Differentiable Causal Discovery with Smooth Acyclic Orientations

Riccardo Massidda [1] **Francesco Landolfi** [1] **Martina Cinquini** [1] **Davide Bacciu** [1]

Abstract

Most differentiable causal discovery approaches constrain or regularize an optimization problem using a continuous relaxation of the acyclicity property. The cost of computing the relaxation is cubic on the number of nodes and thus affects the scalability of such techniques. In this work, we introduce COSMO, the first quadratic and constraint-free continuous optimization scheme. COSMO represents a directed acyclic graph as a priority vector on the nodes and an adjacency matrix. We prove that the priority vector represents a differentiable approximation of the acyclic orientation of the graph, and we demonstrate the existence of an upper bound on the orientation acyclicity. In addition to being asymptotically faster, our empirical analysis highlights how COSMO performs comparably to constrained methods for graph discovery.

1. Introduction

Graphical approaches, such as Structural Causal Models (SCMs), emerged as the dominant framework to represent causal information about the world (Pearl, 2009). A fundamental problem in this context concerns the discovery of causal relations between a set of variables, i.e., the problem of identifying which arcs exist between nodes associated to the variables of interest (Spirtes et al., 2000). Continuous causal discovery techniques approach the problem by optimizing an acyclic causal graph (Vowels et al., 2022). In this context, a well-established methodology (Zheng et al., 2018) defines a smooth relaxation of the acyclicity property and frames causal discovery as a constrained optimization problem. Despite their widespread adoption, exact acyclicity constraints impose a cubic number of operations in the number of nodes that severely affects scalability.

Motivated by such limitation, we propose the first unconstrained optimization scheme to learn acyclic causal graphs from data in quadratic time. We represent Directed Acyclic Graphs (DAGs) as a differentiable function of a directed graph and a priority vector. By applying a tempered sigmoid to the pair-wise priority differences, we define a smooth relaxation of an acyclic orientation matrix where each node has an outgoing arc to all nodes with higher priority. In particular, our smooth orientation matrix is equal to the discrete orientation whenever the sigmoid temperature tends to zero. Thus, by annealing the temperature during training, the solution of the optimization problem is acyclic by construction. Further, since our approach only requires a quadratic number of operations per optimization step, its constraint-free scheme is significantly faster than existing methods. Overall, we refer to our approach as COSMO, for **C**ausal **O**rdering Discovery with **SM**ooth Acyclic **O**rientations.

2. Background and Related Works

Graphical causal models represents causal relations between d variables as a DAG. In the linear case, a Structural Causal Model (SCM) consists of a weighted acyclic adjacency matrix $\mathbf{W} \in \mathbb{R}^{d\times d}$ such that, for any random vector $\mathbf{x} \in \mathbb{R}^d$, it holds $\mathbf{x} = \mathbf{W}^\top\mathbf{x} + \mathbf{z}$, given an unobserved noise vector $\mathbf{z}$. NOTEARS (Zheng et al., 2018) formalizes causal discovery as the constrained optimization problem

$$\min_{\mathbf{W}\in\mathbb{R}^{d\times d}} \mathcal{L}(\mathbf{X}, \mathbf{W}^\top\mathbf{X}) + \lambda\|\mathbf{W}\|_1 \tag{1}$$

$$\text{s.t. } \operatorname{tr}(e^{\mathbf{W}\circ\mathbf{W}}) - d = 0 \tag{2}$$

where $\mathcal{L}$ is the Mean Squared Loss (MSE). In particular, the constraint equals zero if and only if the weighted adjacency matrix $\mathbf{W}$ is acyclic. The authors propose to solve the problem using the Augmented Lagrangian method (Nocedal & Wright, 1999), which in turn requires to solve multiple unconstrained problems and to compute the constraint value at each optimization step. Because of the matrix exponential, computing the constraint requires $O(d^3)$ operations.

Subsequent works mostly tried to tackle the computational cost of NOTEARS by either: (i) replacing the matrix exponential with theoretically cubic but practically faster operations,(ii) approximating the constraint,or (iii) defining an unconstrained problem that starts from the solution of a shorter constrained problem.We report the relevant related works along with their computational complexity in Table 1.

[1]Department of Computer Science, University of Pisa, Italy. Correspondence to: Riccardo Massidda <riccardo.massidda@phd.unipi.it>.

Published at the Differentiable Almost Everything Workshop of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. July 2023.

Table 1. Summary comparison of our proposal with related works. To express computational complexity, we define d as the number of nodes and k as the maximum length of iterative approaches. [†]: NOCURL requires a preliminary solution obtained from a cubic-expensive constrained optimization problem.

Method	Complexity	Constraint
NOTEARS (Zheng et al., 2018)	$O(d^3)$	Exact
DAGMA (Bello et al., 2022)	$O(d^3)$	Exact
NOBEARS (Lee et al., 2019)	$O(kd^2)$	Approximated
TMPI (Zhang et al., 2022)	$O(kd^2)$	Approximated
NOCURL (Yu et al., 2021)	$O(d^2)$†	Partial
COSMO (Ours)	$\boldsymbol{O(d^2)}$	**None**

Notably, while recent literature disputes the relevance of plainly minimizing MSE to uncover causal relations in real-world scenarios (Reisach et al., 2021; Kaiser & Sipos, 2022). NOTEARS-like formulations still have a relevant role as a building block in more complex continuous discovery approaches (Lachapelle et al., 2020; Brouillard et al., 2020).

3. Learning Acyclic Orientations with COSMO

To continuously represent the space of d-dimensional DAGs, we introduce a priority vector $\mathbf{p} \in \mathbb{R}^d$. on its vertices V. Consequently, given a strictly positive threshold $\varepsilon > 0$, we define the strict partial order $\prec_{\mathbf{p},\varepsilon}$ as

$$u \prec_{\mathbf{p},\varepsilon} v \iff \mathbf{p}_v - \mathbf{p}_u \geq \varepsilon \tag{3}$$

for any $u, v \in V$. In other terms, a vertex u precedes another vertex v if and only if the priority of v is sufficiently larger than the priority of the vertex u. Notably, with a zero threshold $\varepsilon = 0$, the relation would be symmetric and thus not a strict order. On the other hand, whenever ε is strictly positive, we can represent a subset of all strict partial orders sufficient to express all possible DAGs.

Lemma 3.1. *Let $\mathbf{W} \in \mathbb{R}^{d\times d}$ be a real matrix. Then, for any $\varepsilon > 0$, $\mathbf{W}$ is the weighted adjacency matrix of a DAG if and only if it exists a priority vector $\mathbf{p} \in \mathbb{R}^d$ and a real matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$ such that*

$$\mathbf{W} = \mathbf{H} \circ \mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}, \tag{4}$$

where $\mathbf{T}_{\prec_{\mathbf{p},\varepsilon}} \in \{0,1\}^{d\times d}$ is a binary orientation matrix such that

$$\mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}[uv] = \begin{cases} 1 & \text{if } u \prec_{\mathbf{p},\varepsilon} v \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

for any $u, v \in V$.

Proof. We report the proof in Appendix A.1. □

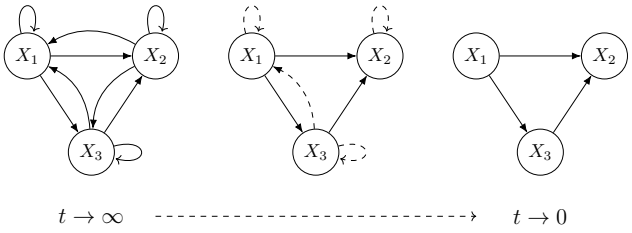

Figure 1. (*left*) With infinite temperature, the sigmoid function is constant and connects all vertices. (*center*) Given two nodes, for positive temperatures the smooth orientation matrix has larger values on the arcs respecting the priority ordering. (*right*) In the limit of the temperature to zero, the smooth orientation matrix contains non-zero entries if and only if the arc respects the order, i.e., it directs a node to another with sufficiently higher priority.

While priority vectors enable the representation of strict partial orders in a continuous space, the construction of the orientation matrix still requires the non-differentiable evaluation of the inequality in Equation 3. To this end, we approximate the comparison of the difference against the threshold ε, using a *tempered* sigmoidal function. We refer to such approximation of the orientation matrix as the *smooth* orientation matrix.

Definition 3.2 (Smooth Orientation Matrix)**.** Let $\mathbf{p} \in \mathbb{R}^d$ be a priority vector, $\varepsilon > 0$ be a strictly positive threshold, and $t > 0$ be a strictly positive temperature. Then, the *smooth* orientation matrix of the strict partial order $\prec_{\mathbf{p},\varepsilon}$ is the real matrix $S_{t,\varepsilon}(\mathbf{p}) \in \mathbb{R}^{d\times d}$ such that, for any $u, v \in V$, it holds

$$S_{t,\varepsilon}(\mathbf{p})_{uv} = \sigma_{t,\varepsilon}(\mathbf{p}_v - \mathbf{p}_u), \tag{6}$$

where $\sigma_{t,\varepsilon}$ is the ε-centered tempered sigmoid, defined as

$$\sigma_{t,\varepsilon}(x) = \frac{1}{1 + e^{-(x-\varepsilon)/t}}. \tag{7}$$

Intuitively, the threshold ε shifts the center of the sigmoid and breaks the symmetry whenever two variables *approximately* have the same priority. The temperature parameter $t > 0$ regulates instead the steepness of the sigmoid. Because of the asymmetry introduced by the threshold, in the limit of the temperature to zero, the zero-entries of a smooth orientation matrix coincide with the zero-entries of the corresponding orientation matrix (Figure 1). Therefore, we prove that any directed acyclic graph can be represented as the element-wise product of a directed adjacency matrix and a smooth orientation. Further, any directed graph resulting from this decomposition is acyclic.

Theorem 3.3. *Let $\mathbf{W} \in \mathbb{R}^{d\times d}$ be a real matrix. Then, for any $\varepsilon > 0$, $\mathbf{W}$ is the weighted adjacency matrix of a DAG if and only if it exists a priority vector $\mathbf{p} \in \mathbb{R}^d$ and a real matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$ such that*

$$\mathbf{W} = \mathbf{H} \circ \lim_{t\to 0} S_{t,\varepsilon}(\mathbf{p}), \tag{8}$$

where $S_{t,\varepsilon}(\mathbf{p})$ is the smooth orientation matrix of $\prec_{\mathbf{p},\varepsilon}$.

Proof. We report the proof in Appendix A.2. □

Given our definition of smooth acyclic orientation, we can effectively parameterize the space of DAGs as a continuous function of a weighted adjacency matrix and a priority vector. Formally, we propose to decompose the weight matrix as the product

$$\mathbf{W} = \mathbf{H} \circ S_{t,\varepsilon}(\mathbf{p}), \tag{9}$$

where $\mathbf{H} \in \mathbb{R}^{d\times d}$ and $\mathbf{p} \in \mathbb{R}^d$. In line with previous work, we propose to address causal discovery as a score-based method minimizing a loss function on the observations. By jointly learning adjacencies and priorities, we avoid the use of acyclicity constraints and reduce to a unique unconstrained problem. Therefore, the computational complexity of our solution reduces to the construction of the weighted matrix $\mathbf{W}$, which can be achieved in $O(d^2)$ time and space per optimization step.

Notably, the smooth orientation matrix $S_{t,\varepsilon}(\mathbf{p})$ represents an acyclic orientation only in the limit of the temperature $t \to 0$. Nonetheless, the gradient loss vanishes whenever the temperature tends to zero. In fact, for a given loss function $\mathcal{L}$, the gradient of the priority vector $\mathbf{p}$ has form

$$\begin{aligned}\frac{\partial \mathcal{L}(\mathbf{X};\mathbf{W}^\top\mathbf{X})}{\partial \mathbf{p}_u} = &\sum_{v\in V} \frac{\partial \mathcal{L}(\mathbf{X};\mathbf{W}^\top\mathbf{X})}{\partial \mathbf{W}_{uv}} \cdot \frac{\partial \mathbf{W}_{uv}}{\partial \mathbf{p}_u} \\ &+ \frac{\partial \mathcal{L}(\mathbf{X};\mathbf{W}^\top\mathbf{X})}{\partial \mathbf{W}_{vu}} \cdot \frac{\partial \mathbf{W}_{vu}}{\partial \mathbf{p}_u},\end{aligned} \tag{10}$$

and, for each partial derivative $\partial \mathbf{W}_{uv}/\partial \mathbf{p}_u$, it holds that

$$\begin{aligned}\lim_{t\to 0} \frac{\partial \mathbf{W}_{uv}}{\partial \mathbf{p}_u} &= \lim_{t\to 0} -\frac{\mathbf{H}_{uv}}{t}\sigma_{t,\varepsilon}(\mathbf{p}_v - \mathbf{p}_u) \\ &\quad \cdot (1 - \sigma_{t,\varepsilon}(\mathbf{p}_v - \mathbf{p}_u)) \\ &= 0,\end{aligned} \tag{11}$$

and, similarly, $\partial \mathbf{W}_{vu}/\partial \mathbf{p}_u$ tends to zero.

To handle this issue, we tackle the optimization problem by progressively reducing the temperature during training. In practice, we perform cosine annealing from an initial positive value t_{start} to a significantly lower target value $t_{\text{end}} \approx 0$. Further, we prove the existence of an upper bound on the acyclicity of the orientation matrix that is a monotone increasing function of the temperature t. Therefore, annealing the temperature during training decreases the upper bound on the *smooth* orientation acyclicity.

Theorem 3.4. *Let $\mathbf{p} \in \mathbb{R}^d$ be a priority vector, $\varepsilon > 0$ be a strictly positive threshold, and $t > 0$ be a strictly positive temperature. Then, given the smooth orientation matrix $S_{t,\varepsilon}(\mathbf{p}) \in \mathbb{R}^{d\times d}$, it holds*

$$h(S_{t,\varepsilon}(\mathbf{p})) \leq e^{d\alpha} - 1, \tag{12}$$

where $h(S_{t,\varepsilon}(\mathbf{p})) = \operatorname{tr}(e^{S_{t,\varepsilon}(\mathbf{p})}) - d$ is the NOTEARS *acyclicity constraint and $\alpha = \sigma_{t,\varepsilon}(0) = \sigma(-\varepsilon/t)$.*

Proof. We report the proof in Appendix B. □

To contrast the discovery of spurious causal relations, we apply L1 regularization on the adjacency matrix $\mathbf{H}$ in order to perform feature selection. Further, during the annealing procedure, even if a vertex u precedes v in the partial order $\prec_{\mathbf{p},\varepsilon}$, the weight of the opposite arc $v \to u$ in the smooth orientation matrix will only be approximately zero. Therefore, sufficiently large values of the weighted adjacency matrix $\mathbf{H}$, might still lead to cyclic paths during training. To avoid this issue, we regularize the L2-norm of the non-oriented adjacency matrix.

Further, we also observe that the partial derivative $\partial \mathbf{W}_{uv}/\partial \mathbf{p}_u$ tends to zero whenever the priorities difference $|\mathbf{p}_v - \mathbf{p}_u|$ tends to infinity. Therefore, we regularize the L2-norm of the priority vector. For the same reason, we initialize each component from the normal distribution $\mathbf{p}_u \sim \mathcal{N}(0, \varepsilon^2/2)$, so that each difference follows the normal distribution $\mathbf{p}_v - \mathbf{p}_u \sim \mathcal{N}(0, \varepsilon^2)$.

Overall, we formalize COSMO as the differentiable and unconstrained problem

$$\begin{aligned}\min_{\mathbf{H}\in\mathbb{R}^{d\times d}, \mathbf{p}\in\mathbb{R}^d} &\mathcal{L}(\mathbf{X}, (\mathbf{H} \circ S_{t,\varepsilon}(\mathbf{p}))^\top \mathbf{X}) \\ &+ \lambda_1\|\mathbf{H}\|_1 + \lambda_2\|\mathbf{H}\|_2 + \lambda_p\|\mathbf{p}\|_2,\end{aligned} \tag{13}$$

where $\lambda_1, \lambda_2, \lambda_p$ are the regularization coefficients for the adjacencies and the priorities. As the regularization coefficients $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \lambda_p\}$, the initial temperature t_{start}, the target temperature t_{end}, and the shift ε are hyperparameters of our proposal. Nonetheless, Theorem 3.4 can guide the choice of the final temperature value and the shift to guarantee a maximum tolerance on acyclicity. We delineate a possible strategy to model nonlinear relations in Appendix C.3.

4. Experiments

We discuss the experimental comparison of COSMO against related state-of-the-art approaches that optimize a DAG from data. Namely, we confront with the graph discovery performance and execution time of NOTEARS (Zheng et al., 2020), NOCURL (Yu et al., 2021), and DAGMA (Bello et al., 2022). NOCURL consists of two phases: firstly it solves a constrained and cubic-expensive optimization problem, then it improves the solution with a further unconstrained problem. We also compare with NOCURL-U, i.e., an enterely

Table 2. Experimental results on linear Erdős–Rényi simulated DAGs with different noise terms and sizes. For each algorithm, we report mean and standard deviation over five independent runs. We highlight in bold the **best** result and in italic bold the ***second best*** result.

		Gauss		Exp		Gumbel	
d	Algorithm	AUC	Time (s)	AUC	Time (s)	AUC	Time (s)
30	COSMO	***0.984 ± 0.02***	**88 ± 2**	**0.989 ± 0.01**	**89 ± 3**	0.914 ± 0.10	**87 ± 2**
	DAGMA	**0.985 ± 0.01**	781 ± 192	***0.986 ± 0.02***	744 ± 75	***0.973 ± 0.02***	787 ± 86
	NOCURL	0.967 ± 0.01	822 ± 15	0.956 ± 0.02	826 ± 24	0.915 ± 0.04	826 ± 17
	NOCURL-U	0.694 ± 0.06	***226 ± 5***	0.694 ± 0.05	***212 ± 5***	0.678 ± 0.05	***212 ± 5***
	NOTEARS	0.973 ± 0.02	5193 ± 170	0.966 ± 0.03	5579 ± 284	**0.981 ± 0.01**	5229 ± 338
100	COSMO	0.961 ± 0.03	**99 ± 2**	***0.985 ± 0.01***	**99 ± 2**	***0.973 ± 0.01***	**98 ± 1**
	DAGMA	**0.982 ± 0.01**	660 ± 141	**0.986 ± 0.01**	733 ± 109	**0.986 ± 0.01**	858 ± 101
	NOCURL	0.962 ± 0.01	1664 ± 14	0.950 ± 0.02	1655 ± 28	0.962 ± 0.01	1675 ± 34
	NOCURL-U	0.682 ± 0.05	***267 ± 10***	0.693 ± 0.05	***242 ± 4***	0.663 ± 0.04	***247 ± 9***
	NOTEARS	***0.963 ± 0.01***	11000 ± 339	0.972 ± 0.01	10880 ± 366	0.969 ± 0.00	11889 ± 343
500	COSMO	***0.933 ± 0.01***	**436 ± 81**	**0.986 ± 0.00**	**390 ± 102**	**0.982 ± 0.01**	**410 ± 106**
	DAGMA	**0.980 ± 0.00**	2485 ± 365	***0.984 ± 0.01***	2575 ± 469	***0.980 ± 0.00***	2853 ± 218
	NOCURL-U	0.683 ± 0.05	***1546 ± 304***	0.715 ± 0.03	***1488 ± 249***	0.728 ± 0.05	***1342 ± 209***

unconstrained variant of the latter that skips the preliminary constrained phase.

For each method, we perform causal discovery by minimizing the Mean Squared Error (MSE) of a model on observational data using the Adam optimizer (Kingma & Ba, 2015). Then, we measure the Area under the ROC Curve (AUC) between the absolute value of the weighted adjacency matrix and the ground truth graph (Heinze-Deml et al., 2018).

By looking at the AUC of the identified causal graphs, we observe that COSMO consistently achieves results comparable with constrained-optimization solutions such as DAGMA or NOTEARS (Table 2). Furthermore, COSMO performs better than NOCURL on most datasets. As pointed out by its authors, we also observe that the discovery performance of NOCURL drops when optimizing the variable ordering instead of inferring it from a preliminary solution. The fact that COSMO outperforms NOCURL-U on all datasets highlights the substantial role and effect of our *smooth* orientation formulation for learning the topological ordering of variables from data in an unconstrained way.

Unsurprisingly, due to its quadratic computational complexity, COSMO is significantly faster than constrained methods on all datasets, especially for increasing graph sizes. Other than requiring multiple optimization problems, all competing methods incur in an higher computational cost per step (Figure 2). Therefore, already for graphs with 500 nodes, only COSMO, DAGMA, and NOCURL-U return a solution before hitting our wall time limit.

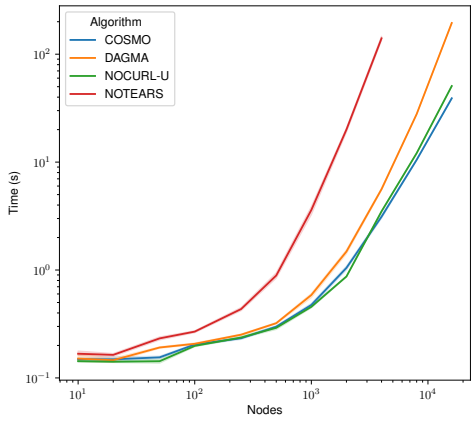

Figure 2. Average duration of a training epoch for an increasing number of nodes. We compute the optimization time on a random ER-4 graph for five iterations over five independent repetitions.

5. Conclusion

We introduced COSMO, an unconstrained and continuous approach for learning an acyclic causal graph from observational data. Our novel definition of a *smooth* orientation matrix ensures acyclicity of the solution without requiring the evaluation of computationally expensive constraints. We prove that annealing the temperature of our smooth acyclic orientation corresponds to decreasing an upper bound on the widely adopted acyclicity relaxation from NOTEARS. Our empirical analysis showed that COSMO performs comparably to constrained methods in significantly less time. Furthermore, the analysis highlights the role of our parameterization, which does not incur the necessity of preliminary solutions and solves a unique unconstrained problem. Overall, COSMO opens up more scalable and more causally grounded continuous causal discovery strategies, without sacrificing — as demonstrated in this work — the theoretical guarantees on DAGs approximation capabilities.

References

Bello, K., Aragam, B., and Ravikumar, P. DAGMA: Learning DAGs via M-matrices and a Log-Determinant Acyclicity Characterization. In *Advances in Neural Information Processing Systems*, 2022.

Brouillard, P., Lachapelle, S., Lacoste, A., Lacoste-Julien, S., and Drouin, A. Differentiable causal discovery from interventional data. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21865–21877. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f8b7aa3a0d349d9562b424160ad18612-Paper.pdf.

Heinze-Deml, C., Maathuis, M. H., and Meinshausen, N. Causal structure learning. *Annual Review of Statistics and Its Application*, 5:371–391, 2018.

Kaiser, M. and Sipos, M. Unsuitability of notears for causal graph discovery when dealing with dimensional quantities. *Neural Processing Letters*, 54(3):1587–1595, 2022.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Lachapelle, S., Brouillard, P., Deleu, T., and Lacoste-Julien, S. Gradient-based neural DAG learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=rklbKA4YDS.

Lee, H.-C., Danieletto, M., Miotto, R., Cherng, S. T., and Dudley, J. T. Scaling structural learning with no-bears to infer causal transcriptome networks. In *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2020*, pp. 391–402. World Scientific, 2019.

Lopez, R., Hütter, J.-C., Pritchard, J., and Regev, A. Large-scale differentiable causal discovery of factor graphs. *Advances in Neural Information Processing Systems*, 35: 19290–19303, 2022.

Nocedal, J. and Wright, S. J. *Numerical optimization*. Springer, 1999.

Pearl, J. *Causality*. Cambridge university press, 2009.

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

Spirtes, P., Glymour, C., and Scheines, R. *Causation, Prediction, and Search, Second Edition*. Adaptive computation and machine learning. MIT Press, 2000.

Vowels, M. J., Camgoz, N. C., and Bowden, R. D'ya like dags? a survey on structure learning and causal discovery. *ACM Computing Surveys*, 55(4):1–36, 2022.

Yu, Y., Gao, T., Yin, N., and Ji, Q. Dags with no curl: An efficient dag structure learning approach. In *International Conference on Machine Learning*, pp. 12156–12166. PMLR, 2021.

Zhang, Z., Ng, I., Gong, D., Liu, Y., Abbasnejad, E., Gong, M., Zhang, K., and Shi, J. Q. Truncated matrix power iteration for differentiable dag learning. *Advances in Neural Information Processing Systems*, 35:18390–18402, 2022.

Zheng, X., Aragam, B., Ravikumar, P. K., and Xing, E. P. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

Zheng, X., Dan, C., Aragam, B., Ravikumar, P., and Xing, E. Learning sparse nonparametric DAGs. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pp. 3414–3425. PMLR, 2020. URL https://proceedings.mlr.press/v108/zheng20a.html. ISSN: 2640-3498.

A. Deferred Proofs

A.1. Proof of Lemma 3.1

Lemma 3.1 *Let $\mathbf{W} \in \mathbb{R}^{d\times d}$ be a real matrix. Then, for any $\varepsilon > 0$, $\mathbf{W}$ is the weighted adjacency matrix of a DAG if and only if it exists a priority vector $\mathbf{p} \in \mathbb{R}^d$ and a real matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$ such that*

$$\mathbf{W} = \mathbf{H} \circ \mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}, \tag{14}$$

where $\mathbf{T}_{\prec_{\mathbf{p},\varepsilon}} \in \{0,1\}^{d\times d}$ is a binary orientation matrix such that

$$\mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}[uv] = \begin{cases} 1 & \text{if } u \prec_{\mathbf{p},\varepsilon} v \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

for any $u, v \in V$.

Proof. Firstly, we prove the existence of a priority vector $\mathbf{p}$ and an adjacency matrix $\mathbf{H}$ for each weighted acyclic matrix $\mathbf{W}$ of a directed acyclic graph $D = (V, A)$. Being a DAG, the arcs follow a strict partial order $\prec$ on the vertices $V = \{1, \dots, d\}$. Therefore, it holds that

$$A \subseteq \{(u, v) \mid u \prec v\}. \tag{16}$$

Consequently, for an arbitrary topological ordering of the variables $\pi \colon V \to \{1, \dots, d\}$, which always exists on DAGs, we define the vector $\mathbf{p} \in \mathbb{R}^d$ such that

$$\mathbf{p}_u = \varepsilon\pi(u). \tag{17}$$

Given the following implications

$$\begin{aligned} u \prec v &\implies \pi(v) > \pi(u) && (18)\\ &\implies \mathbf{p}_v - \mathbf{p}_u = \varepsilon(\pi(v) - \pi(u)) \geq \varepsilon && (19)\\ &\iff u \prec_{\mathbf{p},\varepsilon} v, && (20) \end{aligned}$$

it holds that the order $\prec_{\mathbf{p},\varepsilon}$ contains the order $\prec$. Finally, we can define the adjacency matrix as $\mathbf{H} = \mathbf{W}$, where $\mathbf{W} = \mathbf{H} \circ \mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}$ holds since $\mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}[u, v] = 0$ only if $(u, v) \notin A$.

To prove that any priority vector $\mathbf{p}$ and adjacency matrix $\mathbf{H}$ represent a DAG, we first notice that, since the arcs follow a strict partial order, the orientation $\mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}$ is acyclic. Then, by element-wise multiplying any matrix $\mathbf{H}$ we obtain a sub-graph of a DAG, which is acyclic by definition. □

A.2. Proof of Theorem 3.3

Theorem 3.3 *Let $\mathbf{W} \in \mathbb{R}^{d\times d}$ be a real matrix. Then, for any $\varepsilon > 0$, $\mathbf{W}$ is the weighted adjacency matrix of a DAG if and only if it exists a priority vector $\mathbf{p} \in \mathbb{R}^d$ and a real matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$ such that*

$$\mathbf{W} = \mathbf{H} \circ \lim_{t\to 0} S_{t,\varepsilon}(\mathbf{p}), \tag{21}$$

where $S_{t,\varepsilon}(\mathbf{p})$ is the smooth orientation matrix of $\prec_{\mathbf{p},\varepsilon}$.

Proof. By Lemma 3.1, we know that for any acyclic weighted adjacency matrix $\mathbf{W}$ there exist a priority vector $\mathbf{p}$ and a real matrix $\mathbf{H}$ such that $\mathbf{W} = \mathbf{H} \circ \mathbf{T}_{\prec_{\mathbf{p},\varepsilon}}$. Further, by Definition 3.2, the inner limit of Equation 21 solves to

$$\lim_{t\to 0} S_{t,\varepsilon}(\mathbf{p})_{uv} = \begin{cases} 1 & \mathbf{p}_v - \mathbf{p}_u > \varepsilon \\ 1/2 & \mathbf{p}_v - \mathbf{p}_u = \varepsilon \\ 0 & \mathbf{p}_v - \mathbf{p}_u < \varepsilon. \end{cases} \tag{22}$$

Therefore, we can define $\mathbf{H}' \in \mathbb{R}^{d\times d}$ such that

$$\mathbf{H}_{uv} = \begin{cases} 2\mathbf{H}'_{uv} & \mathbf{p}_v - \mathbf{p}_u = \varepsilon, \\ \mathbf{H}'_{uv} & \text{otherwise.} \end{cases}, \tag{23}$$

from which

$$\mathbf{W} = \mathbf{H} \circ \mathbf{T}_{\prec_{\mathbf{p},\varepsilon}} = \mathbf{H}' \circ \lim_{t\to 0} \mathbf{S}_{t,\varepsilon}(\mathbf{p}). \tag{24}$$

Then, to prove the counter-implication of Theorem 3.3, we notice that

$$\lim_{t\to 0} \mathbf{S}_{t,\varepsilon}(\mathbf{p})_{uv} = 0 \iff \mathbf{p}_v - \mathbf{p}_u < \varepsilon \iff u \not\prec_{\mathbf{p},\varepsilon} v. \tag{25}$$

Therefore, since the smooth orientation contains an arc if and only if the vertex respect the strict partial order $\prec_{\mathbf{p},\varepsilon}$, it is acyclic. Consequently, as in Lemma 3.1, the element-wise product with an acyclic matrix results in a sub-graph of a DAG, which is also acyclic by definition. □

A.3. Priority Vector Initialization

By independently sampling each priority component from a Normal distribution $\mathcal{N}(\mu, s^2/2)$, each difference is consequently sampled from the distribution $\mathcal{N}(0, s^2)$. Therefore, we seek a value for the standard deviation s that maximizes the partial derivative

$$\frac{\partial \mathbf{W}_{uv}}{\partial \mathbf{p}_u} = \frac{\mathbf{H}_{uv}}{t}\sigma_{t,\varepsilon}(\mathbf{p}_v - \mathbf{p}_u)(1 - \sigma_{t,\varepsilon}(\mathbf{p}_v - \mathbf{p}_u)). \tag{26}$$

for arbitrary vertices u, v. Given the definition of the tempered-shifted sigmoid function, this object has maximum in $\mathbf{p}_v - \mathbf{p}_u = \varepsilon$. Therefore, by setting the variance as $s^2 = \varepsilon^2$, we maximize the density function of the point $\mathbf{p}_v - \mathbf{p}_u = \varepsilon$ in the distribution $\mathcal{N}(0, s^2)$.

B. Smooth Acyclic Orientations and the Acyclicity Constraint

In this section, we present the proof for the upper bound on the acyclicity of a smooth acyclic orientation matrix. To this end, we introduce two auxiliary and novel lemmas. Firstly, we introduce a lemma which binds the product of a sigmoid on a sequence of values with zero sum (Lemma B.1). Then, we introduce another lemma on the sum of the priority differences in a cyclic path (Lemma B.2). Finally, we are able to prove the acyclicity upper bound from Theorem 3.4.

Lemma B.1. *(Sigmoid Product Upper Bound) Let $\{x_i\}$ be a sequence of n real numbers such that*

$$\sum_{i=1}^{n} x_i = 0.$$

Then, for any temperature $t > 0$ and shift $\varepsilon \geq 0$, it holds that

$$\prod_{i=1}^{n} \sigma_{t,\varepsilon}(x_i) \leq \alpha^n,$$

where $\alpha = \sigma(-\varepsilon/t)$ is the value of the tempered and shifted sigmoid in zero.

Proof. Before starting, we invite the reader to notice that, for any temperature $t > 0$, if the sum of the sequence $\{x_i\}$ is zero, then also the sequence $\{x_i/t\}$ sums to zero. Therefore, we omit the temperature in the following proof, and assume to divide beforehand all elements of the sequence by the temperature t. Further, we explicitly denote the shifted sigmoid by using the notation $\sigma(x_i - \varepsilon)$.

Firstly, we formulate the left-side of the inequality as

$$\prod_{i=1}^{n} \sigma(x_i - \varepsilon) = \prod_{i=1}^{n} \frac{e^{x_i-\varepsilon}}{1 + e^{x_i-\varepsilon}} = \frac{\prod_{i=1}^{n} e^{x_i-\varepsilon}}{\prod_{i=1}^{n} 1 + e^{x_i-\varepsilon}} = \frac{e^{\sum_{i=1}^{n} x_i-\varepsilon}}{\prod_{i=1}^{n} 1 + e^{x_i-\varepsilon}} = \frac{e^{-n\varepsilon}}{\prod_{i=1}^{n} 1 + e^{x_i-\varepsilon}}.$$

Similarly, we rewrite the right side as

$$\alpha^n = \sigma(-\varepsilon)^n = \left(\frac{e^{-\varepsilon}}{\prod_{i=1}^n 1 + e^{-\varepsilon}}\right)^n = \frac{e^{-n\varepsilon}}{(1+e^{-\varepsilon})^n}.$$

Therefore, proving the left-side smaller or equal than the right-side, reduces to proving the left-denominator is larger than the right-denominator. Formally,

$$\prod_{i=1}^n 1 + e^{x_i - \varepsilon} \geq \left(1 + e^{-\varepsilon}\right)^n,$$

or equivalently, by applying the logarithmic function,

$$\sum_{i=1}^n \log(1 + e^{x_i - \varepsilon}) \geq n \log(1 + e^{-\varepsilon}). \tag{27}$$

To further ease the notation, we refer to the left side of inequality 27, as the target function

$$T(x) = \sum_{i=1}^n \log(1 + e^{x_i - \varepsilon}).$$

In particular, to prove 27, we show that

$$\min_x T(x) = n \log(1 + e^{-\varepsilon}), \tag{28}$$

for $x = \vec{0}$, which is the only stationary point due to the convexity of the target function.

Without loss of generality, we derive the partial derivative of the component x_1 on the target function $T(x)$. To constraint the components sum to zero, we consider the components $\{x_i\}$ for $i > 2$ as free, and then $x_2 = -x_1 - \sum_{i=3}^n x_i$ as a function of the remaining. The choice of x_1, x_2 is independent from the components ordering, and thus applies to any possible pair. Consequently,

$$\frac{\partial T(x)}{\partial x_1} = \frac{\partial(\log(1 + e^{x_1 - \varepsilon}) + \log(1 + e^{-x_1 - \sum_{i=3}^n x_i - \varepsilon}) + \sum_{i=3}^n \log(1 + e^{x_i - \varepsilon}))}{\partial x_1} \tag{29}$$

$$= \frac{\partial(\log(1 + e^{x_1 - \varepsilon}) + \log(1 + e^{-x_1 - \sum_{i=3}^n x_i - \varepsilon})}{\partial x_1} \tag{30}$$

$$= \sigma(x_1 - \varepsilon) - \sigma(-x_1 - \sum_{i=3}^n x_i - \varepsilon). \tag{31}$$

Since $\sigma(-\varepsilon) = \sigma(-\varepsilon)$, the equation is satisfied, for any component x_i, by $x = \vec{0}$,

We finally prove Inequality 27, by showing that the value of the target function $T(x)$, in its only stationary point $x = \vec{0}$, equals the bound. Formally,

$$T(\vec{0}) = \sum_{i=1}^n \log(1 + e^{-\varepsilon}) \tag{32}$$

$$= n \log(1 + e^{-\varepsilon}). \tag{33}$$

□

Lemma B.2. *(Sum of Differences in Cycle) Let $\{p_i\}$ be a sequence of $n+1$ real numbers such that $p_1 = p_{n+1}$. Then, let $\{\delta_i\}$ be a sequence of n numbers such that $\delta_i = p_{i+1} - p_i$. Then,*

$$\sum_{i=1}^n \delta_i = 0. \tag{34}$$

Proof. The proof is immediate from the following sequence of equations:

$$\sum_{i=1}^{n} \delta_i = \sum_{i=1}^{n} p_{i+1} - p_i = -p_1 + \sum_{i=2}^{n} p_i - p_i + p_{n+1} = 0.$$

□

Theorem B.3. *(Orientation Acyclicity Upper Bound) Let $\mathbf{W} \in \mathbb{R}^{d\times d}$ be a real matrix. Then, for any $\varepsilon > 0$, $\mathbf{W}$ is the weighted adjacency matrix of a DAG if and only if it exists a priority vector $\mathbf{p} \in \mathbb{R}^d$ and a real matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$ such that*

$$\mathbf{W} = \mathbf{H} \circ \lim_{t\to 0} S_{t,\varepsilon}(\mathbf{p}), \tag{35}$$

where $S_{t,\varepsilon}(\mathbf{p})$ is the smooth orientation matrix of $\prec_{\mathbf{p},\varepsilon}$.

Proof. The left side of Inequality 35 corresponds to the following infinite series

$$\begin{aligned}\operatorname{tr}(e^{\mathbf{P}}) - d &= \sum_{k=0}^{\infty} \frac{1}{k!} \operatorname{tr}(\mathbf{P}^{(k)}) - d \\ &= \sum_{k=1}^{\infty} \frac{1}{k!} \operatorname{tr}(\mathbf{P}^{(k)})\end{aligned}$$

where $\mathbf{P}^{(k)}$ is the matrix power defined as $\mathbf{P}^{(k)} = \mathbf{P}^{(k-1)}\mathbf{P}$ and $\mathbf{P}^0 = \mathbf{I}$.

By definition of matrix power, the u-th element on the diagonal of $\mathbf{P}^{(k)}$ equals to

$$\begin{aligned}\mathbf{P}^{(k)}_{uu} &= \sum_{v_1\in V} \mathbf{P}^{(k-1)}_{v_1,u} \mathbf{P}_{u,v_1} \\ &= \sum_{v_1\in V} \cdots \sum_{v_{k-1}\in V} \mathbf{P}_{u,v_1} \left(\prod_{i=1}^{k-2} \mathbf{P}_{v_i,v_{i+1}}\right) \mathbf{P}_{v_{k-1},u}.\end{aligned}$$

Intuitively, the u-th element on the diagonal of $\mathbf{P}^{(k)}$ amounts to the sum of all possible paths starting and ending in the variable X_u. Therefore, being the same node, the priority of the first and the last node in the path are equal by definition. Consequently, by Lemma B.2, the difference between the priorities sums to zero. For this reason, given Lemma B.1, it holds that the product of the corresponding sigmoids is smaller or equal than α^k. Therefore,

$$\begin{aligned}\mathbf{P}^{(k)}_{uu} &= \sum_{v_1\in V} \cdots \sum_{v_{k-1}\in V} \mathbf{P}_{u,v_1} \left(\prod_{i=1}^{k-2} \mathbf{P}_{v_i,v_{i+1}}\right) \mathbf{P}_{v_{k-1},u} \\ &\leq \sum_{v_1\in V} \cdots \sum_{v_{k-1}\in V} \alpha^k \\ &= d^{k-1}\alpha^k.\end{aligned}$$

Consequently, we upper bound the trace of the orientation matrix power as

$$\operatorname{tr}(\mathbf{P}^{(k)}) = \sum_{u=1}^{d} \mathbf{P}^{(k)}_{uu} \leq d^k \alpha^k.$$

Finally, we are able to prove the Theorem as

$$\begin{aligned}\operatorname{tr}(e^{\mathbf{P}}) - d &= \sum_{k=0}^{\infty} \frac{1}{k!} \operatorname{tr}(\mathbf{P}^{(k)}) - d \\ &= \sum_{k=1}^{\infty} \frac{1}{k!} \operatorname{tr}(\mathbf{P}^{(k)}) \\ &\leq \sum_{k=1}^{\infty} \frac{1}{k!} d^k \alpha^k \\ &= -1 + e^{d\alpha},\end{aligned}$$

where the last passage is due to the Taylor series of the exponential function. □

C. Implementation Details

We run all the experiments on our internal cluster of Intel(R) Xeon(R) Gold 5120 processors, totaling 56 CPUs per machine. We report details on the evaluation (C.1), the data generation procedure (C.2), and the models (C.3).

C.1. Evaluation Procedure

We ensure a fair comparison by selecting the best hyperparameters for each implemented method on each dataset. We describe the hyperparameter space for each algorithm in the following subsections. Firstly, we sample fifty random configurations from the hyperparameter space. Since the hyperparameter space of COSMO also includes temperature and shift values, we extract more hyperparameters (200 – 800). Due to the significant speedup of COSMO, hyperparameter searches take a comparable amount of time, with NOTEARS being significantly longer on small graphs as well. Then, we test each configuration on five randomly sampled causal DAGs. We select the best hyperparameters according to the average AUC value. Finally, we perform a validation step by running the best configuration on five new random graphs.

Following previous work, we recover the binary adjacency matrix $\mathbf{A}$ of the retrieved causal graph by thresholding the learned weights $\mathbf{W}$ with a small constant $\omega = 0.3$. Formally, $\mathbf{A} = |\mathbf{W}| > \omega$.

C.2. Synthetic Data

As we remarked in the main body, continuous approaches are particularly susceptible to data normalization and might exploit variance ordering between variables (Reisach et al., 2021). Therefore, empirical results on simulated datasets that do not explicitly control this condition might not generalize to real-world scenarios. Nonetheless, our proposal aims at defining a faster parameterization that could replace existing continuous approaches as a building block in more complex discovery solutions. Therefore, as initially done by Zheng et al. (2018) and subsequent work, we tested COSMO and the remaining baselines in the usual synthetic testbed without normalizing the variance.

We include in our code the exact data generation process from the original implementation of NOTEARS.[1] Therefore, the dataset generation procedure firstly produces a DAG with either the Erdős–Rényi (ER) or the scale-free Barabási-Albert (SF) models. Then, it samples 1000 independent observations. In the linear case, the generator simulates equations of the form

$$f_i(x) = \mathbf{W}_i^{\top} x + z_i, \tag{36}$$

where we sample each weight $\mathbf{W}_{ij}$ from the uniform distribution $\mathcal{U}(-2, -0.5) \cup (0.5, 2)$ and each noise term z_i from either the Normal, Exponential ($\lambda = 1$), or Gumbel ($\mu = 0, \beta = 1$) distributions. In the non-linear case, we simulate an additive noise model with form

$$f_i(x) = g_i(x) + z_i, \tag{37}$$

where g_i is a randomly initialized Multilayer Perceptron (MLP) with 100 hidden units and the noise term z_i is sampled from the Normal Distribution $\mathcal{N}(0, 1)$.

[1] NOTEARS implementation is published with Apache license at https://github.com/xunzheng/notears.

Table 3. Hyperparameter ranges and values for COSMO.

Hyperparameter	Range/Value
Learning Rate	(1e-3, 1e-2)
λ_1	(1e-4, 1e-3)
λ_2	(1e-3, 5e-3)
λ_p	(1e-3, 3e-3)
t_{start}	0.45
t_{end}	(5e-4, 1e-3)
ε	(5e-3, 2e-2)

C.3. Models

Since we focus on the role of acyclic learners as a building block within more comprehensive discovery solutions, we slightly detach from experimental setups considering such algorithms as standalone structure learners. Therefore, instead of dealing with full-batch optimization, we perform mini-batch optimization with batch size $B = 64$. Similarly, instead of explicitly computing the gradient of the loss function, we implement all methods in PyTorch to exploit automatic differentiation. By avoiding differentiation and other overhead sources, the time expenses results are not directly comparable between our implementations and the results reported in the original papers. However, our implementation choices are common to works that employed NOTEARS *et similia* to ensure the acyclicity of the solution (Lachapelle et al., 2020; Brouillard et al., 2020; Lopez et al., 2022).

For the non-linear setting, similarly to NOTEARS (Zheng et al., 2020), we model the outcome of each variable X_u with a neural network $f_u \colon \mathbb{R}^d \to \mathbb{R}$, where we represent first-layer weights as a tensor $\mathbf{H} \in \mathbb{R}^{d \times d \times h}$. Intuitively, each entry $\mathbf{H}_{uvi}$ stands for the weight from the variable X_u to the i-th first-layer neuron in the MLP f_v. Then, the weighted adjacency matrix results from the summation on the hidden dimension. Formally, $\mathbf{W}_{uv} = \sum_{i=1}^{h} \mathbf{H}_{uvi}$.

By checking the convergence of the model, both NOTEARS, DAGMA, and NOCURL can dynamically stop the optimization procedure. On the other hand, COSMO requires a fixed number of epochs in which to anneal the temperature value. For a fair comparison, while we stop optimization problems after a maximum of 5000 training iterations, we do not disable early-stopping conditions on the baselines. Therefore, when sufficiently large, the maximum number of epochs should not affect the overall execution time of the methods. For COSMO, we interrupt the optimization after 2000 epochs. For the non-linear version of DAGMA, we increased the maximum epochs to 7000. Overall, we interrupt the execution of an algorithm whenever it hits a wall time limit of 20000 seconds.

As previously discussed in Subsection C.1, we perform a hyperparameter search on each model for each dataset. In particular, we sampled the learning rate from the range (1e-4, 1e-2) and the regularization coefficients from the interval (1e-4, 1e-1). For the specific constrained optimization parameters, such as the number of problems or decay factors, we replicated the baseline parameters, for which we point the reader to the original papers or our implementation. For COSMO, we sample hyperparameters from the ranges in Table 3, given our theoretical findings on the relation between acyclicity and temperature (Theorem 3.4), we ensure sufficiently small acyclicity values. In the non-linear variant, we employ Multilayer Perceptrons with $h = 10$ hidden units for each variable.

D. Additional Results

In this section, we report further results on simulated causal DAGs with different noise terms, graph types, and increasing numbers of nodes. For each algorithm, we present the mean and standard deviation of each metric on five independent runs. We report the Area under the ROC Curve (AUC), the True Positive Ratio (TPR), and the Structural Hamming Distance normalized by the number of nodes (NHD). The reported duration of NOCURL includes the time to retrieve the necessary preliminary solution through two optimization problems regularized with the NOTEARS acyclicity constraint. We denote as NOCURL-U the variation of NOCURL that solves a unique unconstrained optimization problem without preliminary solution. When not immediate, we highlight in bold the **best** result and in italic bold the ***second best*** result. We do not report methods exceeding our wall time limit of 20000 seconds.

D.1. ER4 - Gaussian Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	***0.867 ± 1.01***	**0.953 ± 0.04**	***0.984 ± 0.02***	**88 ± 3**
	DAGMA	**0.707 ± 0.57**	0.940 ± 0.04	**0.985 ± 0.01**	781 ± 193
	NOCURL	1.653 ± 0.17	***0.942 ± 0.02***	0.967 ± 0.01	822 ± 15
	NOCURL-U	5.623 ± 0.92	0.492 ± 0.08	0.694 ± 0.06	***227 ± 5***
	NOTEARS	0.913 ± 0.60	0.940 ± 0.05	0.973 ± 0.02	5193 ± 170
100	COSMO	***1.388 ± 0.69***	***0.917 ± 0.04***	0.961 ± 0.03	**99 ± 2**
	DAGMA	**1.026 ± 0.40**	0.876 ± 0.02	**0.982 ± 0.01**	661 ± 142
	NOCURL	5.226 ± 1.34	**0.921 ± 0.02**	0.962 ± 0.01	1664 ± 15
	NOCURL-U	10.108 ± 4.11	0.427 ± 0.05	0.682 ± 0.05	***267 ± 10***
	NOTEARS	2.380 ± 2.10	0.898 ± 0.03	***0.963 ± 0.01***	11001 ± 340
500	COSMO	***4.149 ± 1.14***	***0.819 ± 0.02***	***0.933 ± 0.01***	**437 ± 81**
	DAGMA	**2.246 ± 0.40**	**0.882 ± 0.01**	**0.980 ± 0.00**	2485 ± 366
	NOCURL-U	27.675 ± 16.52	0.410 ± 0.04	0.683 ± 0.05	***1546 ± 304***

D.2. ER4 - Exponential Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	**0.600 ± 0.54**	**0.970 ± 0.02**	**0.989 ± 0.01**	**89 ± 3**
	DAGMA	***0.613 ± 0.91***	***0.958 ± 0.05***	***0.986 ± 0.02***	744 ± 75
	NOCURL	2.300 ± 0.97	0.918 ± 0.04	0.956 ± 0.02	826 ± 24
	NOCURL-U	5.313 ± 0.17	0.423 ± 0.05	0.694 ± 0.05	***212 ± 5***
	NOTEARS	1.320 ± 0.67	0.880 ± 0.10	0.966 ± 0.03	5579 ± 284
100	COSMO	1.642 ± 0.26	**0.952 ± 0.02**	***0.985 ± 0.01***	**99 ± 2**
	DAGMA	***1.294 ± 0.52***	***0.944 ± 0.02***	**0.986 ± 0.01**	733 ± 109
	NOCURL	5.652 ± 1.35	0.854 ± 0.03	0.950 ± 0.02	1655 ± 28
	NOCURL-U	11.642 ± 4.34	0.478 ± 0.05	0.693 ± 0.05	***242 ± 4***
	NOTEARS	**1.156 ± 0.44**	0.904 ± 0.03	0.972 ± 0.01	10880 ± 366
500	COSMO	***2.342 ± 0.86***	**0.944 ± 0.02**	**0.986 ± 0.00**	**390 ± 102**
	DAGMA	**2.147 ± 1.08**	***0.902 ± 0.04***	***0.984 ± 0.01***	2575 ± 469
	NOCURL-U	20.183 ± 7.43	0.437 ± 0.03	0.715 ± 0.03	***1488 ± 249***

D.3. ER4 - Gumbel Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	2.220 ± 1.65	0.862 ± 0.14	0.914 ± 0.10	**87 ± 2**
	DAGMA	***1.680 ± 0.73***	***0.937 ± 0.03***	***0.973 ± 0.02***	787 ± 86
	NOCURL	3.873 ± 1.26	0.853 ± 0.08	0.915 ± 0.04	826 ± 17
	NOCURL-U	5.260 ± 0.57	0.475 ± 0.08	0.678 ± 0.05	***212 ± 5***
	NOTEARS	**0.587 ± 0.38**	**0.962 ± 0.03**	**0.981 ± 0.01**	5229 ± 338
100	COSMO	2.398 ± 0.70	**0.936 ± 0.02**	***0.973 ± 0.01***	**98 ± 1**
	DAGMA	**1.132 ± 0.79**	***0.921 ± 0.04***	**0.986 ± 0.01**	858 ± 101
	NOCURL	4.714 ± 1.77	0.905 ± 0.03	0.962 ± 0.01	1675 ± 34
	NOCURL-U	6.914 ± 0.80	0.383 ± 0.04	0.663 ± 0.04	***247 ± 9***
	NOTEARS	***1.402 ± 0.40***	0.869 ± 0.04	0.969 ± 0.00	11889 ± 343
500	COSMO	***3.574 ± 1.44***	**0.932 ± 0.02**	**0.982 ± 0.01**	**410 ± 106**
	DAGMA	**1.737 ± 0.64**	***0.871 ± 0.03***	***0.980 ± 0.00***	2853 ± 218
	NOCURL-U	18.182 ± 9.28	0.462 ± 0.06	0.728 ± 0.05	***1342 ± 209***

D.4. SF4 - Gaussian Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	**0.300 ± 0.09**	***0.973 ± 0.01***	**0.997 ± 0.00**	**89 ± 5**
	DAGMA	***0.360 ± 0.30***	**0.973 ± 0.02**	***0.996 ± 0.01***	653 ± 198
	NOCURL	0.967 ± 0.43	0.893 ± 0.03	0.983 ± 0.01	828 ± 23
	NOCURL-U	4.410 ± 0.72	0.566 ± 0.11	0.741 ± 0.08	***226 ± 7***
	NOTEARS	0.553 ± 0.54	0.944 ± 0.06	0.984 ± 0.02	5292 ± 261
100	COSMO	***0.482 ± 0.31***	***0.962 ± 0.02***	0.991 ± 0.01	**99 ± 3**
	DAGMA	0.712 ± 0.33	0.951 ± 0.02	**0.995 ± 0.00**	479 ± 75
	NOCURL	2.030 ± 0.46	0.883 ± 0.03	0.982 ± 0.01	1667 ± 25
	NOCURL-U	5.521 ± 0.61	0.596 ± 0.09	0.788 ± 0.06	***269 ± 9***
	NOTEARS	**0.280 ± 0.35**	**0.972 ± 0.04**	***0.993 ± 0.01***	10112 ± 492
500	COSMO	***1.566 ± 0.68***	**0.953 ± 0.02**	***0.989 ± 0.01***	**541 ± 15**
	DAGMA	**1.343 ± 0.46**	***0.915 ± 0.04***	**0.992 ± 0.00**	***1345 ± 33***
	NOCURL-U	7.146 ± 3.19	0.504 ± 0.08	0.780 ± 0.07	1394 ± 217

D.5. SF4 - Exponential Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	0.613 ± 0.39	***0.965 ± 0.02***	0.985 ± 0.02	**87 ± 2**
	DAGMA	**0.127 ± 0.20**	**0.991 ± 0.01**	**0.999 ± 0.00**	592 ± 200
	NOCURL	0.887 ± 0.21	0.845 ± 0.02	***0.985 ± 0.01***	824 ± 25
	NOCURL-U	4.067 ± 0.73	0.460 ± 0.15	0.685 ± 0.09	***212 ± 7***
	NOTEARS	***0.513 ± 0.30***	0.962 ± 0.03	0.984 ± 0.01	5189 ± 271
100	COSMO	***0.724 ± 0.71***	***0.963 ± 0.04***	0.985 ± 0.02	**100 ± 2**
	DAGMA	**0.586 ± 0.56**	**0.969 ± 0.03**	**0.995 ± 0.00**	395 ± 108
	NOCURL	1.998 ± 0.40	0.907 ± 0.03	0.980 ± 0.00	1670 ± 28
	NOCURL-U	5.912 ± 1.54	0.575 ± 0.06	0.783 ± 0.04	***245 ± 7***
	NOTEARS	0.910 ± 0.43	0.962 ± 0.02	***0.991 ± 0.01***	10243 ± 723
500	COSMO	**1.445 ± 0.58**	**0.950 ± 0.03**	**0.990 ± 0.01**	**517 ± 108**
	DAGMA	***1.653 ± 0.91***	***0.873 ± 0.08***	***0.988 ± 0.01***	1466 ± 247
	NOCURL-U	12.140 ± 7.84	0.482 ± 0.08	0.727 ± 0.06	***1205 ± 257***

D.6. SF4 - Gumbel Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	**0.467 ± 0.51**	**0.962 ± 0.05**	***0.990 ± 0.02***	**88 ± 2**
	DAGMA	***0.487 ± 0.20***	***0.956 ± 0.03***	**0.990 ± 0.01**	754 ± 179
	NOCURL	0.747 ± 0.19	0.938 ± 0.02	0.989 ± 0.00	826 ± 32
	NOCURL-U	3.107 ± 0.64	0.460 ± 0.06	0.737 ± 0.04	***213 ± 5***
	NOTEARS	0.860 ± 0.76	0.924 ± 0.06	0.975 ± 0.02	5199 ± 130
100	COSMO	***0.864 ± 0.24***	***0.968 ± 0.01***	***0.992 ± 0.01***	**98 ± 2**
	DAGMA	**0.388 ± 0.30**	**0.975 ± 0.02**	**0.997 ± 0.00**	422 ± 103
	NOCURL	1.806 ± 0.40	0.898 ± 0.03	0.982 ± 0.01	1676 ± 31
	NOCURL-U	8.756 ± 2.65	0.550 ± 0.05	0.757 ± 0.03	***245 ± 7***
	NOTEARS	1.134 ± 0.81	0.894 ± 0.08	0.989 ± 0.01	11618 ± 1309
500	COSMO	***1.426 ± 0.53***	**0.951 ± 0.03**	**0.994 ± 0.00**	**524 ± 22**
	DAGMA	**1.384 ± 0.38**	***0.849 ± 0.04***	***0.991 ± 0.00***	1359 ± 34
	NOCURL-U	8.931 ± 7.05	0.430 ± 0.10	0.741 ± 0.08	***1193 ± 229***

D.7. ER6 - Gaussian Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	4.087 ± 1.12	0.838 ± 0.06	0.921 ± 0.04	**89 ± 4**
	DAGMA	**2.367 ± 0.63**	***0.847 ± 0.03***	**0.958 ± 0.01**	665 ± 249
	NOCURL	4.480 ± 0.92	**0.869 ± 0.03**	0.908 ± 0.03	909 ± 18
	NOCURL-U	7.490 ± 1.18	0.459 ± 0.08	0.672 ± 0.06	***226 ± 6***
	NOTEARS	***3.327 ± 1.65***	0.840 ± 0.07	***0.922 ± 0.04***	5239 ± 427
100	COSMO	***9.476 ± 3.01***	0.771 ± 0.08	***0.911 ± 0.05***	**98 ± 2**
	DAGMA	10.740 ± 2.83	0.709 ± 0.13	0.902 ± 0.04	761 ± 134
	NOCURL	15.044 ± 1.60	***0.785 ± 0.04***	0.888 ± 0.02	1687 ± 26
	NOCURL-U	30.719 ± 5.20	0.435 ± 0.03	0.580 ± 0.04	***268 ± 9***
	NOTEARS	**6.556 ± 3.10**	**0.842 ± 0.05**	**0.944 ± 0.02**	12053 ± 940
500	COSMO	***25.443 ± 4.47***	**0.736 ± 0.01**	**0.937 ± 0.01**	**526 ± 100**
	DAGMA	**15.952 ± 1.67**	***0.553 ± 0.05***	***0.925 ± 0.01***	3207 ± 271
	NOCURL-U	165.465 ± 20.86	0.433 ± 0.02	0.558 ± 0.03	***1226 ± 293***

D.8. ER6 - Exponential Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	***3.300 ± 0.95***	**0.897 ± 0.05**	***0.947 ± 0.03***	**89 ± 2**
	DAGMA	3.480 ± 1.42	0.861 ± 0.06	0.945 ± 0.03	672 ± 177
	NOCURL	4.573 ± 0.78	0.846 ± 0.05	0.902 ± 0.03	897 ± 13
	NOCURL-U	8.700 ± 0.89	0.426 ± 0.07	0.615 ± 0.06	***226 ± 9***
	NOTEARS	**2.313 ± 1.55**	***0.881 ± 0.09***	**0.953 ± 0.04**	5516 ± 652
100	COSMO	10.170 ± 2.74	0.768 ± 0.09	0.919 ± 0.04	**99 ± 3**
	DAGMA	***8.118 ± 3.10***	***0.793 ± 0.11***	***0.934 ± 0.04***	681 ± 149
	NOCURL	14.860 ± 4.67	0.685 ± 0.10	0.863 ± 0.06	1735 ± 39
	NOCURL-U	30.600 ± 4.34	0.450 ± 0.04	0.591 ± 0.04	***267 ± 8***
	NOTEARS	**5.208 ± 2.54**	**0.796 ± 0.09**	**0.943 ± 0.03**	12663 ± 1555
500	COSMO	***25.854 ± 4.28***	**0.741 ± 0.04**	**0.943 ± 0.01**	**460 ± 123**
	DAGMA	**16.417 ± 4.45**	***0.571 ± 0.11***	***0.925 ± 0.02***	4069 ± 580
	NOCURL-U	152.336 ± 31.97	0.425 ± 0.02	0.567 ± 0.03	***1363 ± 306***

D.9. ER6 - Gumbel Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	2.840 ± 1.08	**0.906 ± 0.04**	***0.954 ± 0.03***	**89 ± 3**
	DAGMA	**2.727 ± 0.83**	***0.906 ± 0.02***	**0.964 ± 0.02**	634 ± 194
	NOCURL	5.003 ± 0.72	0.811 ± 0.04	0.891 ± 0.03	902 ± 9
	NOCURL-U	8.153 ± 0.96	0.422 ± 0.07	0.629 ± 0.04	***226 ± 6***
	NOTEARS	***2.740 ± 1.61***	0.791 ± 0.10	0.938 ± 0.04	5416 ± 446
100	COSMO	10.048 ± 3.15	0.780 ± 0.07	0.899 ± 0.06	**100 ± 3**
	DAGMA	***7.910 ± 3.05***	***0.805 ± 0.09***	***0.935 ± 0.04***	715 ± 203
	NOCURL	11.932 ± 2.68	0.742 ± 0.04	0.894 ± 0.03	1688 ± 34
	NOCURL-U	27.401 ± 4.42	0.431 ± 0.05	0.600 ± 0.04	***266 ± 4***
	NOTEARS	**4.884 ± 0.45**	**0.833 ± 0.05**	**0.951 ± 0.01**	12634 ± 639
500	COSMO	***26.148 ± 4.86***	**0.740 ± 0.04**	**0.941 ± 0.02**	**418 ± 106**
	DAGMA	**16.358 ± 4.94**	***0.563 ± 0.07***	***0.921 ± 0.02***	3527 ± 241
	NOCURL-U	125.858 ± 36.61	0.367 ± 0.06	0.571 ± 0.02	***1612 ± 27***

D.10. SF6 - Gaussian Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	1.273 ± 1.07	0.907 ± 0.10	0.963 ± 0.06	**89 ± 2**
	DAGMA	***1.107 ± 0.37***	**0.930 ± 0.03**	**0.985 ± 0.01**	456 ± 39
	NOCURL	1.573 ± 0.46	0.864 ± 0.04	0.973 ± 0.01	823 ± 14
	NOCURL-U	4.997 ± 0.98	0.506 ± 0.05	0.732 ± 0.05	***226 ± 8***
	NOTEARS	**0.933 ± 0.71**	***0.919 ± 0.05***	***0.984 ± 0.02***	5313 ± 184
100	COSMO	4.478 ± 2.88	0.776 ± 0.15	0.874 ± 0.11	**99 ± 2**
	DAGMA	***2.024 ± 0.71***	***0.914 ± 0.02***	***0.987 ± 0.00***	396 ± 53
	NOCURL	2.824 ± 0.39	0.818 ± 0.02	0.980 ± 0.00	1679 ± 27
	NOCURL-U	10.556 ± 6.00	0.542 ± 0.07	0.751 ± 0.08	***266 ± 5***
	NOTEARS	**1.412 ± 0.59**	**0.939 ± 0.03**	**0.990 ± 0.01**	11156 ± 170
500	COSMO	***4.670 ± 1.99***	**0.912 ± 0.02**	**0.984 ± 0.00**	**460 ± 70**
	DAGMA	**3.825 ± 0.19**	***0.746 ± 0.03***	***0.982 ± 0.00***	1418 ± 54
	NOCURL-U	19.793 ± 11.03	0.368 ± 0.04	0.698 ± 0.04	***1137 ± 231***

D.11. SF6 - Exponential Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	1.393 ± 1.24	0.926 ± 0.05	0.975 ± 0.03	**88 ± 1**
	DAGMA	***1.147 ± 0.48***	***0.943 ± 0.03***	***0.982 ± 0.01***	578 ± 173
	NOCURL	1.987 ± 0.54	0.757 ± 0.08	0.967 ± 0.01	820 ± 8
	NOCURL-U	4.787 ± 0.99	0.534 ± 0.07	0.761 ± 0.06	***227 ± 7***
	NOTEARS	**0.753 ± 0.49**	**0.943 ± 0.04**	**0.986 ± 0.01**	5312 ± 258
100	COSMO	3.836 ± 2.75	0.864 ± 0.09	0.944 ± 0.05	**98 ± 2**
	DAGMA	**1.532 ± 0.61**	0.887 ± 0.04	**0.988 ± 0.00**	373 ± 88
	NOCURL	2.890 ± 0.61	***0.910 ± 0.02***	0.977 ± 0.00	1692 ± 28
	NOCURL-U	6.607 ± 1.05	0.474 ± 0.06	0.760 ± 0.06	***266 ± 2***
	NOTEARS	***1.784 ± 0.52***	**0.939 ± 0.02**	***0.988 ± 0.00***	11369 ± 519
500	COSMO	**3.144 ± 0.47**	**0.919 ± 0.02**	**0.989 ± 0.00**	**457 ± 81**
	DAGMA	***3.854 ± 0.34***	***0.750 ± 0.01***	***0.977 ± 0.01***	***1384 ± 33***
	NOCURL-U	13.763 ± 8.79	0.389 ± 0.05	0.728 ± 0.06	1436 ± 230

D.12. SF6 - Gumbel Noise

d	Algorithm	NHD	TPR	AUC	Time (s)
30	COSMO	**1.047 ± 0.42**	**0.938 ± 0.03**	**0.984 ± 0.01**	**88 ± 1**
	DAGMA	1.347 ± 0.63	***0.933 ± 0.02***	***0.981 ± 0.01***	528 ± 67
	NOCURL	1.787 ± 0.52	0.898 ± 0.02	0.969 ± 0.01	822 ± 29
	NOCURL-U	5.577 ± 0.43	0.549 ± 0.06	0.733 ± 0.05	***225 ± 4***
	NOTEARS	***1.053 ± 0.59***	0.911 ± 0.04	0.978 ± 0.02	5429 ± 251
100	COSMO	3.486 ± 2.62	0.879 ± 0.10	0.947 ± 0.06	**99 ± 2**
	DAGMA	**1.418 ± 0.34**	***0.910 ± 0.03***	**0.990 ± 0.00**	424 ± 90
	NOCURL	3.074 ± 0.50	0.893 ± 0.02	0.976 ± 0.00	1682 ± 22
	NOCURL-U	9.643 ± 4.59	0.464 ± 0.08	0.712 ± 0.10	***267 ± 9***
	NOTEARS	***1.586 ± 1.39***	**0.913 ± 0.06**	***0.987 ± 0.01***	11820 ± 985
500	COSMO	**3.288 ± 0.50**	**0.931 ± 0.01**	**0.992 ± 0.00**	**429 ± 87**
	DAGMA	***4.055 ± 0.88***	***0.802 ± 0.03***	***0.981 ± 0.00***	1465 ± 138
	NOCURL-U	56.103 ± 41.06	0.420 ± 0.06	0.648 ± 0.07	***1201 ± 253***

D.13. ER4 - Non-linear MLP

d	Algorithm	NHD	TPR	AUC	Time (s)
20	COSMO	2.110 ± 0.37	0.726 ± 0.05	0.939 ± 0.02	125 ± 4
	DAGMA	3.540 ± 0.45	0.323 ± 0.06	0.740 ± 0.05	1834 ± 34
40	COSMO	2.420 ± 0.48	0.668 ± 0.04	0.930 ± 0.01	139 ± 3
	DAGMA	3.902 ± 0.07	0.024 ± 0.02	0.767 ± 0.03	2054 ± 45
100	COSMO	4.864 ± 0.95	0.627 ± 0.02	0.918 ± 0.01	228 ± 6
	DAGMA	3.609 ± 0.10	0.244 ± 0.02	0.848 ± 0.02	3537 ± 32

D.14. SF4 - Non-linear MLP

d	Algorithm	NHD	TPR	AUC	Time (s)
20	COSMO	1.725 ± 0.34	0.701 ± 0.06	0.949 ± 0.02	125 ± 5
	DAGMA	3.290 ± 0.15	0.069 ± 0.04	0.604 ± 0.11	1838 ± 48
40	COSMO	2.065 ± 0.18	0.573 ± 0.05	0.960 ± 0.01	139 ± 4
	DAGMA	3.750 ± 0.00	0.000 ± 0.00	0.706 ± 0.09	2069 ± 26
100	COSMO	3.019 ± 0.14	0.374 ± 0.04	0.945 ± 0.01	229 ± 8
	DAGMA	3.835 ± 0.04	0.017 ± 0.01	0.721 ± 0.05	3551 ± 34

D.15. ER6 - Non-linear MLP

d	Algorithm	NHD	TPR	AUC	Time (s)
20	COSMO	2.945 ± 0.23	0.666 ± 0.07	0.938 ± 0.01	124 ± 5
	DAGMA	5.430 ± 0.24	0.132 ± 0.04	0.607 ± 0.07	1826 ± 45
40	COSMO	4.345 ± 0.64	0.594 ± 0.06	0.907 ± 0.01	138 ± 5
	DAGMA	6.000 ± 0.00	0.000 ± 0.00	0.648 ± 0.04	2056 ± 40
100	COSMO	4.390 ± 0.31	0.421 ± 0.02	0.900 ± 0.01	228 ± 6
	DAGMA	6.355 ± 0.22	0.161 ± 0.02	0.806 ± 0.01	3534 ± 29

D.16. SF6 - Non-linear MLP

d	Algorithm	NHD	TPR	AUC	Time (s)
20	COSMO	2.110 ± 0.29	0.673 ± 0.08	0.965 ± 0.01	124 ± 5
	DAGMA	4.915 ± 0.05	0.007 ± 0.01	0.610 ± 0.13	1846 ± 34
40	COSMO	3.255 ± 0.32	0.541 ± 0.04	0.950 ± 0.01	138 ± 2
	DAGMA	5.460 ± 0.02	0.003 ± 0.00	0.590 ± 0.08	2056 ± 42
100	COSMO	4.625 ± 0.20	0.305 ± 0.04	0.936 ± 0.01	229 ± 6
	DAGMA	5.654 ± 0.05	0.023 ± 0.01	0.667 ± 0.07	3480 ± 60