	162.50 ± 16.12
4.	Larger weights in $\mathbf{A}$	$(0.4, 0.8)$	$(0.5, 2)$	0.99	9.20 ± 14.88	$\mathbf{0.70 \pm 0.64}$	102.00 ± 34.06	11.00 ± 13.56	30.40 ± 11.21	107.90 ± 20.95
5.	High sparsity in $\mathbf{C}$	$p = 0.3$	$p = 0.1$	0.96	$\mathbf{0.00 \pm 0.00}$	11.30 ± 4.47	94.00 ± 17.52	15.40 ± 2.33	36.20 ± 7.82	73.50 ± 10.76
6.	Low sparsity in $\mathbf{C}$	$p = 0.3$	$p = 0.6$	0.95	60.20 ± 9.43	$\mathbf{2.40 \pm 1.43}$	96.60 ± 24.19	3.10 ± 2.43	41.00 ± 10.03	70.60 ± 14.94
7.	$\mathbf{N}_f, \mathbf{N}_s$ deviation	$\sigma = 0.01$	$\sigma = 0.1$	0.97	$\mathbf{0.10 \pm 0.30}$	4.10 ± 3.75	93.70 ± 22.07	3.70 ± 1.95	32.70 ± 7.66	76.10 ± 16.98
8.	$\mathbf{N}_f, \mathbf{N}_s$ distribution	Gaussian	Gumbel	0.96	$\mathbf{0.00 \pm 0.00}$	4.40 ± 2.01	86.00 ± 27.25	5.50 ± 2.50	40.00 ± 13.65	74.70 ± 18.88
9.	Signal noise	$\mathbf{N}_f \neq \mathbf{0}$, $\mathbf{N}_s = \mathbf{0}$	$\mathbf{N}_f = 0$, $\mathbf{N}_s \neq 0$	0.96	$\mathbf{0.00 \pm 0.00}$	4.00 ± 3.19	88.30 ± 19.79	4.90 ± 2.98	32.50 ± 6.31	72.80 ± 12.90
10.	Full Noise	$\mathbf{N}_f \neq \mathbf{0}$, $\mathbf{N}_s = \mathbf{0}$	$\mathbf{N}_f \neq 0$, $\mathbf{N}_s \neq 0$	0.94	$\mathbf{0.00 \pm 0.00}$	5.00 ± 3.03	103.60 ± 27.87	4.60 ± 2.83	29.70 ± 8.47	70.40 ± 17.45
11.	Standardization	No	Yes	0.50	84.90 ± 8.53	$\mathbf{68.50 \pm 5.66}$	243.90 ± 27.33	81.40 ± 4.80	102.30 ± 6.63	68.90 ± 14.14
12.	Samples	n=400	$n = 20$	0.90	252.80 ± 13.47	219.60 ± 32.63	error	$\mathbf{125.20 \pm 12.81}$	378.10 ± 28.11	264.60 ± 19.49
13.	Fixed support	No	Yes	0.80	106.00 ± 9.20	161.80 ± 43.09	208.90 ± 45.16	83.00 ± 3.26	287.90 ± 50.23	$\mathbf{74.90 \pm 12.33}$

A EXPERIMENTS

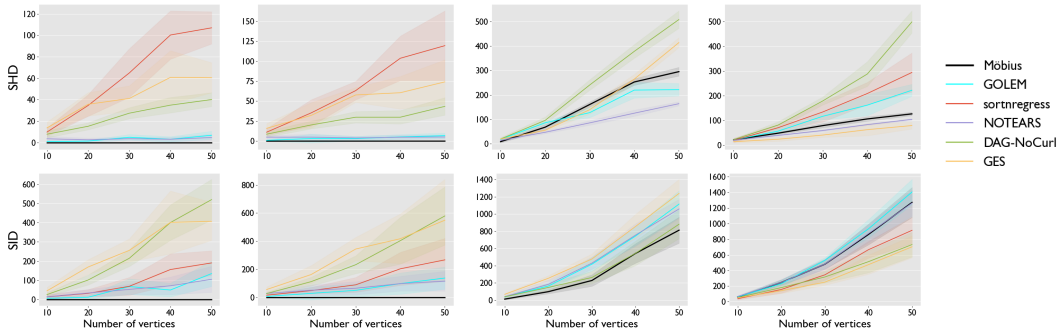
Data generating process and defaults. In the blue column of Table 1 we report the default settings for our experiment explained next. We generate a random Erdös-Renyi graph with $d = 40$ nodes and assign edge directions to make it a DAG as in (Zheng et al., 2018). The number of edges is set to 80, so the ratio of edges to vertices is 2. The entries of the weighted adjacency matrix are sampled uniformly at random from $(-b, -a) \cup (a, b)$, where $a = 0.4$ and $b = 0.8$. Following (Zheng et al., 2018; Ng et al., 2020) the result of the adjacency matrix approximation is post-processed for each algorithm by including only edges with absolute weight larger than a threshold $\omega = 0.3$. Next, the sparse DAG spectra $\mathbf{C}$ are instantiated by setting each entry either to some random uniform value from $(0, 1)$ with probability $p = 0.3$ or to 0 with probability $1 - p = 0.7$ (as in Theorem 3.1, thus, also the support will vary between spectra). The data matrix $\mathbf{X}$ is computed according to (4), using Gaussian noise $\mathbf{N}_f, \mathbf{N}_s$ of standard deviation $\sigma = 0.01$. By default we set $\mathbf{N}_s = \mathbf{0}$, to ensure identifiability based on Theorem 3.1, and consider $\mathbf{N}_s \neq 0$ in a variant next. The low standard deviation ensures that the data are approximately Fourier-sparse. Finally, we do not standardize (scaling for variance $= 1$) the data, and $\mathbf{X}$ contains $n = 400$ samples (rows).

Experiment 1: Different application scenarios. Table 1 compares MÖBIUS to five prior algorithms using the SHD metric. Every row corresponds to a different experiment that alters one particular hyperparameter of the default setting, which is the first row with values as explained above and in the blue column. For example, the second row only changes the graph type from Erdös-Renyi to scale-free, while keeping all other settings. Besides SHD, we also computed the SID and TPR metrics, shown in Appendix B. Note that in the last row, fixed support means that in every execution of the experiment to generate we first fix a random support for the spectra, and then we assign values to these entries uniformly from $[0, 1]$.

Since the graphs have 80 edges, an SHD $\ll 80$ can be considered as good and beyond 80 can be considered a failure. Overall, MÖBIUS achieves very low SHD compared to other methods and often even computes the exact solution, whereas none of the others does so.

First, we examine scenarios that alter the DAG configuration. For the default settings (row 1), for scale-free graphs (row 2), and for more edges (row 3) MÖBIUS performs best and perfectly detects all edges. As in (Zheng et al., 2018; Ng et al., 2020) we consider different weights for the adjacency matrix to examine the sensitivity of the weight scale (row 4). We see that, larger weight bounds affect our performance slightly and our method comes now second after GOLEM.

Next, we consider changing hyperparameters that affect the spectra and the data. Higher sparsity (row 5) keeps the performance perfect, whereas lower sparsity (row 6) degrades it significantly as expected. In contrast, imposing high sparsity is harmful for example for GOLEM and NOTEARS. Higher standard deviation in the spectral noise (row 7) decreases performance overall, but MÖBIUS still performs near optimal. Changing to Gumbel noise (row 8) keeps the perfect performance. Applying noise only in the spectrum or both in signal and spectrum (rows 9 and 10) mostly maintains performance over all, including our perfect reconstruction. The standardization of data (row 11) is generally known to negatively affect algorithms with continuous objectives (Reisach et al., 2021) as is the case here. For a small number of samples (row 12) or fixed sparsity support (row 13) all methods fail. Overall, MÖBIUS achieves the best SHD in most scenarios, often even recovering the true DAG.



(a) Default settings (b) $\mathbf{N}_s \neq \mathbf{0}, \mathbf{N}_f \neq \mathbf{0}$ (c) 20 samples (d) Fixed support

Figure 2: Plots illustrating performance metrics SID and SHD (lower is better) on the default settings, when both $\mathbf{N}_f, \mathbf{N}_s$ are non-zero, on 20 samples and on spectra with fixed support.

Table 2: Performance on the real data (Sachs et al., 2005).

	SHD ↓	SID ↓	NNZ
MÖBIUS	15	45	**16**
NOTEARS	**11**	44	15
GOLEM-EV	21	**43**	19
GOLEM-NV	12	47	10
sortnregress	18	**43**	20
DAG-NoCurl	22	44	23
LiNGAM	20	54	20
GES	13	57	8
fGES	17	62	14
MMHC	17	62	**16**
CAM	12	55	10

Experiment 2: Varying the number of nodes. In this experiment we benchmark MÖBIUS when varying the number of vertices (and thus the number of edges) in the ground truth DAG for four interesting scenarios of Table 1: default (first row), full noise (row 10), low samples (row 12) and fixed support (row 13). The results are shown in Fig. 2 for the two metrics SHD and SID.

In the default settings of Fig. 2a, we see that MÖBIUS achieves optimal performance for each number of nodes. The reason is that the optimization problem (8) is perfectly suited for this setting with exact Fourier-sparsity and sparse DAG. The same holds in Fig. 2b with added noise in the signal domain. The last two experiments in Figs. 2c and 2d are most challenging: allowing only few data or fixing the support, which loses the identifiability guarantees from Section **??**. For few samples MÖBIUS still achieves the best SID and SHD is close to the best by NOTEARS, but the numbers are still high to be considered a success. For fixed spectral support all algorithms practically fail to learn the DAG for any number of vertices.

Varsortability. In Table 1 we also include the varsortability for each different experimental setting. Our measurements for Erdös-Renyi graphs (all rows except row 2) are typically $2 - 5\%$ lower than the measurements reported in (Reisach et al., 2021, Appendix G.1) for linear SEMs, which shows that our experimental setting is not trivial for DAG learning. This can also be deduced by the performance of sortnregress, which overall fails. Note again that for fixed sparsity support (last row), all the methods fail and varsortability is very low. Therefore, in this scenario our data generating process poses a hard problem for DAG learning.

Real data. We also execute our method on the causal protein-signaling network provided by (Sachs et al., 2005). The dataset consists of 7466 [1,2] samples from a network with 11 nodes that represent proteins and 17 edges showing the interaction between them. Even though this network is relatively small, the task of learning it is considered difficult and has been a common benchmark for several

[1]As Lachapelle et al. (2019) we only use the first 853 samples.

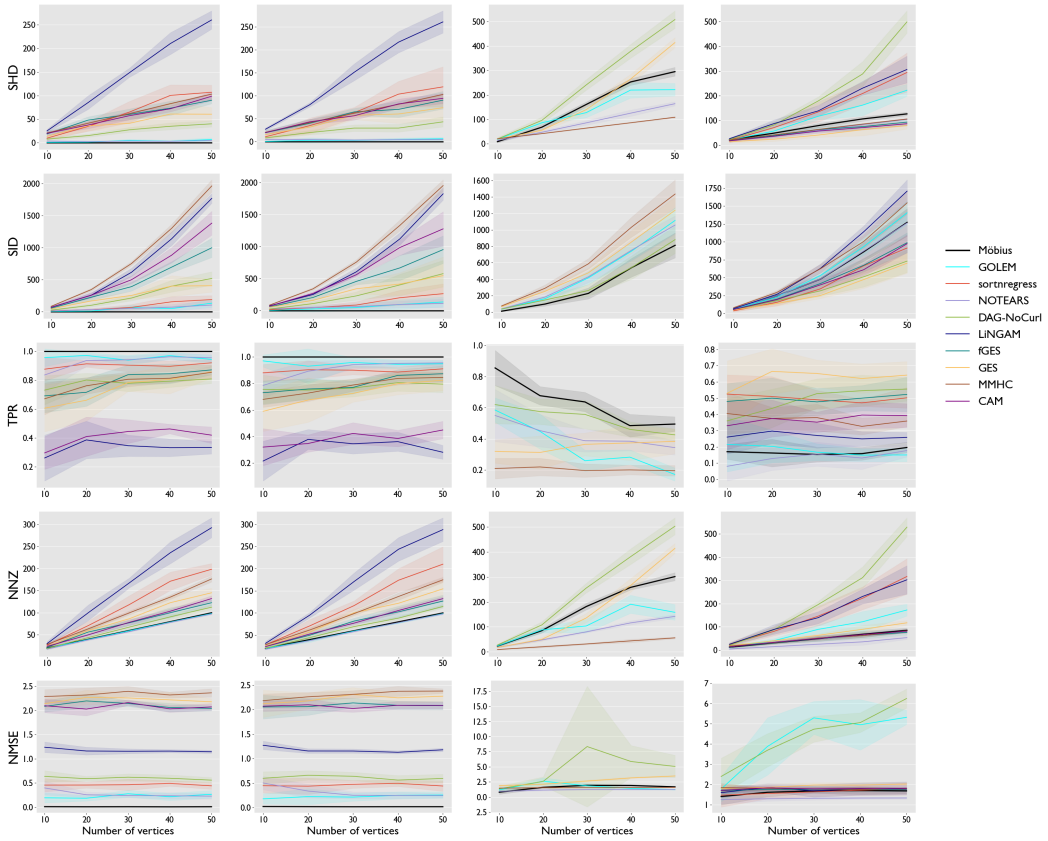
[2]Differences between our results and what is reported in the literature are due to different choice of hyperparameters and to the use of 853 samples, while others might utilize the full dataset.

prior DAG learning methods Ng et al. (2020); Gao et al. (2021); Yu et al. (2019); Zheng et al. (2018). We report the performance metrics for all methods in Table 2. It is not clear whether the assumption of Fourier-sparsity holds in this case.

The best SID is achieved by GOLEM and sortnregress, which, however, have higher SHD. NOTEARS has the best SHD equal to 11 , which is slightly better than ours which is 15. Overall, MÖBIUS performs reasonably well, achieving the closest number of edges to the real one 16 and a with SHD equal to 15 and SID equal to 45.

B ADDITIONAL EXPERIMENTAL RESULTS

In addition to Fig. 2 we include the plots of Fig. 3 for the 4 experiments of interest that vary the number of vertices of the ground truth DAG. The plots additionally contain the metrics TPR, NNZ and NMSE of the weighted approximation of the adjacency matrix with respect to the true one. The plots also include methods LiNGAM, fGES, MMHC and CAM.



(a) Default settings (b) $\mathbf{N}_s \neq \mathbf{0}, \mathbf{N}_f \neq \mathbf{0}$ (c) 20 samples (d) Fixed support

Figure 3: Plots illustrating performance metrics SHD, SID (lower is better), TPR (higher is better) and NMSE (lower is better) on the default settings, when both noises are non-zero, on 20 samples and on spectra with fixed support.

C GLOBAL MINIMIZER

Theorem C.1. *Consider* $\mathbf{A}$ *be the weighted adjacency matrix of the true DAG and assume* $\widehat{\mathbf{A}}$ *is the optimal solution of the optimization problem*

$$\min_{\mathbf{A} \in \mathbb{R}^{d \times d}} \|\mathbf{X} - \mathbf{X}\mathbf{A}\|_0 \quad s.t. \quad h(\mathbf{A}) = 0 \tag{11}$$

We assume that the data $\mathbf{X}$ *are generated from spectra* $\mathbf{C}$*, where each of their entry is non-zero independently with propability* p*, via the equation*

$$\mathbf{X} = \mathbf{C}\left(\mathbf{I} + \overline{\mathbf{A}}\right) \tag{12}$$

Assume that the number of data n *satisfies*

$$n \geq \frac{2^{3d-2} d(d-1)}{(1-\delta)^2 p^k (1-p)^{d-k}} \tag{13}$$

where $k = \lfloor dp \rfloor$ *and*

$$\delta \geq \max\left\{ \frac{1}{p^k (1-p)^{d-k} \binom{d}{k}} \sqrt{\frac{1}{2} \ln\left(\frac{1}{\epsilon}\right)}, \binom{d}{k} \sqrt{\frac{1}{2} \ln\left(\frac{\binom{d}{k}}{\epsilon}\right)} \right\}. \tag{14}$$

Then with probability $(1-\epsilon)^2$ *the solution to the optimization problem coincides with the true DAG, namely* $\widehat{\mathbf{A}} = \mathbf{A}$.

Remark C.2. The reason we choose $k = \lfloor dp \rfloor$ is that because of the Bernoulli trials for the non-zero spectral values, the expected value of the cardinality of the support will be exactly $\lfloor dp \rfloor$. Thus we expect to have more concentration on that value of k which is something we desire, since in the proof we need all possible patterns with support equal to k.

We begin the proof with some important observations and definitions.

Lemma C.3. *If* $\overline{\widehat{\mathbf{A}}} = \overline{\mathbf{A}}$ *then* $\widehat{\mathbf{A}} = \mathbf{A}$.

Proof. We have that

$$\mathbf{I} + \overline{\widehat{\mathbf{A}}} = \mathbf{I} + \overline{\mathbf{A}} \Leftrightarrow \left(\mathbf{I} + \overline{\widehat{\mathbf{A}}}\right)^{-1} = \left(\mathbf{I} + \overline{\mathbf{A}}\right)^{-1} \Leftrightarrow \left(\mathbf{I} - \widehat{\mathbf{A}}\right) = (\mathbf{I} - \mathbf{A}) \Leftrightarrow \widehat{\mathbf{A}} = \mathbf{A} \tag{15}$$

□

Definition C.4. We define $\widehat{\mathbf{C}} = \mathbf{X} - \mathbf{X}\widehat{\mathbf{A}}$ the spectra corresponding to the optimal adjacency matrix.

Definition C.5. Let $S \subset \{1, 2, 3, ..., d\}$ be a set of indices. We say that a spectrum $\mathbf{c}$ has support S if $c_i = 0$ for $i \in [d] \setminus S$.

Definition C.6. For a given support S we consider the set $R \subset [n]$ of the rows of $\mathbf{C}$ that have support S. Then, $\mathbf{C}_R, \widehat{\mathbf{C}}_R$ denote the submatrices consisting of the rows with indices in R.

Lemma C.7. *For any row subset* $R \subset [n]$ *we have that rank* $\left(\widehat{\mathbf{C}}_R\right) =$ *rank* $(\mathbf{C}_R)$

Proof. We have that

$$\widehat{\mathbf{C}}\left(\mathbf{I} + \overline{\widehat{\mathbf{A}}}\right) = \mathbf{X} = \mathbf{C}\left(\mathbf{I} + \overline{\mathbf{A}}\right) \tag{16}$$

Therefore, since both $\mathbf{A}, \widehat{\mathbf{A}}$ are acyclic and $\mathbf{I} + \overline{\widehat{\mathbf{A}}}, \mathbf{I} + \overline{\mathbf{A}}$ are invertible we have that

$$\widehat{\mathbf{C}}_R\left(\mathbf{I} + \overline{\widehat{\mathbf{A}}}\right) = \mathbf{C}_R\left(\mathbf{I} + \overline{\mathbf{A}}\right) \Leftrightarrow \text{rank}\left(\widehat{\mathbf{C}}_R\right) = \text{rank}\left(\mathbf{C}_R\right) \tag{17}$$

□

Lemma C.8. *For any row subset* $R \subset [n]$ *of spectra* $\mathbf{C}$ *with the same support* S *such that* $|S| = k$ *and* $|R| = r \geq k$ *the nonzero columns of* $\mathbf{C}_R$ *are linearly independent with probability* 1 *and therefore rank* $(\mathbf{C}_R) = k$.

Proof. assume $\mathbf{c}_1, ..., \mathbf{c}_k$ are the non-zero columns of $\mathbf{C}_R$. Then each $\mathbf{c}_i$ is a vector of dimension r whose each entry is sampled uniformly at random in the range $[0.5, 1]$. Given any $k-1$ vectors from $\mathbf{c}_1, ..., \mathbf{c}_k$ their linear span can at most make a subspace of $[0.5, 1]^r$ of dimension at most $k-1$. However, since every $\mathbf{c}_i$ is sampled uniformly at random from $[0.5, 1]^r$ and $r \geq k > k-1$ the probability that $\mathbf{c}_i$ lies in the linear span of the other $k-1$ vectors has measure 0. Therefore, the required result holds with probability 1. □

Lemma C.9. *With probability $(1-\varepsilon)^2$ for every different support S with $|S| = k$ there are rows R such that $\mathbf{C}_R$ have support S and*

$$|R| > 2^{3d-2}d(d-1) \tag{18}$$

Proof. Set $l = 2^{3d-2}d(d-1)$ and $K = \frac{l\binom{d}{k}}{(1-\delta)}$. Also set N be the random variable representing the number of spectra $\mathbf{c}$ with $|supp(\mathbf{c})| = k$ and N_i the number of repetitions of the $i-$th $k-$support pattern, $i = 1, ..., \binom{d}{k}$.

We first use conditional probability.

$$\mathbb{P}\left(N \geq K \cap N_i \geq l,\, \forall i\right) = \mathbb{P}\left(N_i \geq l,\, \forall i | N \geq k\right)\mathbb{P}\left(N \geq K\right) \tag{19}$$

Now we will show that $\mathbb{P}\left(N \geq K\right) \geq (1-\epsilon)$ and $\mathbb{P}\left(N_i \geq l,\, \forall i | N \geq k\right) \geq (1-\epsilon)$ using Chernoff bounds. From Hoeffding inequality we get:

$$\mathbb{P}\left(N \leq (1-\delta)\mu\right) \leq e^{-2\delta^2\mu^2/n^2} \tag{20}$$

where $\mu = np^k(1-p)^{d-k}\binom{d}{k}$ is the expected value of N, $\delta \geq \frac{1}{p^k(1-p)^{d-k}\binom{d}{k}}\sqrt{\frac{1}{2}\ln\left(\frac{1}{\epsilon}\right)}$ and $n \geq \frac{1}{1-\delta}\frac{K}{p^k(1-p)^{d-k}\binom{d}{k}}$ so

$$\mathbb{P}\left(N \leq K\right) \leq \mathbb{P}\left(N \leq (1-\delta)\mu\right) \leq e^{-2\delta^2\mu^2/n^2} \leq \varepsilon \Leftrightarrow \mathbb{P}\left(N \geq K\right) \geq 1-\epsilon \tag{21}$$

Now given that we have at least K spectra with support exactly k, we will show that, on exactly K such spectra, each different pattern will appear at least l times with probability $(1-\epsilon)$. From the union bound

$$\mathbb{P}\left(\bigcup_i N_i \leq l\right) \leq \sum_i \mathbb{P}\left(N_i \leq l\right) \tag{22}$$

So we need to show that $\mathbb{P}\left(N_i \leq l\right) \leq \frac{\epsilon}{\binom{d}{k}}$. We use again the Hoeffding inequality.

$$\mathbb{P}\left(N_i \leq (1-\delta)\mu\right) \leq e^{-2\delta^2\mu^2/K^2} \tag{23}$$

where the expected value is $\mu = \frac{K}{\binom{d}{k}} = \frac{l}{1-\delta}$ since now the probability is uniform over all possible $k-$sparsity patterns. Given $\delta \geq \binom{d}{k}\sqrt{\frac{1}{2}\ln\left(\frac{\binom{d}{k}}{\epsilon}\right)}$ we derive

$$\mathbb{P}\left(N_i \leq l\right) = \mathbb{P}\left(N_i \leq (1-\delta)\mu\right) \leq e^{-2\delta^2\mu^2/K^2} \leq \frac{\epsilon}{\binom{d}{k}} \tag{24}$$

The result follows. □

Lemma C.10. *Assume that the data number n is large enough so that for every different support S we have the rows R such that $\mathbf{C}_R$ have support S and $|R| > 2^d l$ where*

$$l > 2^{2d-2}d(d-1) = 2^d\sum_{k=2}^{d}\binom{d}{k}k(k-1) \tag{25}$$

Then there exist rows $\hat{R}$ such that $|\hat{R}| > k$, $\mathbf{C}_{\hat{R}}$ has support S and $\widehat{\mathbf{C}}_{\hat{R}}$ has support $\hat{S}$ with $|\hat{S}| = k$. Moreover, both the non-zero columns of $\mathbf{C}_{\hat{R}}, \widehat{\mathbf{C}}_{\hat{R}}$ form two linearly independent set of vectors. In words, this means that the data are many enough so we can always find pairs of corresponding spectral submatrices with each consisting of k non-zero columns.

Proof. According to the assumption there exists a row-submatrix $\mathbf{C}_R$ where all rows have support R and $|R| > 2^d l$. Group the spectra $\widehat{\mathbf{C}}_R$ into all different supports, which are 2^d in number. Take any such group $R' \subset R$ of rows of $\widehat{\mathbf{C}}$ that all have the same support S'. If $|R'| \geq k$ then from Lemma C.7 rank $\left(\widehat{\mathbf{C}}_{R'}\right) = \text{rank}\left(\mathbf{C}_{R'}\right) = k$ so $\widehat{\mathbf{C}}_{R'}$ will have at least k non-zero columns, and since the support is fixed, it will have at least k non-zero elements at each row, which means at least as many non-zeros

as $\mathbf{C}_{R'}$. Therefore $\widehat{\mathbf{C}}_{R'}$ can only have less non-zero elements if $|R'| < k$, and in that case $\mathbf{C}_{R'}$ has at most $k(k-1)$ more elements. If we count for all $k = 1, ..., d$ all different supports of $\widehat{\mathbf{C}}_{R'}$ for all possible supports of $\mathbf{C}_R$ this gives that $\widehat{\mathbf{C}}$ can have at most $\sum_{k=2}^{d} \binom{d}{k} 2^d k(k-1)$ less non-zero elements compared to $\mathbf{C}$.

Due to the pigeonhole principle, there exists $\hat{R} \subset R$ and $|\hat{R}| > l$ with $\widehat{\mathbf{C}}_{\hat{R}}$ all having the same support $\hat{S}$, not necessarily equal to S. According to our previous explanation we need to have at least k non-zero columns in $\widehat{\mathbf{C}}_{\hat{R}}$. If we had $k+1$ columns then this would give l more non-zero elements, but

$$l > 2^{2d-2} d(d-1) = 2^d d(d-1) 2^{d-2} = 2^d \sum_{k=0}^{d-2} \binom{d-2}{k-2} d(d-1) = 2^d \sum_{k=2}^{d} \binom{d}{k} k(k-1) \quad (26)$$

So then $\|\widehat{\mathbf{C}}\|_0 > \|\mathbf{C}\|$ which is a contradiction due to the optimality of $\widehat{\mathbf{A}}$. Therefore, $\widehat{\mathbf{C}}_{\hat{R}}$ has exactly k non-zero columns which necessarily need to be linearly independent in order to have rank $\left(\widehat{\mathbf{C}}_{\hat{R}}\right) = \text{rank}\left(\mathbf{C}_{\hat{R}}\right) = k$ as necessary. The linear independence of the columns of $\mathbf{C}_{\hat{R}}$ follows from Lemma C.8 since $l \gg k$. □

Definition C.11. A pair $\mathbf{C}_R, \widehat{\mathbf{C}}_R$ constructed according to Lemma C.10 that have fixed support $S, \hat{S}$ respectively with cardinality k each and $|R|$ large enough so that rank $\left(\widehat{\mathbf{C}}_{\hat{R}}\right) = \text{rank}\left(\mathbf{C}_{\hat{R}}\right) = k$, will be called a $k-$*pair* of submatrices.

Remark C.12. For notational simplicity we will drop the index R whenever the choice of the rows according to the sparsity pattern S is defined in the context.

The complete Theorem 3.2 follows after combining the following two propositions.

Proposition C.13. *If the data* $\mathbf{X}$ *are indexed such that* $\mathbf{A}$ *is upper triangular, then so is* $\widehat{\mathbf{A}}$.

Remark C.14. Notice that $\mathbf{A}$ is upper triangular if and only if $\overline{\mathbf{A}}$ is upper triangular. This holds as the polynomial $1 + x + x^2 + ... + x^{d-2}$ doesn't have real roots. Thus we only need to show that $\overline{\hat{a}}_{ji} = 0$ for all $i < j$.

Before we proceed to the proof of Prop. C.13 we first prove the following helpful lemma.

Lemma C.15. *Consider* $\mathbf{C}, \widehat{\mathbf{C}}$ *a* $k-$*pair. Also let* $\mathbf{X}$ *be the corresponding submatrix of data. If* $\mathbf{X}_{:,i} = \mathbf{0}$ *then*

$$\widehat{\mathbf{C}}_{:,i} = \mathbf{0} \text{ and } \overline{\hat{a}}_{ji} = 0 \quad \forall j \in \text{supp}\left(\widehat{\mathbf{C}}\right) \quad (27)$$

Proof.

$$\mathbf{0} = \mathbf{X}_{:,i} = \sum_{j=1}^{d} \overline{\hat{a}}_{ji} \widehat{\mathbf{C}}_{:,j} + \mathbf{C}_{:,i} = \sum_{j \in \text{supp}\left(\widehat{\mathbf{C}}\right)} \overline{\hat{a}}_{ji} \widehat{\mathbf{C}}_{:,j} + \mathbf{C}_{:,i} \quad (28)$$

If $\mathbf{C}_{:,i} \neq \mathbf{0}$ then $i \in \text{supp}\left(\widehat{\mathbf{C}}\right)$, and the expression above constitutes a linear combination of the support columns with not all coefficients non-zero. This contradicts Lemma C.8. Thus $\mathbf{C}_{:,i} \neq \mathbf{0}$ and $\overline{\hat{a}}_{ji} = 0 \quad \forall j \in \text{supp}\left(\widehat{\mathbf{C}}\right)$. □

We are now ready to prove Prop. C.13.

Proof. We first choose a $k-$pair $\mathbf{C}, \widehat{\mathbf{C}}$ such that the support S of $\mathbf{C}$ is concetrated to the last k columns, namely $\mathbf{C}_{:,i} = 0$ for $i = 1, ..., d-k$. Then the corresponding data submatrix $\mathbf{X}$ will necessarily have the same sparsity pattern since the values are computed according to predecessors, which in our case lie in smaller indices. Thus $\mathbf{X}_{:,i} = \mathbf{0}. \quad \forall i = 1, ..., d-k$ and according to Lemma C.15 we get that

$$\overline{\hat{a}}_{ji} = 0, \text{ for all } 1 \leq i \leq d-k \text{ and } d-k+1 \leq j \leq d. \quad (29)$$

We notice that the desired condition is fullfilled for $i = d-k$, i.e. it has no non-zero influence from a node with larger index. We now prove the same sequentially for $i = d-k-1, d-k-2, ..., 1$

by moving each time left the leftmost index of the support S. We prove the following statement by induction:

$$P(l) : \overline{\hat{a}}_{ji} = 0 \text{ for } l < j,\ i < j\ i \leq d-k \tag{30}$$

We know that $P(d-k)$ is true and $P(1)$ gives the required relation for all indices $i = 1, ..., d-k$. Now we assume that $P(l)$ holds. If we pick $k-$pair $\mathbf{C}, \widehat{\mathbf{C}}$ such that $\mathbf{C}$ has support $S = \{l, d-k+2, d-k+3..., d\}$. Then $\mathbf{X}_{:,i} = \mathbf{0}$ for all $i < l$ which with Lemma C.15 gives $\mathbf{C}_{:,i} = \mathbf{0}$ for $i < l$ which means supp $\left(\widehat{\mathbf{C}}\right) \subset \{l, l+1, ..., d\}$ and $\overline{\hat{a}}_{ji} = 0$ for $j \in$ supp $\left(\widehat{\mathbf{C}}\right)$. Note, that by the induction hypothesis we have that $\overline{\hat{a}}_{jl} = 0$ for all $l < j$. However it is true that $\mathbf{X}_{:,l} = \mathbf{C}_{:,l}$ and also

$$\mathbf{X}_{:,l} = \sum_{j \in \text{supp}(\widehat{\mathbf{C}})}^{d} \overline{\hat{a}}_{jl} \widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,l} = \widehat{\mathbf{C}}_{:,l} \tag{31}$$

Therefore $l \in$ supp $\left(\widehat{\mathbf{C}}\right)$ and thus $\overline{\hat{a}}_{li} = 0$ for all $i < l$ which combined with $P(l)$ gives

$$\overline{\hat{a}}_{ji} = 0 \text{ for } l-1 < j,\ i < j,\ i \leq d-k \tag{32}$$

which is exactly $P(l-1)$ and the induction is complete.

Now it remains to show that $\overline{\hat{a}}_{ji} = 0$ for $d-k+1 \leq i \leq d$ and $i < j$. We will again proceed constructively using induction. This time we will sequentially choose support S that is concentrated on the last $k+1$ columns and at each step we will move the zero column one index to the right. For $l = d-k+1, ..., d$ let's define:

$$Q(l) : \overline{\hat{a}}_{jl} = 0 \text{ for } l < j \text{ and } \overline{\hat{a}}_{jl} = \overline{a}_{jl} \text{ for } d-k \leq j < l \tag{33}$$

First we show the base case $Q(d-k+1)$. For this we choose a $k-$pair $\mathbf{C}, \widehat{\mathbf{C}}$ such that $\mathbf{C}$ has support $S = \{d-k, d-k+2, d-k+3, ..., d\}$. It is true that $\mathbf{X}_{:,i} = \mathbf{0}$ for $i = 1, ..., d-k-1$ hence $\widehat{\mathbf{C}}_{:,i} = \mathbf{0}$ for $i \leq d-k-1$ and therefore the node $d-k$ doesn't have any non-zero parents, since also $\overline{\hat{a}}_{j(d-k)} = 0$ for $d-k < j$ from previous claim. Therefore $\widehat{\mathbf{C}}_{:,d-k} = \mathbf{X}_{:,d-k} = \mathbf{C}_{:,d-k}$. Also, for $l = d-k+1$

$$\mathbf{X}_{:,l} = \overline{a}_{d-k,l} \mathbf{C}_{:,d-k} = \overline{a}_{d-k,l} \widehat{\mathbf{C}}_{:,d-k} \tag{34}$$

The equation from $\widehat{\mathbf{A}}$ gives:

$$\mathbf{X}_{:,l} = \sum_{j=d-k}^{d} \overline{\hat{a}}_{jl} \widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,l} \Rightarrow \left(\overline{a}_{d-k,l} - \overline{\hat{a}}_{d-k,l}\right) \widehat{\mathbf{C}}_{:,d-k} + \sum_{j=d-k+1}^{d} \overline{\hat{a}}_{jl} \widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,l} = \mathbf{0} \tag{35}$$

From the linear Independence Lemma C.8 of the support of $\widehat{\mathbf{C}}$ we necessarily need to have $\widehat{\mathbf{C}}_{:,l} = \mathbf{0}$, $\overline{a}_{d-k,l} = \overline{\hat{a}}_{d-k,l}$ and $\overline{\hat{a}}_{jl} = 0$ for $l < j$ which gives the base case.

For the rest of the induction we proceed in a similar manner. We assume with strong induction that all $Q(d-k+1), ..., Q(l)$ are true and proceed to prove $Q(l+1)$. Given these assumptions we have that

$$\overline{\hat{a}}_{ji} = \overline{a}_{ji} \text{ for } d-k \leq j < i,\ d-k \leq i \leq l \text{ and } \overline{\hat{a}}_{ji} = 0 \text{ for } i < j,\ d-k \leq i \leq l \tag{36}$$

Consider spectral support $S = \{d-k, d-k+1, ..., l, l+2, ..., d\}$ (the $(l+1)-$th column is $\mathbf{0}$) for the $k-$pair $\mathbf{C}, \widehat{\mathbf{C}}$. Then we have the equations:

$$\mathbf{X}_{:,d-k} = \mathbf{C}_{:,d-k} = \widehat{\mathbf{C}}_{:,d-k} \tag{37}$$

$$\mathbf{X}_{:,d-k+1} = \overline{a}_{d-k,d-k+1} \mathbf{C}_{:,d-k} + \mathbf{C}_{:,d-k+1} = \overline{\hat{a}}_{d-k,d-k+1} \widehat{\mathbf{C}}_{:,d-k} + \widehat{\mathbf{C}}_{:,d-k+1} \Rightarrow \widehat{\mathbf{C}}_{:,d-k+1} = \mathbf{C}_{:,d-k+1} \tag{38}$$

$$\vdots \tag{39}$$

$$\mathbf{X}_{:,l} = \sum_{j=d-k}^{l-1} \overline{a}_{jl} \mathbf{C}_{:,j} + \mathbf{C}_{:,l} = \sum_{j=d-k}^{l-1} \overline{\hat{a}}_{jl} \widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,l} \Rightarrow \widehat{\mathbf{C}}_{:,l} = \mathbf{C}_{:,l} \tag{40}$$

Where we used the linear independence lemma and sequentially proved that the spectral columns up to l are equal. The equation for the $(l+1)-$th column now becomes:

$$\mathbf{X}_{:,l+1} = \sum_{j=d-k}^{l} \overline{a}_{j,l+1}\mathbf{C}_{:,j} + \mathbf{C}_{:,l} = \sum_{j=d-k}^{d} \overline{\hat{a}}_{j,l+1}\widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,l+1} \tag{41}$$

$$\Leftrightarrow \sum_{j=d-k}^{d} \left(\overline{\hat{a}}_{j,l+1} - \overline{a}_{j,l+1}\right)\widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,l+1} = \mathbf{0} \Rightarrow \begin{cases} \overline{\hat{a}}_{j,l+1} = 0 \text{ for } l+1 < j \\ \overline{\hat{a}}_{j,l+1} = \overline{a}_{j,l+1} \text{ for } j < l+1 \\ \widehat{\mathbf{C}}_{:,l+1} = \mathbf{0} \end{cases} \tag{42}$$

where the last set of equalities follows from linear independence. This concludes the induction and the proof. □

To complete the proof of Theorem 3.2 it remains to show the following proposition.

Proposition C.16. *If both* $\mathbf{A}, \widehat{\mathbf{A}}$ *are upper triangular then* $\widehat{\mathbf{A}} = \mathbf{A}$.

For our proof we use the following definition.

Definition C.17. We denote by $\mathcal{P}_k$ the set of all $k-$pairs $\mathbf{C}_R, \widehat{\mathbf{C}}_R$ for all possible support patterns.

Now we proceed to the proof.

Proof. We will show equivalently that $\overline{\widehat{\mathbf{A}}} = \overline{\mathbf{A}}$ using two inductions. First we show for $l = 1, ..., k$ the following statement.

$P(l)$: For all $k-$pairs $\mathbf{C}, \widehat{\mathbf{C}}$ in $\mathcal{P}_k$ the first l non-zero columns $\mathbf{C}_{:,i_1}, \mathbf{C}_{:,i_2}, ..., \mathbf{C}_{:,i_l}$ and $\widehat{\mathbf{C}}_{:,\hat{i_1}}, \widehat{\mathbf{C}}_{:,\hat{i_2}}, ..., \widehat{\mathbf{C}}_{:,\hat{i_l}}$ are in the same positions, i.e. $i_j = \hat{i_j}$ and

- either they are respectively equal $\mathbf{C}_{:,i_j} = \widehat{\mathbf{C}}_{:,i_j}$
- or $\mathbf{C}_{:,i_l}$ is in the last possible index, namely $i_l = d - (l-1)$

For the base case $P(1)$, consider a $k-$pair $\mathbf{C}, \widehat{\mathbf{C}}$ and let i_1 be the position of the first non-zero spectral column of $\mathbf{C}$. Then $\mathbf{X}_{:,i} = \mathbf{0}$ for $i < i_1$ and therefore from Lemma C.15 $\widehat{\mathbf{C}}_{:,i} = \mathbf{0}$ for $i < i_1$. Hence

$$\mathbf{X}_{:,i_1} = \sum_{j<i_1} \overline{\hat{a}}_{ji_1}\widehat{\mathbf{C}}_{:,j} + \widehat{\mathbf{C}}_{:,i_1} = \widehat{\mathbf{C}}_{:,i_1} \tag{43}$$

Therefore $\widehat{\mathbf{C}}_{:,i_1} = \mathbf{C}_{:,i_1}$ and we proved $P(1)$, by satisfying both the positioning and the first requirement.

Assuming now that $P(l)$ holds, we will show $P(l+1)$. Take any $k-$pair of $\mathcal{P}_k$ which we denote by $\mathbf{C}, \widehat{\mathbf{C}}$. Then, if $\mathbf{C}_{:,i_l}$ is in the last possible position, then necessarily $\mathbf{C}_{:,i_{l+1}}$ is in the last possible position. Moreover, from the induction hypothesis the first l spectral columns are in the same positions. Therefore in the same manner, $\widehat{\mathbf{C}}_{:,i_l}$ is in the last position and $\widehat{\mathbf{C}}_{:,i_{l+1}}$, too. This case fulfills the desired statement.

If $\mathbf{C}_{:,i_l}$ is not in the last position, then from induction hypothesis, the first l spectral columns are equal. If $\mathbf{C}_{:,i_{l+1}}$ is in the last position and the same holds $\widehat{\mathbf{C}}_{:,\hat{i}_{l+1}}$, the requirement is satisfied. Otherwise $\hat{i}_{l+1} < i_{l+1}$ and the equation for $\hat{i}_{l+1}$ gives:

$$\mathbf{X}_{:,\hat{i}_{l+1}} = \sum_{j=1}^{l} \overline{\hat{a}}_{i_j,\hat{i}_{l+1}}\widehat{\mathbf{C}}_{:,i_j} + \widehat{\mathbf{C}}_{:,\hat{i}_{l+1}} = \sum_{j=1}^{l} \overline{a}_{i_j,\hat{i}_{l+1}}\mathbf{C}_{:,i_j} + \mathbf{0} \Leftrightarrow \sum_{j=1}^{l} \left(\overline{\hat{a}}_{i_j,\hat{i}_{l+1}} - \overline{a}_{i_j,\hat{i}_{l+1}}\right)\widehat{\mathbf{C}}_{:,i_j} + \widehat{\mathbf{C}}_{:,\hat{i}_{l+1}} = \mathbf{0} \tag{44}$$

According to linear independence Lemma C.8 we necessarily derive $\widehat{\mathbf{C}}_{:,\hat{i}_{l+1}} = \mathbf{0}$, absurd. Thus $\hat{i}_{l+1} = i_{l+1}$ and the induction statement is fullfilled in that case.

It remains to consider the case where $\mathbf{C}_{:,i_{l+1}}$ is not in the last position. Since, i_{l+1} is not the last position there exists a $k-$pair $\mathbf{C}', \widehat{\mathbf{C}}'$ such that the column i_{l+1} is zero and the $(l+1)-$ spectral column of $\mathbf{C}'$ lies at $i'_{l+1} > i_{l+1}$. The equation at i_{l+1} for $\mathbf{X}'$ gives:

$$\mathbf{X}'_{:,i_{l+1}} = \sum_{j=1}^{l} \overline{a}_{i_j,i_{l+1}} \mathbf{C}'_{:,i_j} = \sum_{j=1}^{l} \overline{\hat{a}}_{i_j,\hat{i}_{l+1}} \widehat{\mathbf{C}}'_{:,i_j} + \widehat{\mathbf{C}}'_{:,\hat{i}_{l+1}} \tag{45}$$

For the pair $\mathbf{C}', \widehat{\mathbf{C}}'$ the induction hypothesis holds, thus we derive:

$$\sum_{j=1}^{l} \left(\overline{\hat{a}}_{i_j,\hat{i}_{l+1}} - \overline{a}_{i_j,i_{l+1}} \right) \widehat{\mathbf{C}}'_{:,i_j} + \widehat{\mathbf{C}}'_{:,\hat{i}_{l+1}} = \mathbf{0} \tag{46}$$

Therefore, Lemma C.8 gives $\overline{\hat{a}}_{i_j,\hat{i}_{l+1}} = \overline{a}_{i_j,i_{l+1}}$ for all $j = 1, ..., l$ and returning back to the equation for $\mathbf{X}_{:,i_{l+1}}$ we derive:

$$\mathbf{X}_{:,i_{l+1}} = \sum_{j=1}^{l} \overline{a}_{i_j,i_{l+1}} \mathbf{C}_{:,i_j} + \mathbf{C}_{:,i_{l+1}} = \sum_{j=1}^{l} \overline{\hat{a}}_{i_j,\hat{i}_{l+1}} \widehat{\mathbf{C}}_{:,i_j} + \widehat{\mathbf{C}}_{:,\hat{i}_{l+1}} \tag{47}$$

$$\overset{P(l)}{=} \sum_{j=1}^{l} \overline{a}_{i_j,\hat{i}_{l+1}} \mathbf{C}_{:,i_j} + \widehat{\mathbf{C}}_{:,\hat{i}_{l+1}} \tag{48}$$

$$\Rightarrow \widehat{\mathbf{C}}_{:,\hat{i}_{l+1}} = \mathbf{C}_{:,\hat{i}_{l+1}} \tag{49}$$

which completes the induction step. Notice that $P(k)$ gives that for all $k-$pairs, the k spectral columns are in the same position. Given this fact we will now show that $\widehat{\mathbf{A}} = \mathbf{A}$. We will prove by induction that for $l = k-1, ..., 1, 0$

$$Q(l): \ \overline{\hat{a}}_{ij} = \overline{a}_{ij} \text{ for } 1 \leq i < j < d - l \tag{50}$$

To prove $Q(k-1)$ we choose all the $k-$pairs $\mathbf{C}, \widehat{\mathbf{C}}$, such that $\mathbf{C}_{:,i_2}$ is in the last possible position, $i_2 = d - k + 2$. Then for $i_1 \leq d - k$ the columns $\mathbf{C}_{:,i_1}, \widehat{\mathbf{C}}_{:,i_1}$ lie at the same position and are equal. Choosing $i_1 = i$ and computing the equation for $\mathbf{X}_{:,j}$ where $i < j \leq d - k + 1$ gives:

$$\mathbf{X}_{:,j} = \overline{a}_{ij} \mathbf{C}_{:,i} = \overline{\hat{a}}_{ij} \widehat{\mathbf{C}}_{:,i} = \overline{\hat{a}}_{ij} \mathbf{C}_{:,i} \tag{51}$$

Therefore $\overline{\hat{a}}_{ij} = \overline{a}_{ij}$ for all $1 \leq i < j \leq d - (k-1)$ and $Q(k-1)$ is satisfied. Next, assume that $Q(k-l)$ is true. We want to show $Q(k-l-1)$. Similarly to the base case we consider all $k-$pairs such that $\mathbf{C}_{:,l+2}$ lies in its last possible position $i_{l+2} = d - k + l + 2$, and $i_{l+1} \leq d - k + l$. Since the $(l+1)-$th column is not in the last position, from the previous induction we have that:

$$\begin{cases} \widehat{\mathbf{C}}_{:,i_1} = \mathbf{C}_{:,i_1} \\ \widehat{\mathbf{C}}_{:,i_2} = \mathbf{C}_{:,i_2} \\ \vdots \\ \widehat{\mathbf{C}}_{:,i_{l+1}} = \mathbf{C}_{:,i_{l+1}} \end{cases} \tag{52}$$

The equation for $d - k + l + 1$ gives:

$$\mathbf{X}_{:,d-k+l+1} = \sum_{j=1}^{l+1} \overline{a}_{i_j,d-k+l+1} \mathbf{C}_{:,i_j} \tag{53}$$

$$= \sum_{j=1}^{l+1} \overline{\hat{a}}_{i_j,d-k+l+1} \widehat{\mathbf{C}}_{:,i_j} \tag{54}$$

$$= \sum_{j=1}^{l+1} \overline{\hat{a}}_{i_j,d-k+l+1} \mathbf{C}_{:,i_j} \tag{55}$$

$$\xRightarrow{\text{Lemma C.8}} \overline{\hat{a}}_{i_j,d-k+l+1} = \overline{a}_{i_j,d-k+l+1} \tag{56}$$

By choosing all such possible $k-$pairs the indices i_j span all possibilities $1 \leq i < d - k + l + 1$. Combining this with $Q(k-l)$ we get that $Q(k-l-1)$ is true and the desired result follows. $\square$