Triangular Monotonic Generative Models Can Perform Causal Discovery

Quanhan Xi
University of British Columbia
johnny.xi@stat.ubc.ca

Sebastian Gonzalez
University of British Columbia
bastigonzalez2000@gmail.com

Benjamin Bloem-Reddy
University of British Columbia
benbr@stat.ubc.ca

Abstract

Many causal discovery algorithms exploit conditional independence signatures in observational data, recovering a Markov equivalence class (MEC) of possible DAGs consistent with the data. In case the MEC is non-trivial, additional assumptions on the data generating process can be made, and generative models can be fit to further resolve the MEC. We show that triangular monotonic increasing (TMI) maps parametrize generative models that perform conditional independence-based causal discovery by searching over permutations, that additionally are flexible enough as generative models to fit a wide class of causal models. In this paper, we characterize the theoretical properties that make these models relevant as tools for causal discovery, make connections to existing methods, and highlight open challenges towards their deployment.

1 Introduction

Causal discovery aims to use data—possibly observational—to uncover causal relationships assumed to be encoded as a directed acyclic graph (DAG). Traditionally, this is either done by testing for conditional independence between observed variables, or optimizing some appropriately defined data-driven scoring function over the space of DAGs, such as the data likelihood (Spirtes et al., 2000; Chickering, 2002; Kalisch and Bühlman, 2007; Raskutti and Uhler, 2018; Solus et al., 2021). These methods typically are able to identify causal structure only up to Markov equivalence—the set of DAGs that encode the same observed conditional independencies. Notably, this excludes the possibility of discovery in the two-variable setting, where the DAGs $X \to Y$ and $Y \to X$ are Markov equivalent.

Markov equivalence might be resolved via interventional data (Hauser and Bühlmann, 2012; Yang et al., 2018; Squires et al., 2020) or by making certain causal generative modelling assumptions that break the symmetry of conditional independence (Shimizu et al., 2006; Zhang and Hyvärinen, 2009; Peters et al., 2014). The latter can be combined with conditional independence based methods to first identify a Markov equivalence class (MEC), and then to identify the DAG within the MEC by scoring with the generative model (Monti et al., 2020; Khemakhem et al., 2021). Here, we propose using generative models parametrized by triangular monotonic increasing (TMI) maps, and show that they perform conditional independence based causal discovery while also learning the causal generative model, as well as automatically satisfying DAG constraints. In this way, we can use the same model to identify the MEC, use the underlying causal generative model to resolve indeterminacies within the MEC, and finally perform causal inference.

Accepted to the NeurIPS 2023 Workshop on Causal Representation Learning.

Triangular monotonic increasing (TMI) mappings from $\mathbb{R}^d \to \mathbb{R}^d$ are natural candidates for causal generative modelling. Mathematically, a TMI map is a function $T\colon \mathbb{R}^d \to \mathbb{R}^d$ such that

$$T(x) = \begin{bmatrix} T_1(x_1) \\ T_2(x_1, x_2) \\ \vdots \\ T_d(x_1, \ldots, x_d) \end{bmatrix}, \tag{1}$$

where each map $x_k \mapsto T_k(x_{1:k-1}, x_k)$ is monotone increasing (hence invertible), for each $x_{1:k-1}$.

When the causal ordering is known, TMI maps define a flexible, yet identifiable (Xi and Bloem-Reddy, 2023) generative model that contains a wide class of non-linear structural causal models (SCM), enabling causal inference (Javaloy et al., 2023). However, when the causal ordering is unknown, it is less clear whether TMI maps can also be used for causal discovery.

Our contribution is to show that TMI maps can also be used to perform conditional independence based causal discovery by finding the maximally sparse permutation (Raskutti and Uhler, 2018). Here, sparsity refers to the sparsity of the Jacobian of the learned generative model, which we equivalently refer to as simply the sparsity of the model, a commonly used notion for causal discovery in non-linear models (Zheng et al., 2020; Lachapelle et al., 2020; Reizinger et al., 2022).

Given two distributions P_X and P_ϵ, the TMI map that transports between them as $T_* P_X = P_\epsilon$ is unique up to a permutation of its inputs. We denote a permuted TMI map by T^π, for a permutation π of $[d] = \{1, 2, \ldots, d\}$. Our key observation is that the permutation that yields maximal sparsity of the Jacobian of a learned T, J_T, identifies the MEC of the underlying DAG.

Theorem 1.1. *Let P_X be an observed distribution over $\mathbb{R}^d$ faithful to a DAG $\mathcal{G}$, and P_ϵ be a fixed base distribution that is independent over $\mathbb{R}^d$. Let $(T^\pi)_{\pi \in \mathbb{S}_d}$ index the set of unique TMI maps such that $T^\pi_* P_X = P_\epsilon$. Then, the permutation π^* such that $J_{T^{\pi^*}}(x)$ is maximally sparse, is such that the Jacobian, seen as an adjacency matrix, is Markov equivalent to $\mathcal{G}_{\pi^*}$.*

This result actually holds under a weaker condition known as restricted faithfulness, as shown in Theorem 2.1 c) of Raskutti and Uhler (2018), which shows that the permutation yielding the sparsest graph from an appropriate sequence of conditional independence tests identifies the Markov equivalence class. The equivalence here is that TMI maps test precisely the same conditional independencies via the sparsity of the Jacobian. Raskutti and Uhler (2018) however do not study the properties of permutations that are not maximally sparse. Under the stronger assumption of strict faithfulness, we are able to give a constructive result using ideas from variable elimination in undirected graphs that further identifies the additional edges induced by permutations that are not maximally sparse (Theorem 3.1).

1.1 Related Works

Jacobian and score-based Causal Discovery Our contribution in this work is complementary to various recent advancements using the Jacobian of a generator function, or in score matching (Zheng et al., 2020; Lachapelle et al., 2020; Rolland et al., 2022; Montagna et al., 2023a,b). The main novelty of using TMI maps to parametrize generative models is that the resulting causal discovery method is *model-free*, as it does not depend on the assumption that observations are generated by a specific underlying structural model. Instead, TMI maps model the conditional independencies directly due to their properties as probability transport maps. In particular, the result for structure learning (Theorem 1.1) does not require the observations to be generated by an SCM at all. However, without assuming an identifiable SCM, the structure can fundamentally only be identified up to a MEC. As such, we do still assume an underlying SCM to refine the MEC. We attempt to do so minimally in Theorem 3.2, requiring only that the observations are generated from an SCM that monotonically depends on the noise variables, which is a strict generalization over additive noise models, by using ICA-based causal discovery to orient edges (Monti et al., 2020). This is particularly appealing as TMI generative models perform ICA-style latent variable inference simply via a forward pass of observations through the trained model, and does not require knowledge of the noise mechanism. However, this requires the sparsest permutation search to first be solved, for example via greedy methods (Solus et al., 2021), which is significantly more difficult in a generative model (see Section 4). This is in contrast to stronger assumptions such as additive noise, which allow methods such

as (Rolland et al., 2022; Montagna et al., 2023a,b) to reduce the permutation search, which scales combinatorially, to a recursive search of leaf nodes, which scales linearly.

TMIs as Conditional Independence Tests The connection between sparsity in TMI maps and conditional independence is well known and has been successfully applied for learning undirected graphical models (Morrison et al., 2017; Spantini et al., 2018; Baptista et al., 2021). Recently, these results were leveraged by (Akbari et al., 2023) in the context of causal discovery. The SING algorithm of Morrison et al. (2017) uses a second-order Hessian condition on the TMI map to deduce conditional independence. Interestingly, SING also attempts to find the sparsest permutation for the TMI map, optimizing the same criteria as in Theorem 1.1, but only to improve sample efficiency as the undirected graph has no order structure. Akbari et al. (2023) then propose to use SING as the conditional independence test within the PC algorithm (Spirtes et al., 2000) to recover the MEC. Our proposal can be seen as using the sparsest permutation directly to learn the MEC, without the need of the PC algorithm. In particular, Akbari et al. (2023) requires explicitly re-learning sparsest TMI maps in each SING subroutine over different subsets at each iteration of the PC algorithm, while our proposed method can be seen as a single iteration of SING. Then, to break the MEC, Akbari et al. (2023) propose a criteria based on additive noise, which requires then learning the inverse TMI map separately. Our proposed approach, via Theorem 3.2, only requires a conditional independence test on the latent variables after a forward pass through the learned TMI map.

Overall, TMI maps have many attractive properties for causal discovery, many of which have already been exploited, but not precisely as TMI generative models. In view of recent interest in using generative models for causality, the connection to existing causal discovery methods is to the best of our knowledge novel, and allows us to leverage theoretical backing from a rich literature. These connections cover a wide range of the existing literature: traditional conditional independence-based methods (Theorem 1.1), learning undirected graphical models (our proof of Theorem 3.1), and nonlinear-ICA based methods for causal discovery (Theorem 3.2).

2 Triangular Monotonic Maps and the Knöthe–Rosenblatt Transport

TMI maps enjoy several appealing functional and probabilistic properties. A triangular Jacobian makes it easy to evaluate the Jacobian determinant, and monotonicity ensures invertibility; these properties make them particularly appealing as layers for normalizing flows (Jaini et al., 2019; Papamakarios et al., 2021), though there the order is often alternated and thus the final map is not necessarily triangular. The most appealing fact in our context is that TMI maps naturally generalize the 1-dimensional CDF transform. It is well-known that if $P_\epsilon = T_* P_X$, where P_X, P_ϵ have strictly positive density and T is a TMI map, then T is equivalent to the Knöthe–Rosenblatt (KR) transport almost everywhere (see Jaini et al. (2019, Theorem 1), or Bogachev et al. (2005)). Specializing to the case where $P_\epsilon = \mathcal{U}[0,1]^d$, the KR transport is described recursively as $F_{X_k|X_{1:k-1}}(x_k|x_{1:k-1})$, the conditional CDF of the k-th component of $X_k|X_{1:k-1}$. Because P_X has strictly positive density, so do each of its conditional distributions, and therefore $F_{X_k|X_{1:k-1}}$ is monotone increasing. Note the first component is simply the CDF transform:

$$T_1(x_1) = F_{X_1}(x_1),$$

where it is well known that the CDF is the unique monotone increasing transformation from X_1 to the uniform. In each subsequent dimension, the CDF is again the unique monotone increasing transformations between the 1-dimensional unique (almost everywhere) regular conditional probabilities (Bogachev et al., 2005; Jaini et al., 2019).

2.1 Graphical Models and Conditional Independence

A directed acyclic graph (DAG) with d vertices is a tuple $\vec{\mathcal{G}} = ([d], E)$ with nodes identified to their indices $[d]$, and edges $E \in [d] \times [d]$, where $(i,j) \in E$ denotes $i \to j$ in the graph. In $\mathcal{G}$, a node i is said to be a parent of j if the edge $i \to j$ exists, and similarly j is said to be a child of i. The set of parents of j are denoted $Pa(j, \vec{\mathcal{G}})$, and the children $Ch(j, \vec{\mathcal{G}})$. The ancestors of j, $An(j, \vec{\mathcal{G}})$ denote all nodes with directed paths into j, and including j. When the DAG under consideration is clear, we drop $\vec{\mathcal{G}}$ from the notation. In all cases, parents, children and ancestors of sets of nodes are defined as the union of the constituent nodes. All DAGs admit a topological ordering π_t (though it is not necessarily unique), where $\pi_t(i) \in Pa(\pi_t(j)) \implies \pi_t(i) < \pi_t(j)$. A particularly important

sub-graph is the v-structure, $i_1 \to i_3 \leftarrow i_2$, where crucially there is no edge between i_1 and i_2. Finally, the skeleton of $\vec{\mathcal{G}}$ is defined as the undirected graph obtained by removing direction on all edges.

P_X is said to satisfy the global Markov property w.r.t. $\vec{\mathcal{G}}$ if:

$$I_1 \perp\!\!\!\perp_{\vec{\mathcal{G}}} I_2 \mid I_3 \implies X_{I_1} \perp\!\!\!\perp X_{I_2} \mid X_{I_3}, \tag{2}$$

for all subsets $I_1, I_2, I_3 \subseteq [d]$, where $\perp\!\!\!\perp_{\vec{\mathcal{G}}}$ denotes d-separation (Pearl, 2009, Def. 1.2.3). In words, all d-separations in $\vec{\mathcal{G}}$ correspond to CI statements in P_X. In general, P_X is permitted to satisfy additional CI statements that are not encoded by the graph. Even with perfect knowledge of the CI statements from P_X, a graph that encodes all such statements (if it even exists) may not correspond to the true DAG. To avoid such cases, we will make the following assumption on any P_X under consideration, known as faithfulness:

$$I_1 \perp\!\!\!\perp_{\vec{\mathcal{G}}} I_2 \mid I_3 \iff X_{I_1} \perp\!\!\!\perp X_{I_2} \mid X_{I_3}, \tag{3}$$

for $I_1, I_2, I_3 \subseteq [d]$. Faithfulness is a strong assumption, see Uhler et al. (2013); Sadeghi (2017) for discussions. We assume faithfulness for simplicity in proving our Theorem 3.1, but note that, due to the equivalence of our method to sparsest-permutation based methods, Theorem 1.1 holds under a weaker condition known as SMR (Raskutti and Uhler, 2018).

Before proceeding, we define some important concepts in undirected graphs, which we denote $\mathcal{G} = ([d], E)$. To distinguish from the directed case, we will use the set notation $\{i, j\} \in E$ to denote an edge, indicating that the order is irrelevant. The neighbourhood of a node i in $\mathcal{G}$ is defined as $Nb(j, \mathcal{G}) = \{i \in [d] \mid \{i, j\} \in E\}$. A connected component of $\mathcal{G}$ is any subgraph such that all nodes within it are reachable via paths to each other. We will identify connected components to their nodes, i.e., $C \subseteq [d]$ denotes a connected component. Next, we will define the outer boundary of a subset $I \subseteq [d]$ to be the following:

$$OBd(I, \mathcal{G}) = \{i \in [d] \mid i \notin I, \{i, j\} \in E \text{ for some } j \in I)\}. \tag{4}$$

In words, the outer boundary contains nodes that are neighbours of nodes in I, but not within I themselves.

3 TMI Models and Causal Discovery

When looking for conditional independence, we will be interested in the components T_k that do not depend on all its possible arguments. In other words, we are interested in the sparsity pattern of the Jacobian of T, J_T. The sparsity pattern describes specific conditional independence (CI) statements, since

$$F_{X|Y,Z} = F_{X|Y} \iff X \perp\!\!\!\perp Z \mid Y. \tag{5}$$

In general, we do not require $P_\epsilon = \mathcal{U}[0,1]^d$, merely that it is independent, in which case the TMI map differs from the CDF by an univariate monotonic transformation. To reduce the risk of misspecification in practice, the generative model can be parametrized as ICA via (T, P_ϵ), where T is a TMI map and P_ϵ is fully supported with independent components. Such a model remains identifiable to the extent that any indeterminacy does not affect any conditional independencies, and hence their causal implications (Xi and Bloem-Reddy, 2023; Javaloy et al., 2023). For the sake of simplicity however, we continue to view TMI maps as conditional CDFs. The Jacobian J_T must be lower-triangular and $J_T(x)_{k,k} > 0$ for each k representing the (strictly positive) conditional density. There are $n(n-1)/2$ remaining entries corresponding to $J_T(x)_{j,k}$, $j < k$, the sparsity of which describe the following CI statements:

$$X_k \perp\!\!\!\perp X_j \mid \{X_1, \ldots, X_{k-1}\} \setminus \{X_j\}, \quad \text{for each } j < k. \tag{6}$$

The above sequence of CI statements is precisely those that are tested for a given permutation in permutation-based causal discovery.

Assuming faithfulness, we can interpret the CI statements tested by a TMI model equivalently as d-separation statements in a DAG $\vec{\mathcal{G}}$. If a topological order is known, the TMI model recovers the graph exactly, since the conditional dependencies correspond precisely to the parents of node k. The

sparsity pattern is highly sensitive to the order of the variables however. Denote a permutation of variables by $\pi(i)$, where π is a bijection from $[d]$ to itself. We denote $\vec{\mathcal{G}}_\pi$ as $\vec{\mathcal{G}}$ with nodes relabelled according to π. Note this relablling is arbitrary, and does not change the semantic meaning of the DAG. Thus we consider recovering $\vec{\mathcal{G}}_\pi$ to be equivalent to recovering the true DAG.

In what follows, we use $[d]$ to denote an arbitrary ordering, it is not necessarily the topological order of $\vec{\mathcal{G}}$ (e.g., we might have $i \to j$ with $i > j$). The following result determines the graph recovered by fitting the TMI map (in the ordering $[d]$), which may not be maximally sparse, and hence lets us identify the erroneously added edges when the learned DAG is not in the MEC.

Theorem 3.1. *Let P_X be faithful to a DAG $\vec{\mathcal{G}} = ([d], E)$. The TMI map T with $T_* P_X = P_\epsilon$, has non-zero Jacobian entries as follows. For each k, $J_T(x)_{k,l} \neq 0$ if and only if l is in the following set:*

$$\big[(Ch(k) \cup Pa(k)) \cap [k]\big] \cup (Pa(Ch(k) \cap A_k) \cap [k]) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_{M,A_k}), \tag{7}$$

where $A_k = An([k])$, $\mathcal{G}_{M,A_k}$ is the moral graph of $\vec{\mathcal{G}}|_{A_k}$, and $E_{k,C} \in \mathcal{C}_k$ denotes the connected components of the nodes $E_k := \{k+1, \dots, d\} \cap A_k$ such that the node k is in their outer boundary.

The proof is inspired by the undirected case studied in Spantini et al. (2018), and can be found in the Appendix, where we also show that this result implies Theorem 1.1. The idea is that the first term $\big[(Ch(k) \cup Pa(k)) \cap [k]\big]$ captures the skeleton, which enumerates the total number of edges in the truth graph. Hence, any additional edges added in the second and third terms are erroneous. In particular, the second term corresponds to edges added by incorrectly oriented v-structures, and the last term corresponds to the fill edges, as in variable elimination (Koller and Friedman, 2009).

The above result performs permutation-based causal discovery, which doesn't require assuming that data arise from an underlying SCM, thus allowing us to recover the MEC. To further refine the MEC, we can also assume that the data arise from a non-linear SCM. We define a SCM based on a DAG $\vec{\mathcal{G}}$ as a causal model of the following form:

$$\epsilon = (\epsilon_1, \dots, \epsilon_d) \sim P_\epsilon, \quad X_i = f_i(X_{pa(i)}, \epsilon_i), \tag{8}$$

with $\epsilon_i \perp\!\!\!\perp \epsilon_j$, and assuming that each f_i is monotone in ϵ_i. Such an SCM can always be iteratively unrolled to be of the form $X = g(\epsilon)$, and when a topological order is known, g is triangular. Javaloy et al. (2023) show that this g can be made triangular in a canonical way following the KR transport, and thus TMI maps are able to fit any $\mathcal{C}^1$ SCM. Note that although we technically fit a TMI map in the $X \to \epsilon$ direction, this is an equivalent model as TMI maps are closed under inversion—see Javaloy et al. (2023, Section 4) for a discussion on the forward versus backward model.

The key benefit here is that while conditional-independence based causal discovery methods require interventional data to break Markov equivalence, TMI models are able to naturally do this under the SCM assumption. Monti et al. (2020) propose a method using nonlinear ICA to resolve the equivalence between edges of the form $X_i \to X_j$ and $X_i \leftarrow X_j$ by noticing the following. Without loss of generality, assume $i = 1, j = 2$:

$X_1 \to X_2$	$X_1 \leftarrow X_2$
$X_1 = \epsilon_1$	$X_1 = f_1(X_2, \epsilon_1)$
$X_2 = f_2(X_1, \epsilon_2)$	$X_2 = \epsilon_2$
$X_1 \perp\!\!\!\perp \epsilon_2, \quad X_2 \not\perp\!\!\!\perp \epsilon_1$	$X_2 \perp\!\!\!\perp \epsilon_1, \quad X_1 \not\perp\!\!\!\perp \epsilon_2$

Table 1: Causal and anti-causal edge permutations.

In Monti et al. (2020), an equivalence between SCMs and nonlinear ICA is drawn and the "sources" ϵ_i are inferred up to a permutation, which requires further optimization to select the correct permutation. The TMI model only requires finding the sparsest causal ordering, after which exact inference of ϵ_i can be made via a simple pass through the model, as $P_\epsilon = T_* P_X$. We collect this observation into the following result.

Theorem 3.2. *Assuming the setting of Theorem 3.1, suppose that π is identified as a maximally sparse permutation where $T_\pi P_X = P_\epsilon$, and suppose that the model class includes the KR transport between P_X and P_ϵ, producing the estimated graph $\hat{\mathcal{G}}_\pi$. For each edge (i, j) not participating in a v-structure, estimate $\hat{\epsilon}_i, \hat{\epsilon}_j = T(X_i), T(X_j)$ and perform the independence tests in Table 1. Assuming access to a consistent independence test, orienting the edge according to whether $X_i \perp\!\!\!\perp \epsilon_j$ or $X_j \perp\!\!\!\perp \epsilon_i$ recovers $\vec{\mathcal{G}}$.*

Proof. By Theorem 1.1, $\hat{\mathcal{G}}^{\pi}$ is Markov equivalent to $\vec{\mathcal{G}}$, and so their skeleton and v-structures coincide. In other words, they differ by edge flips not participating in a v-structure. Applying Table 1 correctly orients these edges, thus recovering $\vec{\mathcal{G}}$. □

4 Challenges and Discussion

So far, we have highlighted appealing properties of TMI maps as generative models that perform causal discovery. Javaloy et al. (2023) further highlight their ability to perform causal inference, enabling interventional and counterfactual queries. However, these appealing properties depend heavily on two practical problems, to which we do not yet have satisfying solutions:

- The difficulty of estimating KR transport between arbitrary distributions as TMI maps. More generally, what are the properties of TMI maps as conditional independence tests, and how should we enforce sparsity to this effect (e.g., via ℓ_1 regularization of a masking layer, or thresholding)?
- The difficulty of searching over permutations efficiently while training the TMI model at each iteration. It will likely not be feasible to train near convergence at each proposed permutation, except for very small problems.

Estimating KR Transports Our method for causal discovery hinges on thresholding, or sparsity-regularizing the Jacobian of the TMI model as a *de facto* conditional independence test. If the TMI model successfully learns the KR transport, this is a valid procedure. However, this has a number of practical difficulties—valid conditional independence testing is fundamentally not possible without distributional assumptions (Shah and Peters, 2020). Simultaneously, the approximation to the KR transport may converge at an arbitrarily slow rate without smoothness assumptions (Irons et al., 2022). Furthermore, the model class used in practice may be misspecified, even in the more flexible ICA-based model.

Permutation Search Searching over all permutations rapidly becomes prohibitively expensive, requiring the use of heuristics or greedy approaches. However, as this is a shared problem with order-based methods in general, there are several existing options. In particular, greedy sparsest permutation (GSP) (Solus et al., 2021), which proposes edge reversals that decrease the number of edges in the corresponding DAG, has been developed for use with permutation-based causal discovery, and hence its usage here would be natural. More classically, Teyssier and Koller (2005) propose a simple greedy local hill-climbing procedure with adjacent swaps, for any scoring function. Additive-noise SCM assumptions further allow sequential methods, which search for sink or source nodes amongst the remaining nodes at each iteration (Shimizu et al., 2011; Peters et al., 2014; Wang and Drton, 2020; Ye et al., 2020; Rolland et al., 2022; Montagna et al., 2023a,b), which can also be applied here.

The grand challenge of operationalizing TMI maps as causal discovery algorithms is that any algorithm will contain the first challenge as an inner loop to the second challenge, similar to the setting addressed in Deng et al. (2023). That is, for each permutation, evaluating the score depends on finding the correct conditions under which TMI maps are conditional independence tests. Further, in cases where TMI maps are serviceable conditional independence tests, the complexity of the causal discovery algorithm can depend significantly on its optimization efficiency. Although the overall challenge is significant, progress has been made on each sub-problem.

5 Empirical Study

We perform proof-of-concept experiments to demonstrate that our methodology can work for small scale problems under ideal settings, and also highlight the challenges yet to be addressed. For our experiments, we consider the DAG $\vec{\mathcal{G}}$, and its label-permuted variants, in Figure 1 as ground truth. We generate non-linear synthetic data according to this graph by sampling from a standard Gaussian and passing it through an MLP with masked first-layer connections, and weights initialized to positive values to ensure monotonicity, and finally standardizing by mean and standard deviation (Reisach et al., 2021). In all experiments, we use SOS flows (Jaini et al., 2019) with $k = 5$ quadratic terms in the integrand and a single flow layer. In order to use a uniform threshold on the mean of the absolute value of the Jacobian to determine non-zero entries, we divide each row by its ℓ_1-norm.

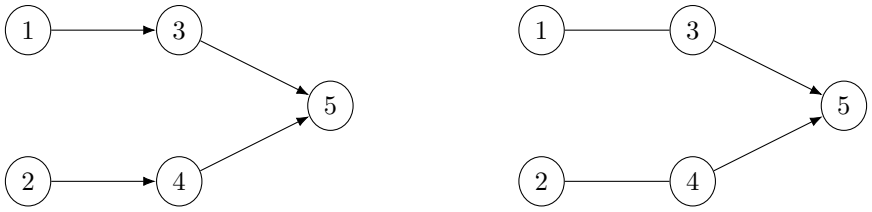
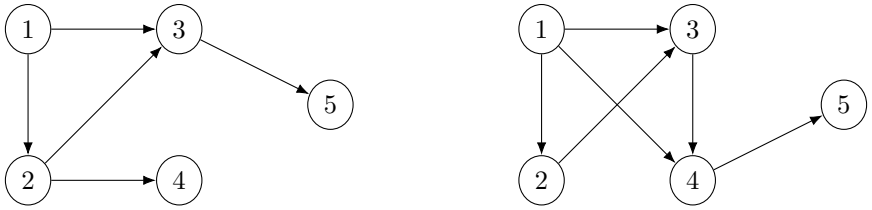


Figure 1: Ground truth $\vec{\mathcal{G}}$ (left) and associated MEC drawn as a mixed graph (right).



(a) DAG implied by $\pi_r = \{5, 4, 3, 2, 1\}$. (b) DAG implied by $\pi_{rs} = \{5, 2, 4, 3, 1\}$.

Figure 2: Incorrect orders introduce additional edges. Construction details are in the Appendix.

To examine Theorem 3.1, we consider three possible permutations, ranging from least to most sparse. First, the correct topological order $\pi^* = [d] = \{1, 2, 3, 4, 5\}$ should correctly recover the graph, with 4 edges. Second, the reverse order $\pi_r = \{5, 4, 3, 2, 1\}$ results in an extra edge added due to the incorrectly oriented v-structure $3 \to 1 \leftarrow 2$, resulting in the $2 \to 3$ edge being added. Finally, the reverse-swap order $\pi_{rs} = \{5, 2, 4, 3, 1\}$ has the incorrectly oriented v-structure, which in this case results in the $3 \to 4$ edge, and a fill-in edge $1 \to 2$, resulting in 6 edges. The corresponding graphs are given in Figure 2.

We give results for several different thresholds for a typical run in Table 2, evaluated on a held-out test set. All permutations lead to nearly identical fit in terms of negative log-likelihood values. This table shows the difficulty of using thresholding to search for a sparsest permutation—choosing the sparsest permutation appears to select the correct permutation in this toy problem when the threshold is small, but can be misleading at larger thresholds. A visual inspection of the Jacobian values (figures are provided in the Appendix) for each permutation details the problem. The true permutation is well-behaved in the sense that edges are dramatically larger than non-edges (Fig. 6), and so thresholding works well for determining the sparsity. However, this is not the case for incorrect permutations (Fig. 7, Fig. 8), where the estimated sparsity differs dramatically by the threshold, as spurious Jacobian entries can be close in value to Jacobian entries that do correspond to an edge in the DAGs detailed in Fig. 2.

Threshold/Permutation	π^* (Expect 4 edges)	π_r (Expect 5 edges)	π_{rs} (Expect 6 edges)
0.05	5	8	7
0.1	4	6	7
0.2	4	3	3
0.3	3	2	2

Table 2: Number of dense entries in the Jacobian at different thresholding values.

Finally, we examined the ability of TMI models to break the Markov equivalence (Theorem 3.2) by fitting an SCM-type model. Specifically, we estimated the latent variables from a TMI map in two maximally sparse, and therefore Markov equivalent permutations: the true ordering $\pi^* = [1, 2, 3, 4, 5]$, and the ordering $\pi_f = [3, 2, 1, 4, 5]$, which flips the edges $1 \to 3$ to the anti-causal direction. We used Hoeffding's permutation test for independence (Hoeffding, 1948) to test for the independencies in Table 1. We observed very weak evidence for $X_1 \perp\!\!\!\perp \epsilon_3$ in π^* (test statistic ≈ 0.01, smaller is more independent) over $X_1 \perp\!\!\!\perp \epsilon_3$ in π_f (test statistic ≈ 0.02). We leave a detailed study using more sophisticated independence tests for future work.

References

Akbari, S., Ganassali, L., and Kiyavash, N. (2023). Learning causal graphs via monotone triangular transport maps. *arXiv preprint arXiv:2305:18210*.

Baptista, R., Marzouk, Y., Morrison, R. E., and Zahm, O. (2021). Learning non-gaussian graphical models via hessian scores and triangular transport. *arXiv preprint arXiv:2101.03093*.

Bogachev, V. I., Kolesnikov, A. V., and Medvedev, K. V. (2005). Triangular transformations of measures. *Sbornik: Mathematics*, 196(3).

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

Deng, C., Bello, K., Aragam, B., and Ravikumar, P. K. (2023). Optimizing NOTEARS objectives via topological swaps. In *ICML 2023*, pages 7563–7595.

Hauser, A. and Bühlmann, P. (2012). Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1):2409–2464.

Hoeffding, W. (1948). A Non-Parametric Test of Independence. *The Annals of Mathematical Statistics*, 19(4):546 – 557.

Irons, N. J., Scetbon, M., Pal, S., and Harchaoui, Z. (2022). Triangular flows for generative modeling: Statistical consistency, smoothness classes, and fast rates. In *AISTATS 2022*.

Jaini, P., Selby, K. A., and Yu, Y. (2019). Sum-of-squares polynomial flow. In *ICML 2019*.

Javaloy, A., Sánchez-Martín, P., and Valera, I. (2023). Causal normalizing flows: from theory to practice. *arXiv:2306.05415*.

Kalisch, M. and Bühlman, P. (2007). Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research*, 8(3).

Khemakhem, I., Monti, R., Leech, R., and Hyvarinen, A. (2021). Causal autoregressive flows. In *AISTATS 2023*.

Koller, D. and Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.

Lachapelle, S., Brouillard, P., Deleu, T., and Lacoste-Julien, S. (2020). Gradient-based neural DAG learning. In *ICLR 2020*.

Maathuis, M., Drton, M., Lauritzen, S., and Wainwright, M. (2018). *Handbook of graphical models*. CRC Press.

Montagna, F., Noceti, N., Rosasco, L., Zhang, K., and Locatello, F. (2023a). Causal discovery with score matching on additive models with arbitrary noise. In *Conference on Causal Learning and Reasoning (CLeaR)*.

Montagna, F., Noceti, N., Rosasco, L., Zhang, K., and Locatello, F. (2023b). Scalable causal discovery with score matching. In *Conference on Causal Learning and Reasoning (CLeaR)*.

Monti, R. P., Zhang, K., and Hyvärinen, A. (2020). Causal discovery with general non-linear relationships using non-linear ICA. In *UAI 2020*.

Morrison, R., Baptista, R., and Marzouk, Y. (2017). Beyond normality: Learning sparse probabilistic graphical models in the non-gaussian setting. *NeurIPS*.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *The Journal of Machine Learning Research*, 22(1):2617–2680.

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

Peters, J., Janzig, D., and Schölkopf, B. (2017). *Elements of Causal Inference*. MIT Press.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15(58):2009–2053.

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

Reisach, A., Seiler, C., and Weichwald, S. (2021). Beware of the simulated dag! causal discovery benchmarks may be easy to game. *NeurIPS 2021*, 34:27772–27784.

Reizinger, P., Sharma, Y., Bethge, M., Schölkopf, B., Huszár, F., and Brendel, W. (2022). Jacobian-based causal discovery with nonlinear ICA. *Transactions on Machine Learning Research*.

Rolland, P., Cevher, V., Kleindessner, M., Russell, C., Janzing, D., Schölkopf, B., and Locatello, F. (2022). Score matching enables causal discovery of nonlinear additive noise models. In *ICML*.

Sadeghi, K. (2017). Faithfulness of probability distributions and graphs. *Journal of Machine Learning Research*, 18(148).

Shah, R. D. and Peters, J. (2020). The hardness of conditional independence testing and the generalised covariance measure. *The Annals of Statistics*, 48(3):1514 – 1538.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., Kerminen, A., and Jordan, M. (2006). A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10).

Shimizu, S., Inazumi, T., Sogawa, Y., Hyvarinen, A., Kawahara, Y., Washio, T., Hoyer, P. O., Bollen, K., and Hoyer, P. (2011). Directlingam: A direct method for learning a linear non-gaussian structural equation model. *Journal of Machine Learning Research*, 12:1225–1248.

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

Spantini, A., Bigoni, D., and Marzouk, Y. (2018). Inference via low-dimensional couplings. *The Journal of Machine Learning Research*, 19(1).

Spirtes, P., Glymour, C. N., and Scheines, R. (2000). *Causation, prediction, and search*. MIT press.

Squires, C., Wang, Y., and Uhler, C. (2020). Permutation-based causal structure learning with unknown intervention targets. In *UAI 2020*, pages 1039–1048.

Teyssier, M. and Koller, D. (2005). Ordering-based search: a simple and effective algorithm for learning bayesian networks. In *UAI 2005*.

Uhler, C., Raskutti, G., Bühlmann, P., and Yu, B. (2013). Geometry of the faithfulness assumption in causal inference. *The Annals of Statistics*.

Verma, T. and Pearl, J. (1990). Causal networks: Semantics and expressiveness. *Machine intelligence and pattern recognition*, 9:69–76.

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

Xi, Q. and Bloem-Reddy, B. (2023). Indeterminacy in generative models: Characterization and strong identifiability. In *AISTATS 2023*.

Yang, K., Katcoff, A., and Uhler, C. (2018). Characterizing and learning equivalence classes of causal dags under interventions. In *ICML 2018*, pages 5541–5550. PMLR.

Ye, Q., Amini, A. A., and Zhou, Q. (2020). Optimizing regularized cholesky score for order-based learning of bayesian networks. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3555–3572.

Zhang, K. and Hyvärinen, A. (2009). On the identifiability of the post-nonlinear causal model. In *UAI 2009*, pages 647–655. AUAI Press.

Zheng, X., Dan, C., Aragam, B., Ravikumar, P., and Xing, E. (2020). Learning sparse nonparametric dags. In *AISTATS 2020*, pages 3414–3425.

6 Supplementary Material

6.1 Proofs

Our proof of Theorem 3.1, and hence of Theorem 1.1, is based on iteratively marginalizing the moral graphs required to evaluate the d-separations

$$k \perp\!\!\!\perp j \mid \{1, \ldots, k-1\} \setminus \{j\}, \quad \text{for each } j < k. \tag{9}$$

We first introduce some background concepts.

6.1.1 Moral Graphs

For background, in an undirected graph $\mathcal{G}$, the subsets $I_1, I_2 \subseteq [d]$ are said to be u-separated by $I_3 \subseteq [d]$ if every possible path from a node in I_1 to a node in I_2 must traverse through a non-empty subset of I_3. We denote this as

$$I_1 \perp\!\!\!\perp_{\mathcal{G}} I_2 \mid I_3. \tag{10}$$

An equivalent way to evaluate d-separation is to use u-separation in the moral graph Koller and Friedman, 2009, Proposition 4.10. Let $I_1, I_2, I_3 \subseteq [d]$, let $I = I_1 \cup I_2 \cup I_3$, and consider their ancestral set $An_I := An(I, \vec{\mathcal{G}})$. For each v-structure in the subgraph $\vec{\mathcal{G}}_{An_I}$, connect the co-parents in each v-structure and drop directions on all directed edges to create the undirected moral graph over An_I, which we denote $\mathcal{G}_{M,An_I}$. Then, we have

$$I_1 \perp\!\!\!\perp_{\vec{\mathcal{G}}} I_2 \mid I_3, \tag{11}$$

denoting d-separation in $\vec{\mathcal{G}}$, if and only if

$$I_1 \perp\!\!\!\perp_{\mathcal{G}_{M,An_I}} I_2 \mid I_3, \tag{12}$$

denoting u-separation in the moralized graph.

6.1.2 Marginalization

One key difference between undirected graphs and directed graphs is closure under marginalization. More formally, let $I \subseteq [d]$, and $\vec{\mathcal{G}} = ([d], E)$ be a DAG. The existence of a DAG $\vec{\mathcal{G}}_I = (I, E')$ such that, for any $I_1, I_2, I_3 \subseteq I$, we have

$$I_1 \perp\!\!\!\perp_{\vec{\mathcal{G}}} I_2 \mid I_3 \iff I_1 \perp\!\!\!\perp_{\vec{\mathcal{G}}_I} I_2 \mid I_3, \tag{13}$$

is referred to as closure under marginalization. Many DAGs are not closed under arbitrary marginalization, see Maathuis et al. (2018, Figure 2.2) for an example.

On the other hand, undirected graphs are always closed (in terms of u-separation) under marginalization. Let $\mathcal{G} = ([d], E)$ be a graph and consider marginalization over a single index, i.e., $I = [d] \setminus \{i\}$. The marginal graph is $\mathcal{G}_I = (I, E')$, where E' is obtained as follows. Take the subgraph $E|_I$ (retain all edges with both vertices in I), and connect the neighbours of i (in other words, turning its neighbourhood into a clique (Spantini et al., 2018)). That is, for each pair $j, k \in Nb(i, \mathcal{G})$, we add $(j, k) \in E'$, these are known as fill edges (Koller and Friedman, 2009). In general, marginalizing over a set turns its outer boundary into a clique, as described in the following Lemma.

Lemma 6.1. *Let $\mathcal{G} = ([d], E)$ be an undirected graph. Let $I \subseteq [d]$ be nonempty, and let $M = [d] \setminus I$ denote the set of nodes to be marginalized out. Let $\mathcal{G}_M$ denote the sub-graph on M and let $M_C \in \mathcal{C}$ denote its connected components. Then, the marginal graph over I is $\mathcal{G}_I = (I, E')$, where*

$$E' = E|_I \bigcup_{M_C \in \mathcal{C}} \{(j, k) \mid j, k \in OBd(M_C, \mathcal{G})\}. \tag{14}$$

Proof. Let $i \in M_C$. Marginalizing over i connects its neighbours. If $|M_C| = 1$, $Nb(i, \mathcal{G}) = OBd(M_C, \mathcal{G})$, not affecting marginalization over other $i' \in M$ since $|M_C| = 1$ implies it has no other neighbours in M. Otherwise, let $|M_C| = k < d$. Marginalizing over an arbitrary $i_1 \in M_C$, its neighbours not in M_C are passed onto some i_2. Continuing until i_k connects all neighbours of $i_1, \ldots, i_k$ outside of M_C, $OBd(M_C, \mathcal{G})$. By definition of M_C, $OBd(M_C, \mathcal{G})$ cannot overlap with any other neighbourhoods in M. □

6.1.3 Proof of Theorem 3.1

The proof of Theorem 3.1 builds the sequence of moral graphs corresponding to $\{1, 2, \ldots, k\} \subseteq [d]$, for a given permutation $\pi = [d]$, and marginalizes out the ancestors to identify the neighbourhood of node k at each step.

Proof of Theorem 3.1. By faithfulness, the TMI map T has sparse entries corresponding to the d-separations

$$k \perp\!\!\!\perp j \mid \{1, \ldots, k-1\} \setminus \{j\}, \quad \text{for each } j < k. \tag{15}$$

First consider node d. The relevant d-separations in this step are:

$$d \perp\!\!\!\perp j \mid \{1, \ldots, d-1\} \setminus \{j\}, \quad \text{for all } j < d. \tag{16}$$

We evaluate these d-separations via the moral graph. Since the participating subsets form the whole graph, the moral graph $\mathcal{G}^d := \mathcal{G}_M$ is obtained simply by connecting co-parents and dropping directions. Then, notice that

$$X_d \not\perp\!\!\!\perp_{\mathcal{G}^d} X_j \mid \{X_1, \ldots, X_{d-1}\} \setminus \{X_j\}, \tag{17}$$

is equivalent to $j \in Nb(d, \mathcal{G}_{UG})$, since all other nodes are in the separating set and thus only immediate neighbours are u-connected (i.e., they are the tail boundary (Verma and Pearl, 1990)). We have

$$Nb(d, \mathcal{G}_{UG}) = Pa(d) \cup Ch(d) \cup Pa(Ch(d)) \tag{18}$$

where $Pa(d), Ch(d)$ are obviously neighbours of d, and $Pa(Ch(d))$ captures the previously unconnected co-parents, that are added via moralization (this is the Markov blanket of node d (Peters et al., 2017, Proposition 6.27)).

Now, consider node $k < d$. The relevant d-separations in this step are:

$$k \perp\!\!\!\perp j \mid \{1, \ldots, k-1\} \setminus \{j\}, \quad \text{for all } j < k. \tag{19}$$

Notice we can use the same moral graph to evaluate these d-separations, since the participating subsets are such that their union is always $[k] = \{1, \ldots, k\}$. We define $A_k := An([k])$. Thus, the moral graph is $\mathcal{G}_{M,A_k}$. However, simply using the moral graph includes possible ancestors not in $[k]$, and thus we cannot directly use the neighbourhood as we did with node d. However, due to closure under marginalization, we can use the neighbourhood of the marginalized undirected graph, which we denote $\mathcal{G}^k := \mathcal{G}_{UG,[k]}$, to determine the dense entries. We describe its neighbourhood as follows.

Consider the full moral graph $\mathcal{G}_{M,A_k}$. The neighbourhood is:

$$Nb(k, \mathcal{G}_{M,A_k}) = \big[Ch(k) \cup Pa(k) \cup Pa(Ch(k) \cap A_k)\big] \cap A_k. \tag{20}$$

In words, the neighbourhood contains the children and parents of k in A_k, plus any unconnected coparents of children in A_k. Next, by Lemma 6.1, the marginalized graph is $\mathcal{G}^k = ([k], E')$, where

$$E' = E|_{[k]} \bigcup_{E_{k,C} \in \mathcal{C}} \{(j,k) \mid j, k \in OBd(E_{k,C}, \mathcal{G}_{M,A_k})\}, \tag{21}$$

and $E_{k,C} \in \mathcal{C}$ denotes connected components in the subgraph defined by E_k. Notice now if $k \in OBd(E_{k,C})$ for any C, then all of $OBd(E_{k,C})$ is added to its neighbourhood. Denote $\mathcal{C}_k \subseteq \mathcal{C}$ (possibly empty) the set of connected components where $k \in OBd(E_{k,C})$. We thus have

$$Nb(k, \mathcal{G}^k) = (Nb(k, \mathcal{G}^k_e) \setminus E_k) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_{M,A_k}) \tag{22}$$

$$= \left(\left[\big[Ch(k) \cup Pa(k) \cup Pa(Ch(k) \cap A_k)\big] \cap A_k \right] \setminus E_k \right) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_{M,A_k}) \tag{23}$$

$$= \left(\big[Ch(k) \cup Pa(k) \cup Pa(Ch(k) \cap A_k)\big] \cap [k] \right) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_{M,A_k}) \tag{24}$$

$$= \big[(Ch(k) \cup Pa(k)) \cap [k]\big] \cup (Pa(Ch(k) \cap A_k) \cap [k]) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_{M,A_k}). \tag{25}$$

□

6.1.4 Proof of Theorem 1.1

We can now prove a version of Theorem 1.1, that the permutation corresponding to the sparsest Jacobian (equivalently, graph) under the sequence in Theorem 3.1 yields a graph in the MEC of $\vec{\mathcal{G}}$. First, we state a intermediate result that may be of independent interest, stating that the number of edges in $\vec{\mathcal{G}}$ lower bounds the number of dense entries in J_{T_π}.

Corollary 6.2. *Assume the setting of Theorem 3.1, where $\vec{\mathcal{G}} = ([d], E)$. For any ordering π, the Jacobian $J_{T_\pi}(x)$ is such that*

$$\| \max_{x \in \mathbb{R}^d} |J_{T_\pi}(x)| \|_0 \geq |E|, \tag{26}$$

with equality if π is a topological order, in which case we also have

$$J_T(x)_{\pi(k),\pi(l)} \neq 0, \textit{ for } \pi(l) \in Pr(\pi_t(k)) \iff \pi(l) \to \pi(k), \tag{27}$$

thus recovering the true $\vec{\mathcal{G}}_\pi$.

Proof. To prove the first claim, note that the first term of the sequence in Theorem 3.1:

$$\big[(Ch(\pi(k)) \cup Pa(\pi(k))) \cap \{\pi(1), \ldots, \pi(k)\}\big] \tag{28}$$

enumerates all the edges of the true DAG $\vec{\mathcal{G}}_\pi$ in any order. This is easy to see. For $\pi(k) = d$, the above term counts the children and parents of node d, each of which represents an edge into or out of d. Then, for $\pi(k) = d - 1$, the above term counts the edges into or out of $d - 1$, but not related to d. Continuing in this way enumerates all edges of the DAG. This gives us a lower bound for the cardinality of dense entries in $J_T(x)$ (i.e., the ℓ_0-norm $\| \max_{x \in \mathbb{R}^d} |J_T(x)| \|_0$).

For the second claim, W.L.O.G., assume the canonical order $\pi(i) = i$ is a topological order. Invoking Theorem 3.1, we have

$$J_T(x)_{k,l} \neq 0 \iff \tag{29}$$

$$l \in \big[(Ch(k) \cup Pa(k)) \cap [k]\big] \cup (Pa(Ch(k) \cap A_k) \cap [k]) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_e^k). \tag{30}$$

Since the nodes are in topological order, ancestors of k are always in $[k]$, and thus $An([k]) = [k]$. This implies that $E_k = An([k]) \cap \{k+1, \ldots, d\} = \emptyset$, and that $Pa(Ch(k) \cap A_k) = Pa(\emptyset) = \emptyset$. Thus, for each k, we have the following dense entries:

$$(Ch(k) \cup Pa(k)) \cap [k] = Pa(k). \tag{31}$$

Thus, the k-th row contains dense entries in the parents of k, recovering the DAG $\vec{\mathcal{G}}$. □

The above corollary reveals that we can view the dense entries in row k of $J_T(x)$ are the inferred parents of node k, assuming that the order of the variables is a topological order (even if it is not). In this way, we can characterize the extra dense entries as mistakenly added edges:

$$\big[(Pa(Ch(k) \cap A_k) \cap [k]) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_e^k)\big] \setminus \big[(Ch(k) \cup Pa(k)) \cap [k]\big]. \tag{32}$$

These are the elements of the second and third terms that do not overlap with the existing edges. We can see that orders that induce maximal sparsity must correctly orient the v-structures of a DAG, or else it will incur a dense entry corresponding to the first term. Using this, we can show that maximally sparse orders are able to recover a graph that is Markov equivalent to the true DAG. We prove the following version of Theorem 1.1.

Corollary 6.3. *Assume the setting of Theorem 3.1, where $\vec{\mathcal{G}} = ([d], E)$. Let π be an order such that*

$$\| \max_{x \in \mathbb{R}^d} |J_{T,\pi}(x)| \|_0 = |E|. \tag{33}$$

Then, the adjacency matrix implied by $J_{T^\pi}(x)$ is Markov equivalent to $\vec{\mathcal{G}}_\pi$.

Proof. Again, assume W.L.O.G. the canonical order $\pi(i) = i$. We first show that all v-structures must be correctly oriented. This means that, for any $i \to j \leftarrow k$, we have $j > i, k$. To do this, W.L.O.G. assume for a contradiction the existence of a v-structure such that $j < k$. Then, in the k-th step of the sequence in Theorem 3.1, we have $j \in Ch(k) \cap A_k$, and thus $i \in Pa(Ch(k) \cap A_k)$. Furthermore, $i \notin Ch(k) \cup Pa(k)$, and thus

$$\big[(Pa(Ch(k) \cap A_k) \cap [k]) \bigcup_{E_{k,C} \in \mathcal{C}_k} OBd(E_{k,C}, \mathcal{G}_e^k)\big] \setminus \big[(Ch(k) \cup Pa(k)) \cap [k]\big] \neq \emptyset, \tag{34}$$

implying that $\| \max_{x \in \mathbb{R}^d} |J_T(x)| \|_0 > |E|$, a contradiction.

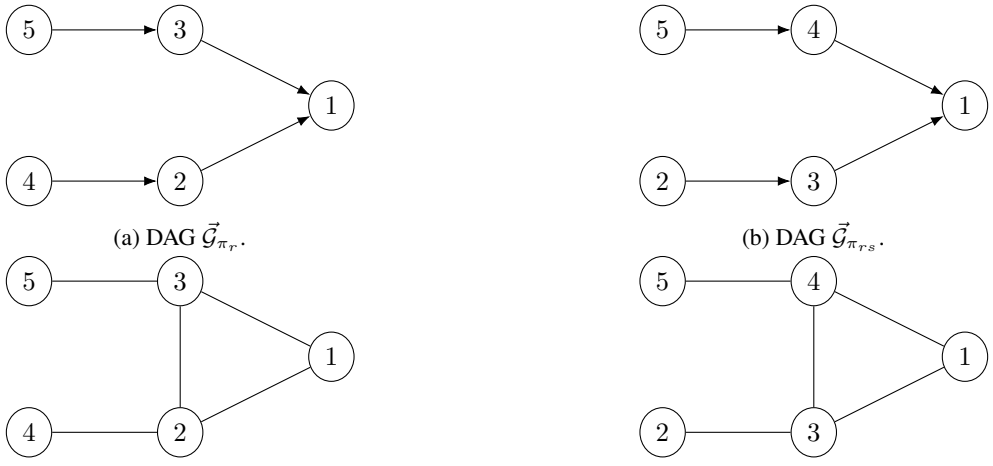
Now, recall the skeleton of the DAG is guaranteed via the first term:

$$\big[(Ch(k) \cup Pa(k) \cap [k]\big]. \tag{35}$$

Recall that these appear as parents of the inferred DAG. Thus, the DAG inferred by the TMI Jacobian is such that the v-structures are correctly oriented, and where outside of a v-structure, children may appear as parents. Thus, it is Markov equivalent to $\vec{\mathcal{G}}_\pi$. □

6.2 Detailed Description of Figure 2

As an example, we give further details on the incorrectly inferred DAGs in Figure 2. The ground-truth DAGs $\mathcal{G}DAG, \pi_r$ and $\vec{\mathcal{G}}_{\pi_{rs}}$, and their moral graphs are given in Figure 3. Note in both cases $An([k]) = \{1, 2, 3, 4, 5\}$ for all $k = 1, 2, 3, 4, 5$, since 1 is always a sink node, and thus the variable elimination sequence always operates on the same moral graph. In particular, as seen in Figure 2, π_r introduces an additional edge due to moralization, but does not produce any fill-in, since it remains a perfect elimination ordering. On the other hand, π_{rs} has the additional moralization edge, but is not a perfect elimination ordering, and thus introduces an additional fill-in edge as well. Details are in Fig. 4 and Fig. 5 respectively.



(a) DAG $\vec{\mathcal{G}}_{\pi_r}$.

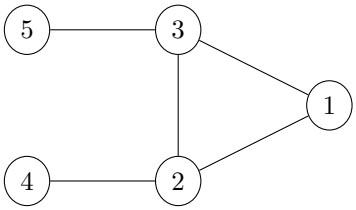
(b) DAG $\vec{\mathcal{G}}_{\pi_{rs}}$.

(c) Moral graph $\mathcal{G}_{M,\pi_r}$.

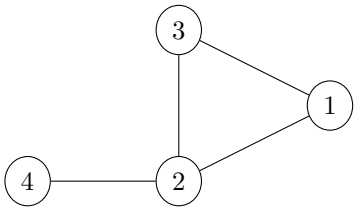
(d) Moral graph $\mathcal{G}_{M,\pi_{rs}}$.

Figure 3: Ground-truth DAGs for π_r and π_{rs}, and associated moral graphs.

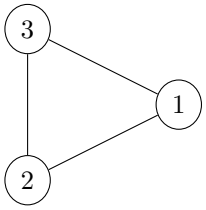
6.3 Mean Jacobian Plots for Table 2



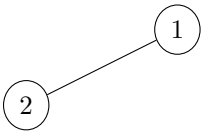
(a) At $k = 5$, $Nb(5, \mathcal{G}_{M,\pi_r}) = \{3\}$, and the dense entry is $(5, 3)$.



(b) At $k = 4$, $Nb(4, \mathcal{G}^4_{M,\pi_r}) = \{2\}$, and the dense entry is $(4, 2)$.

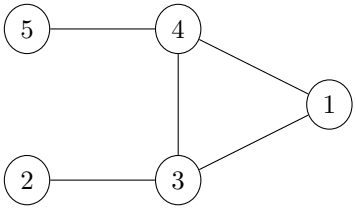


(c) At $k = 3$, $Nb(3, \mathcal{G}^3_{M,\pi_r}) = \{2, 1\}$, and the dense entries are $(3, 2)$, $(3, 1)$.

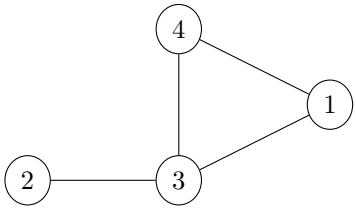


(d) At $k = 2$, $Nb(2, \mathcal{G}^2_{M,\pi_r}) = \{1\}$, and the dense entry is $(2, 1)$.

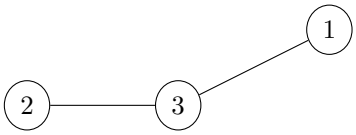
Figure 4: Sequence of marginalized moral graphs for π_r. Notice π_r does not produce any fill edges—i.e., it is a perfect elimination ordering.



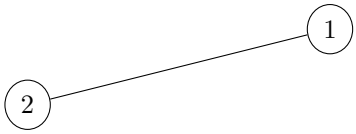
(a) At $k = 5$, $Nb(5, \mathcal{G}_{M,\pi_r}) = \{4\}$, and the dense entry is in position $(5, 4)$.



(b) At $k = 4$, $Nb(4, \mathcal{G}^4_{M,\pi_r}) = \{3, 1\}$, and the dense entries are $(4, 3)$, $(4, 1)$.



(c) At $k = 3$, $Nb(3, \mathcal{G}^3_{M,\pi_r}) = \{1, 2\}$, and the dense entries are $(3, 2)$, $(3, 1)$.



(d) At $k = 2$, $Nb(2, \mathcal{G}^2_{M,\pi_r}) = \{1\}$, and the dense entry is in position $(2, 1)$. This is a fill-in edge.

Figure 5: Sequence of marginalized moral graphs for π_{rs}. Notice π_{rs} is not a perfect elimination ordering, and thus introduces the fill-in edge $(2, 1)$, in addition to the moralization edge.

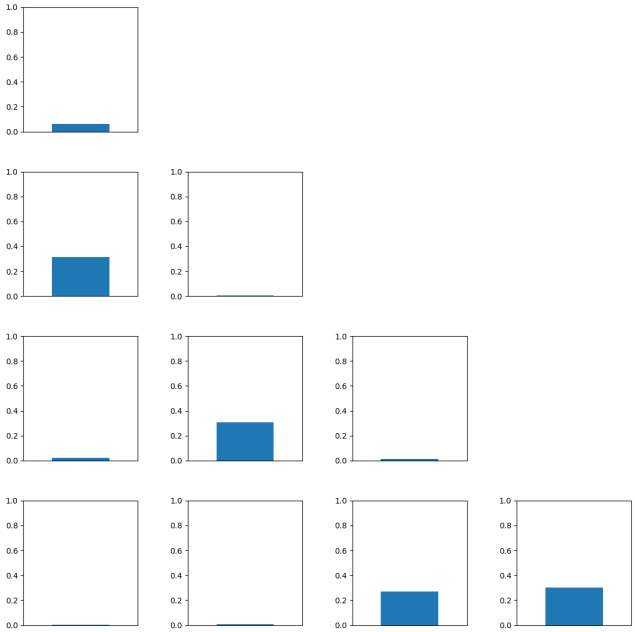


Figure 6: Array of lower-triangular mean Jacobian values for the true permutation π^*. A dense entry indicates row index $\rightarrow$ column index in the learned graph. Notice the small magnitudes of the sparse entries compared to the dense entries.

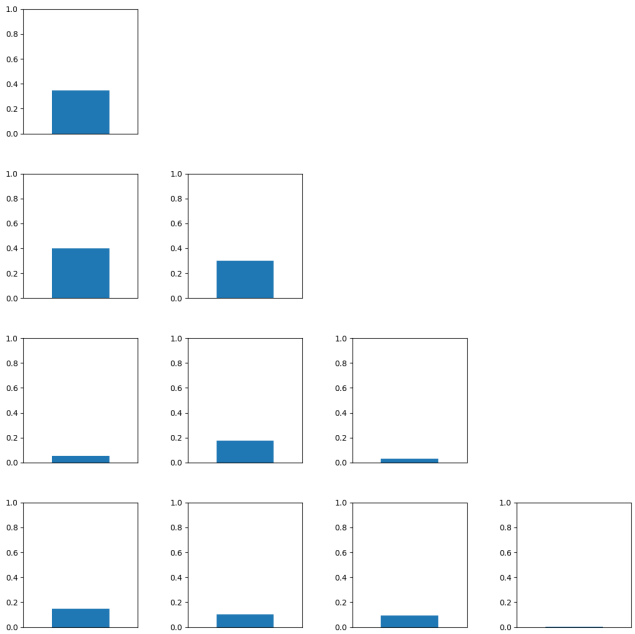


Figure 7: Array of lower-triangular mean Jacobian values for the permutation π_r. A dense entry indicates row index $\rightarrow$ column index in the learned graph.

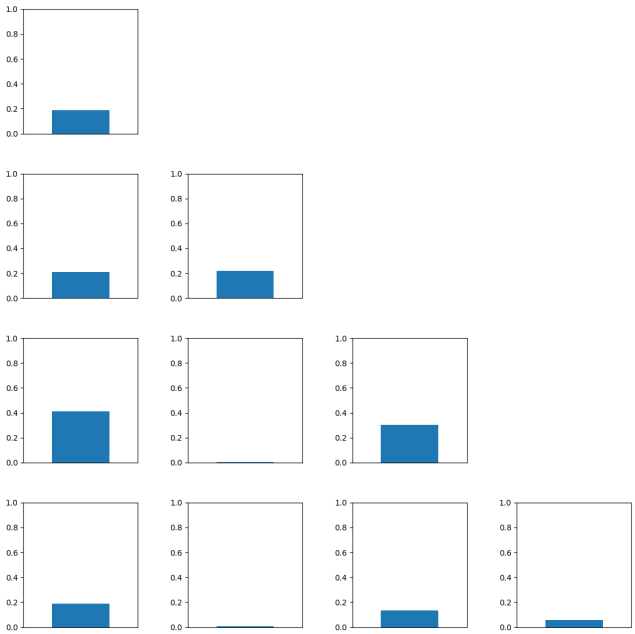


Figure 8: Array of lower-triangular mean Jacobian values for the permutation π_{rs}. A dense entry indicates row index $\rightarrow$ column index in the learned graph.