ARTICLE Communicated by Kun Zhang

Learning Minimal Latent Directed Information Polytrees

Jalal Etesami
etesami2@illinois.edu
Department of Industrial and Enterprise Systems Engineering, Coordinated Science Laboratory, University of Illinois at Urbana Champaign, Urbana, IL 61801, U.S.A.

Negar Kiyavash
kiyavash@illinois.edu
Department of Industrial and Enterprise Systems Engineering, Coordinated Science Laboratory, and Department of Electrical and Computer Engineering, University of Illinois at Urbana Champaign, Urbana, IL 61801, U.S.A.

Todd Coleman
tpcoleman@ucsd.edu
Department of Bioengineering, University of California, San Diego, La Jolla, CA 92093, U.S.A.

We propose an approach for learning latent directed polytrees as long as there exists an appropriately defined discrepancy measure between the observed nodes. Specifically, we use our approach for learning directed information polytrees where samples are available from only a subset of processes. Directed information trees are a new type of probabilistic graphical models that represent the causal dynamics among a set of random processes in a stochastic system. We prove that the approach is consistent for learning minimal latent directed trees. We analyze the sample complexity of the learning task when the empirical estimator of mutual information is used as the discrepancy measure.

1 Introduction

"Latent graphical models" refers to a class of probabilistic graphical models that encode the relationship between a set of observed and a set of hidden variables. Introducing latent variables can greatly improve the flexibility of probabilistic modeling, allowing it to address a diverse range of problems with hidden factors.

A class of directed graphical models that provide a succinct representation of causal dynamics was recently introduced by Quinn, Kiyavash, and Coleman (2015). In such models, nodes are random processes, and edges depict causation measured by causally conditioned directed information (Massey, 1990).

Neural Computation **28, 1723–1768** (2016) © 2016 Massachusetts Institute of Technology
doi:10.1162/NECO_a_00874

The main contribution of this article is to develop an approach for structure learning of a directed graphical model with polytree structure when only a subset of random processes is observed. Specifically, we will consider the scenario of latent directed information polytrees, where the directed information graph representing observed and unobserved processes is a tree with multiple roots. Learning such graphs requires both finding the number of hidden processes and recovering the connections among all hidden and observed nodes. To perform the learning task, we define a discrepancy measure between nodes of a directed tree and introduce an algorithm that identifies the structure given the discrepancies between only a subset of nodes (observed nodes). We further show the applicability of the proposed algorithm through simulating both synthetic and real-world data sets.

Polytree models have applications in real world. For instance, polytrees were implemented to enhance caching strategies in distributed databases (Messaouda, Oommen, & Matwin, 2003). Dependency polytrees were also applied to develop an inference framework that optimizes hardware components according to the performance and price of architectures (Zaveri & Hammerstrom, 2010). Sucar, Pérez-Brito, Ruiz-Suárez, and Morales (1997) applied a polytree structure graphical model for ozone prediction in Mexico City, where the ozone level is used as global indicator for air quality. Moreover, protein signaling pathways might be modeled by causal polytrees. For instance, an NFkB protein signaling pathway activates mammalian immune system cells to produce antibodies against inflammation (Lodish et al., 2000). de Campos (1994) characterizes dependency graphical models that are isomorphic to polytree graphs.

Even if the underlying structure is not a tree, there are efficient algorithms that approximate the underlying causal structure by the best directed tree such as Rebane and Pearl (1987); Quinn, Kiyavash, and Coleman (2013). Rebane and Pearl (1987) introduce an algorithm similar to the Chow-Liu algorithm (Chow & Liu, 1968) to construct a polytree-shaped network to approximate the probability distribution of the network.

Since in a directed polytree, a natural notion of hierarchy (depth) exists, polytree approximation can be used to infer the influence hierarchy among the processes. Such an inference could be helpful in, for instance, determining root causes of events or where to intervene for regulatory action such that it could effectively trickle down.

In contrast to previous latent tree learning approaches such as Choi, Tan, Anandkumar, and Willsky (2011), we do not require joint gaussianity or symmetry properties for the joint distribution of processes.

The remainder of the article is organized as follows. We formally introduce the background on directed information graphs and the concept of a minimal latent directed information tree in sections 3 and 4. In section 5, we present an approach that, given an appropriately defined discrepancy measure on the observed nodes, recovers the entire latent directed polytree under mild assumptions. In section 6, we introduce an example of such a discrepancy measure based on the time delays between pairs of processes that can be used to learn directed information polytrees. In section 7, we characterize the sample complexity of the learning algorithm. Our experimental results are presented in section 8. We conclude in section 9.

2 Related Work

Graphical models for representing the dependency and the causal relationship between a set of random variables have been studied extensively. We refer interested readers to Pearl (1988) and Koller and Friedman (2009), which provide good foundations for the theory and techniques. As the focus of this work is causal inference, we review the main approaches to represent causal interaction in a stochastic network.

Granger causality and the principle of intervention are the main frameworks to identify causal interactions in a causal network. The principle of intervention (Pearl, 2000) infers the causal relationships by fixing certain variables and allowing others to change, to see how these changes influence the statistics of the other variables. The idea of Granger causality is that a random process **X** is causing **Y**, if incorporating the past of **X** improves the prediction of the future of **Y**. Granger (1969) proposed a framework to capture this in an autoregressive (linear) setting. As part of the effort to generalize Granger's causality to more general settings, a class of graphical models, the Granger causality graph, was developed (Dahlhaus & Eichler, 2003; Dahlhaus, 2000; Eichler, 2007; Runge, 2014). This class of graphs consists of a mix of both directed and undirected edges for multivariate autoregressive time series, and the nodes in the graph represent processes. Although, Dahlhaus and Eichler (2003) suggested that, conceptually, Granger causality graphs could be employed for nonlinear relationships, in this case some canonical properties of the graphical models do not hold.

Directed information (DI) is an information-theoretic quantity that generalizes Granger causality beyond linear models (Rissanen & Wax, 1987). It was introduced by (Marko, 1973) and then later formalized by Massey (1990). DI has been used in many applications to infer causal relationships. For example, it has been used for analyzing neuroscience data (Quinn, Coleman, Kiyavash, & Hatsopoulos, 2011; Kim, Putrino, Ghosh, & Brown, 2011; Liu & Aviyente, 2010), gene regulatory data (Rao, Hero, States, & Engel, 2007), and video recordings (Chen, Hero, & Savarese, 2012).

Transfer entropy, introduced by Schreiber (2000), is another measure of causality in the literature (Chávez, Martinerie, & Le Van Quyen, 2003; Gourévitch & Eggermont, 2007). The relationship of DI and transfer entropy is discussed in detail in (Quinn, Coleman et al., 2011). Transfer entropy is defined only for processes that satisfy Markov property, in which case the DI can be written as a sum of a sequence of transfer entropies.

Directed information graphs (DIGs) define a graphical model that captures the generalization of Granger causality through the DI metric among stochastic processes (Quinn, Coleman et al., 2011). DIGs subsume Granger causal graphs. Quinn et al. (2015) showed that in order to guarantee the uniqueness of DIGs, the joint dynamics of the system must be strictly causal. In this letter, we consider learning the structure of DIGs in which a set of nodes is not observable (latent).

There are other type of graphical models, such as Bayesian networks (Pearl, 2000) and ancestral graphs (Zhang, 2008), that have been used to encode conditional independence relationships in directed acyclic graphs (DAGs). A dynamic Bayesian network (DBN) (Murphy, 2002) is a class of graphical models that extends Bayesian networks to model probability distributions over a semi-infinite collection of random variables. For example, hidden Markov models (HMMs) can be represented as DBNs. Since the size of DBNs depends on the time homogeneity and the Markov order of the random processes, in general, the graphs can grow with time. Thus, they are not well suited for providing succinct visualization of relationships between the past and the future of processes. As an example, the DBN graph of a vector autoregressive (VAR) process $\underline{\mathbf{X}}(t) \in \mathbb{R}^m$ of order L requires mL nodes (Dahlhaus & Eichler, 2003). Directed information graphs, the alternative we study, represent each random process as a node in the graph. Therefore, their size depends on neither the Markov order of processes nor the time (for the VAR example, the size is m).

In the past few decades, several approaches have been developed for learning these graphical models (e.g., Quinn, Coleman et al., 2011; Shimizu, Hoyer, Hyvärinen, & Kerminen, 2006; Hyvärinen & Smith, 2013). As the goal of this article is learning graphical models with latent nodes, we review some of the previous relevant latent learning algorithms. We categorize the learning approaches to graphs that represent conditional independence relationships among random variables such as Bayesian networks or ancestral graphs and random processes such as DI graphs. Note that some of the learning methods proposed for the former can be extended to the latter, but the methods such as the one presented in this work, which requires the notion of time delay among processes, are applicable to only the second type of graphical models.

One approach for learning latent graphical models is to fix the number of latent vertices and the structural relationships between latent and observed variables and subsequently use the expectation maximization (EM) algorithm to estimate the model parameters. Given that often the optimization is over a nonconvex function, the performance depends on initialization and the algorithm may get trapped in suboptimal local minima (Elidan, Friedman, & Chickering, 2005).

Jalali and Sanghavi (2012) consider learning a VAR model with hidden components. The model is identifiable under the assumptions that connections between observed variables are sparse and each latent variable interacts with many observed variables. Geiger, Zhang, Schölkopf, Gong, and Janzing (2015) and Boyen, Friedman, and Koller (1999) apply a method based on the EM algorithm to infer properties of partially observed Markov processes. Geiger et al. (2015) relax the finite-state condition required by Boyen et al. (1999) and provide sufficient conditions under which the partially observed Markov process is identifiable. Essentially they show that when the noise is independent and nongaussian or the observed variables do not influence the hidden variables, the model is identifiable. Chandrasekaran, Parrilo, and Willsky (2010) consider learning latent graphical models in the setting in which the latent and observed variables are jointly gaussian, the conditional statistics of the observed variables given the latent variables is a sparse graph, and the number of latent nodes is small relative to the number of observed variables. They propose a tractable convex program based on regularized maximum likelihood for latent-variable graphical model selection. However, our approach does not specify any model for the joint distribution between the observed and the latent variables. Furthermore, it may have a relatively large number of hidden variables. An alternative method (Elidan, Nachman, & Friedman, 2007) is based on a greedy, combinatorial heuristic that assigns latent variables to groups of observed variables via clustering of the observed variables. This approach has no consistency guarantees. In contrast, our approach guarantees consistency under mild assumptions even when the observed nodes are internal.

A provably sound algorithm known as FCI was developed for leaning maximal ancestral graphs (MAG) (Spirtes, Meek, & Richardson, 1995; Zhang, 2008).[1] A MAG is a mixed graph consisting of both directed and undirected edges on the set of observable variables that probabilistically represents the conditional independence among both latent and observable variables in an accompanying DAG. More precisely, consider any DAG (e.g., G over $V = O \cup L \cup S$) that encodes a set of conditional independence relations among nodes in V, where O and L denotes the set of observed and latent variables, respectively, and S denotes a set of unobserved selection variables to be conditioned on. Suppose there exists a MAG, $M(G)$, over O such that for any three disjoint sets of variables $A, B, C \subseteq O$, A and B are conditionally independent given $C \cup S$ in G if and only if A and B are conditionally independent given C in $M(G)$. In this case, $M(G)$ is said to probabilistically represent G. FCI algorithm recovers not the latent nodes and the relations between latent and observed nodes but the MAG on the set of observable nodes. Our algorithm recovers the graph on both observable and latent nodes.

Classical approaches to learning latent graphical models, in which nodes represent random variables, are of the following flavors. Latent cluster models (LCMs) learn a tree-structured Bayesian network, in which only one

[1] Soundness is defined as follows: given a perfect oracle of conditional independence, the algorithm outputs the Markov equivalence class of the true causal maximal ancestral graph.

single hidden variable exists (Lazarsfeld & Henry, 1968). Hierarchical latent class (HLC) models generalize the previous model by allowing multiple hidden variables, but they confine the observed variables to the leaves of the tree (Zhang & Kocka, 2004). Since in HLC models, root walking leads to a marginally equivalent model (two models are marginally equivalent if they share the same conditional distribution between the observable variables given the latent variables), it is impossible to learn edge orientation from the data.[2] Furthermore, learning algorithms for such models has a greedy structure, which is both computationally expensive and not guaranteed to be consistent.

Other popular learning methods for latent Markov graphical models use quartet-based distances (Jiang, Kearney, & Li, 2001; Erdos, Steel, Székely, & Warnow, 1999) to discover the structure.[3] Quartet-based methods first construct a set of quartets for all subsets of four observable nodes and then combine them to form a latent tree. It is known that the problem of determining a latent tree that agrees with the maximum number of quartets is NP-hard (Steel, 1992). As a result many heuristics have been developed (Farris, 1972; Sattath & Tversky, 1977). Ishteva, Park, and Song (2013) propose a quartet-based approach that uses rank characterization of the tensor associated with the marginal distribution of a quartet. This characterization allows them to design a nuclear norm-based test for resolving quartet relations. Additionally, in practice, quartet-based methods are often much less accurate than the neighbor-joining (NJ) method (John, Warnow, Moret, & Vawter, 2003). NJ (Saitou & Nei, 1987) is another distance-based algorithm that proceeds by repeatedly pairing the two closest nodes from the list by adding a new latent node as their parent and replacing the pair with the newly added node. Both NJ and the quartet-based methods rely on the existence of a notion of distance between nodes of a tree, which may not exist in many practical scenarios. In this work, we propose a new method based on a discrepancy measure between the observed nodes, which is not required to be a distance measure.

Recently the quartet-based approaches were applied to learn linear multivariate tree models when only the leaves are observed (Anandkumar et al., 2011). In such trees, nodes are multivariate random vectors. Anandkumar et al. (2011) further assumed that the conditional expected value of each node given the parent is a linear function of its parents. Recursive grouping (RG) (Choi et al., 2011) and Chow-Liu recursive grouping (CLRG) proposed in Chow and Liu (1968) and Choi et al. (2011) are two other distance-based learning algorithms that can recover latent Markov graphical models in which some of the observed nodes are internal. Both RG and CLRG can

[2] Root walking is an operation on a directed tree that reverses an arrow that goes from the root to one of its neighbors.

[3] A quartet is an unrooted binary tree on a set of four observed nodes.

only recover latent models on a set of hidden and observed random variables that are jointly gaussian or have a symmetric discrete joint distribution (Choi et al., 2011). No such restrictions on the joint are required in our approach.

3 Preliminary

In this section we present the background materials on directed information graphs.

3.1 Directed Information. Consider a stochastic dynamical system described by m random processes $\underline{\mathbf{X}} = (\mathbf{X}_1, \dots, \mathbf{X}_m)$ with joint distribution $P_{\underline{\mathbf{X}}}$ such that each random process contains n random variables. We denote the tth random variable in the ith process by $X_{i,t}$ and the random process $\mathbf{X}_i$ from time 1 up to time t by $X_{i,1}^t$. We use underlined characters to represent a collection of processes; for example, $\underline{X}_{\mathcal{K},1}^t$ is used to denote a set of random processes with index set $\mathcal{K}$ from time 1 up to time t. Using the chain rule (Schum, 1994) over increasing time indices, the joint distribution can be factored as

$$P_{\underline{\mathbf{X}}} = \prod_{t=1}^{n} P_{X_{1,t},\dots,X_{m,t}|X_{1,1}^{t-1},\dots,X_{m,1}^{t-1}}.$$

Throughout this article, we use $P_{X|Y}$ to denote the conditional distribution of X given $Y = y$, $P_{X|Y}(X|Y = y)$. There are many classes of models (e.g., stochastic differential equations) for which the joint statistics of all processes at time t are statistically independent, given the past of all other processes.

Example 1. Consider a dynamical system where the processes evolve over time by the following coupled differential equations:

$$\begin{aligned} dX &= f(X, Y)dt + dW, \\ dY &= g(X, Y)dt + dV, \end{aligned}$$

where W and V are independent Wiener processes. For small Δ, this becomes

$$\begin{aligned} X_{t+\Delta} &\approx X_t + \Delta f(X_t, Y_t) + W_{t+\Delta} - W_t, \\ Y_{t+\Delta} &\approx Y_t + \Delta g(X_t, Y_t) + V_{t+\Delta} - V_t. \end{aligned}$$

In this system, given the full past of the system, (X_1^t, Y_1^t), $X_{t+\Delta}$ is independent of $Y_{t+\Delta}$.

In these strictly causal systems, given the full past, the future of processes is conditionally independent, and the joint distribution can be simplified to

$$P_{\underline{\mathbf{X}}} = \prod_{t=1}^{n}\prod_{j=1}^{m} P_{X_{j,t}|X_{1,1}^{t-1},\dots,X_{m,1}^{t-1}}. \tag{3.1}$$

Using causal conditioning notation introduced by Kramer (1998),[4]

$$P_{\mathbf{X}_j \| \underline{\mathbf{X}}_{-\{j\}}} := \prod_{t=1}^{n} P_{X_{j,t}|X_{1,1}^{t-1},\dots,X_{j,1}^{t-1},\dots,X_{m,1}^{t-1}}, \tag{3.2}$$

we can rewrite equation 3.1 as

$$P_{\underline{\mathbf{X}}} = \prod_{j=1}^{m} P_{\mathbf{X}_j \| \underline{\mathbf{X}}_{-\{j\}}}, \tag{3.3}$$

where $-\{j\} := \{1,\dots,m\}\setminus\{j\}$. Note that the notation in equation 3.2 is defined analogous to the definition of regular conditional distribution: $P_{\mathbf{X}_j | \underline{\mathbf{X}}_{-\{j\}}}$. Recall that using chain rule, one can rewrite it as

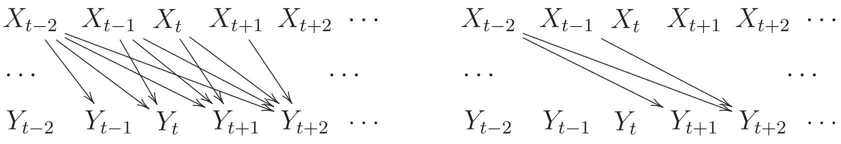
$$P_{\mathbf{X}_j | \underline{\mathbf{X}}_{-\{j\}}} = \prod_{t=1}^{n} P_{X_{j,t}|X_{j,1}^{t-1},\underline{\mathbf{X}}_{-\{j\}}}. \tag{3.4}$$

Notation 3.2 is defined by excluding the present and the future of $\underline{\mathbf{X}}_{-\{j\}}$ in equation 3.4. Note that we always condition on the past of $\mathbf{X}_j$ in both equations 3.2 and 3.4. Let $\mathcal{K} = \{k_1,\dots,k_s\} \subseteq -\{j\}$ be a subset of indices except j and $\underline{\mathbf{X}}_{\mathcal{K}}$ to be the set of processes with index set $\mathcal{K}$. It is possible to generalize notation 3.2 as

$$P_{\mathbf{X}_j \| \underline{\mathbf{X}}_{\mathcal{K}}} := \prod_{t=1}^{n} P_{X_{j,t}|X_{j,1}^{t-1},X_{k_1,1}^{t-1},\dots,X_{k_s,1}^{t-1}}.$$

In equation 3.2, the random process $\mathbf{X}_j$ depends on the set of random processes $\underline{\mathbf{X}}_{-\{j\}}$ by one time delay. This notation may be generalized to d-step delay ($d \in \mathbb{N}$). We denote the causal conditioned distribution with d-step delay as

[4] Note the slight difference in conditioning upon $X_{j,1}^{i-1}$ in this definition as compared to $X_{j,1}^{i}$ in the original causal conditioning definition.



(a) $P_{\mathbf{Y}\|\mathbf{X}}$.

(b) $P_{\mathbf{Y}\|_3\mathbf{X}}$.

Figure 1: Time dependencies between random processes **X** and **Y** for a unit delay and three-step delay in example 2. Directed edges show the causal conditioned dependencies between variables in process **Y** and the corresponding variables in process **X**.

$$P_{\mathbf{X}_j\|_d\,\underline{\mathbf{X}}_{\mathcal{K}}} := \prod_{t=1}^{n} P_{X_{j,t}|X_{j,1}^{t-1},\underline{\mathbf{X}}_{\mathcal{K},1}^{t-d}}. \tag{3.5}$$

In equation 3.5, $\underline{\mathbf{X}}_{\mathcal{K},1}^{t-1}$ stands for $(X_{k_1,1}^{t-1}, \ldots, X_{k_s,1}^{t-1})$.

Example 2. Let **X** and **Y** to be two random processes. In this example, we compare the causal conditioned ($P_{\mathbf{Y}\|\mathbf{X}}$) with causal conditioned distribution with three-step delay ($P_{\mathbf{Y}\|_3\mathbf{X}}$). Figure 1 illustrates the time interdependencies between **X** and **Y** in these two cases. Directed edges in these figures show the causal conditioned dependencies between variables in process **Y** and the corresponding variables in process **X**. Note that the intradependencies for each process are omitted.

It is easy to see that for $d = 1$, equation 3.5 becomes Kramer's causal conditioned distribution 3.2. For simplicity, we will write $P_{\mathbf{X}\|\mathbf{Y}}$ instead of $P_{\mathbf{X}\|_1\mathbf{Y}}$.

Assumption 1. *For the remainder of this article, we only consider a collection of random processes for which there exists a reference measure ϕ such that $P_{\underline{\mathbf{X}}} \ll \phi$ and $\frac{dP_{\underline{\mathbf{X}}}}{d\phi} > 0$ (such joint distribution is called positive) and the joint distribution is strictly causal; it is given by equation 3.3.*

Remark 1. Assumption 1 is to avoid degenerate cases that arise with deterministic relationships. For instance, suppose **X** is a random process with a continuous distribution and **Y** represents **X** passed through a deterministic invertible system. Then $P_{\mathbf{X},\mathbf{Y}}$ is not positive since the distribution of **Y** given **X** is a point mass. Moreover, this assumption holds for any continuous-time generative model described by coupled stochastic differential equations such as the one presented in example 1 corresponding to a system, which is both strictly causal and nondegenerate.

Denote the Kullback-Leibler divergence for $P_{\underline{\mathbf{X}}}$ and $Q_{\underline{\mathbf{X}}}$ as

$$D(P_{\underline{\mathbf{X}}}||Q_{\underline{\mathbf{X}}}) := \mathbb{E}_{P_{\underline{\mathbf{X}}}}\left[\log \frac{dP_{\underline{\mathbf{X}}}}{dQ_{\underline{\mathbf{X}}}}\right],$$

where $\frac{dP_{\underline{\mathbf{X}}}}{dQ_{\underline{\mathbf{X}}}}$ denotes the Radon-Nikodym derivative. Consider two random processes $\mathbf{X}_i$ and $\mathbf{X}_j$ and a set of indices $\mathcal{K}$ such that $\mathcal{K} \subseteq -\{i, j\}$, then entropy, the mutual information, and the conditional mutual information between $\mathbf{X}_i$ and $\mathbf{X}_j$ are given, respectively, by

$$H(\mathbf{X}_i) := \mathbb{E}_{P_{\mathbf{X}_i}}\left[-\log P_{\mathbf{X}_i}\right],$$

$$I(\mathbf{X}_j; \mathbf{X}_i) := \mathbb{E}_{P_{\mathbf{X}_j,\mathbf{X}_i}}\left[\log \frac{dP_{\mathbf{X}_i|\mathbf{X}_j}}{dP_{\mathbf{X}_i}}\right], \tag{3.6}$$

$$I(\mathbf{X}_j; \mathbf{X}_i|\underline{\mathbf{X}}_{\mathcal{K}}) := \mathbb{E}_{P_{\underline{\mathbf{X}}_{\mathcal{K}\cup\{i,j\}}}}\left[\log \frac{dP_{\mathbf{X}_i|\mathbf{X}_j,\underline{\mathbf{X}}_{\mathcal{K}}}}{dP_{\mathbf{X}_i|\underline{\mathbf{X}}_{\mathcal{K}}}}\right]. \tag{3.7}$$

Analogously, directed information and conditional directed information from $\mathbf{X}_j$ to $\mathbf{X}_i$ are defined, respectively, as

$$I(\mathbf{X}_j \to \mathbf{X}_i) := \mathbb{E}_{P_{\mathbf{X}_j,\mathbf{X}_i}}\left[\log \frac{dP_{\mathbf{X}_i||\mathbf{X}_j}}{dP_{\mathbf{X}_i}}\right], \tag{3.8}$$

$$I(\mathbf{X}_j \to \mathbf{X}_i||\underline{\mathbf{X}}_{\mathcal{K}}) := \mathbb{E}_{P_{\underline{\mathbf{X}}_{\mathcal{K}\cup\{i,j\}}}}\left[\log \frac{dP_{\mathbf{X}_i||\mathbf{X}_j,\underline{\mathbf{X}}_{\mathcal{K}}}}{dP_{\mathbf{X}_i||\underline{\mathbf{X}}_{\mathcal{K}}}}\right]. \tag{3.9}$$

Consequently, the directed information rate and the conditional directed information rate are defined, respectively, as

$$I_\infty(\mathbf{X}_j \to \mathbf{X}_i) := \lim_{t\to-\infty} \frac{1}{n-t+1} I(\mathbf{X}_{j,t}^n \to \mathbf{X}_{i,t}^n),$$

$$I_\infty(\mathbf{X}_j \to \mathbf{X}_i||\underline{\mathbf{X}}_{\mathcal{J}}) := \lim_{t\to-\infty} \frac{1}{n-t+1} I(\mathbf{X}_{j,t}^n \to \mathbf{X}_{i,t}^n||\underline{\mathbf{X}}_{\mathcal{J},t}^n).$$

Since in this work, the lengths of processes are assumed to be finite, $n < \infty$, the directed information and conditional directed information are finite. Thus, it suffices to work with equations 3.8 and 3.9. If $n \to \infty$, the same proof ideas hold by replacing equations 3.8 and 3.9 with the aforementioned information rates instead.

3.2 Generative Models and Directed Information Graphs. A directed graph $\overrightarrow{G} = (V, \overrightarrow{E})$ is characterized by a set V of vertices (or nodes) and a set of ordered pairs of vertices, called arrows (or edges) $\overrightarrow{E} \subset V \times V$. An undirected graph is called *connected* if there is at least one path between any two nodes; if there is exactly one path between any pair of vertices, then it is called a *tree*. A *polytree* denoted by $\overrightarrow{T} = (V, \overrightarrow{E})$ is a directed acyclic graph (DAG) whose underlying undirected graph, obtained by replacing all arrows with undirected edges, is a tree. Nodes without any incoming arrows in a directed tree are called *roots*. A *path* between two nodes in an undirected graph is a sequence of distinct vertices such that every vertex in the sequence is adjacent to its predecessor and its successor; all nodes except the end nodes on a path are called *internal nodes*. Two paths are called *disjoint* if they do not have any internal vertex in common. A path of the from $v \to \cdots \to u$, on which every edge is an arrow with the arrowheads pointing toward u, is a *directed path* form v to u. The sets of *parents* and *children* of a node v in $\overrightarrow{T}$ are defined, respectively, by

$$\begin{aligned} \mathcal{PA}(v) &:= \{u \in V : (u, v) \in \overrightarrow{E}\}, \\ \mathcal{CH}(v) &:= \{u \in V : (v, u) \in \overrightarrow{E}\}. \end{aligned} \tag{3.10}$$

Node w is called an *ancestor* of node v in $\overrightarrow{T}$ if there exists a directed path from w to v. In this case, v is called a *descendant* of w.

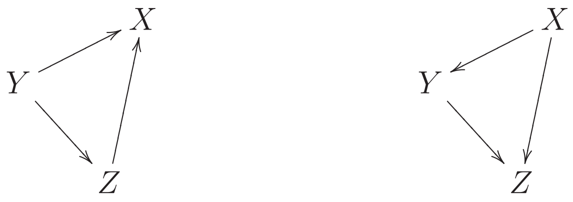
Definition 1. *For a joint distribution a generative model is a function* $A : \{1, ..m\} \to \mathcal{P}(\{1, ..m\})$, *(power set of* $\{1, ..m\}$*) such that for each process* $j \in \{1, ..m\}$, $j \notin A(j)$ *and*

$$D(P_{\underline{\mathbf{X}}} || \, P_A) = 0 \ ,$$

where $P_A := \prod_{j=1}^{m} P_{\mathbf{X}_j || \, \underline{\mathbf{X}}_{A(j)}}$.

Definition 2. *A generative model graph is a directed graph where each node corresponds to a random process, and there is an arrow from i to j, for* $i, j \in \{1, ..\}$ *if and only if* $i \in A(j)$. *It is called minimal if for each i,* $A(i)$ *has minimal cardinality.*

A generative model graph is basically a graphical representation of the factorization of a given joint distribution. Although there might exist several factorizations for a joint distribution, under assumption 1, the minimal factorization (i.e., generative model) is unique (for details, see Quinn et al., 2015). This graphical model is similar to the Bayesian network (Pearl, 1988), since both models depend on the factorization of the joint distribution. The main difference between the two models is that nodes in a Bayesian



$(a)P_{X,Y,Z} = P_Y P_{Z|Y} P_{X|Y,Z}.$ $(b)P_{X,Y,Z} = P_X P_{Y|X} P_{Z|X,Y}.$

Figure 2: Two possible Bayesian networks for one joint distribution.

network represent random variables, but in a generative model graph, they are random processes.

Definition 3. *A directed information graph is a directed graph over a set of random processes $\underline{\mathbf{X}}$ where there is an arrow from i to j for $i, j \in \{1, \ldots, m\}$ if and only if*

$$I(\mathbf{X}_i \to \mathbf{X}_j || \underline{\mathbf{X}}_{-\{i,j\}}) > 0\ . \tag{3.11}$$

Theorem 1 *(Quinn, Kiyavash, & Coleman, 2011). For any joint distribution $P_{\underline{\mathbf{X}}}$ satisfying assumption 1, the corresponding minimal generative model graph and directed information graph are equivalent.*

In the remainder of this article, we refer to generative model graphs and directed information graphs interchangeably.

3.3 Bayesian Networks and Directed Information Graphs.

Bayesian networks are directed graphs representing conditional dependencies in a reduced factorization of the joint distribution. Hence, Bayesian networks depend on the order variables. Figure 2 shows two possible Bayesian networks pertaining to $P_{X,Y,Z}$.

Note that the Bayesian networks are DAGs, since a variable can have only an incoming arrow from the preceding variables. Therefore, in general, DIGs are not in the family of Bayesian networks. However, DIGs and the Bayesian networks share some similar properties, which we review next.

D-separation, introduced in Verma and Pearl (1991), is a set of graphical conditions by which conditional independence relations could be read from a DAG (e.g., a Bayesian network). D-separation has the following implication: If two sets of nodes U and W are d-separated in a DAG by a third set Z (excluding U and W), the corresponding variable sets X_U and X_W are independent given the variables in X_Z:

$$I(X_U; X_W | X_Z) = 0.$$

These independence relations comprise the global Markov property. Before defining d-separation in DAGs, we introduce the concept of a collider. In a DAG, a non-endpoint vertex c on a path is said to be a *collider* if both edges are directed toward c on this path. For example, X in Figure 2a is a collider on the path $Y \rightarrow X \leftarrow Z$.

Definition 4. *Let $\overrightarrow{G} = (V, \overrightarrow{E})$ be a DAG and U, W, and Z be three disjoint subsets of V. Z d-separates U from W, if for every path (not necessarily directed) from a node in U to a node in W, there exists a node c such that either*

1. *c is not a collider and it belongs to Z or*
2. *c is a collider and neither c nor any of c's descendants are in Z.*

Remark 2. It is possible that two DAGs, $\overrightarrow{G}_1$ and $\overrightarrow{G}_2$ with the same vertex set capture the same independence relations, that is, for all disjoint sets U, W, and Z, where U and W are nonempty, Z d-separates U from W in $\overrightarrow{G}_1$ if and only if Z d-separates U from W in $\overrightarrow{G}_2$. In this case, it is said that $\overrightarrow{G}_1$ and $\overrightarrow{G}_2$ are Markov equivalent. For example, two DAGs in Figure 2 are Markov equivalent. Ali, Richardson, and Spirtes (2009) give simple conditions for determining whether two DAGs are Markov equivalent.

Analogously, the causal independences in a DIG can be determined through a graphical separation criterion that we call *c-separation*.

Definition 5. *Let $\overrightarrow{G} = (V, \overrightarrow{E})$ be a DIG and U and Z be two disjoint subsets of V, and $w \in V \setminus (U \cup Z)$. Z c-separates U from w if for every path between a node in U and w there is a node in $Z \cup w$ with an outgoing arrow.*

For example, in Figure 3, **Z** c-separates **U** from **W**. Notice that c-separation, unlike d-separation, is not symmetric; if Z c-separates U from W, it is not necessary that Z c-separates W from U. A directed graph is said to satisfy a global causal Markov property if each c-separation corresponds to a causal independences. In other words, if there exist three disjoint subsets U, $\{w\}$ and Z such that Z c-separates U from w, the corresponding process sets $\underline{\mathbf{X}}_U$ and $\mathbf{X}_w$ are causally independent given the processes in $\underline{\mathbf{X}}_Z$:

$$I(\underline{\mathbf{X}}_U \rightarrow \mathbf{X}_w || \underline{\mathbf{X}}_Z) = 0.$$

Theorem 2. *For any joint distribution $P_{\underline{\mathbf{X}}}$ that satisfies assumption 1, the DIG is a minimal directed graph with global causal Markov property.*

Proof. See appendix A.

Next, we study the relationship between the DIG of a set of random processes and the independence map among the underlying random variables.

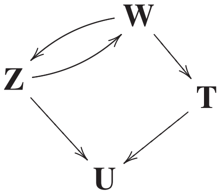


Figure 3: An example of DIG with four random processes.

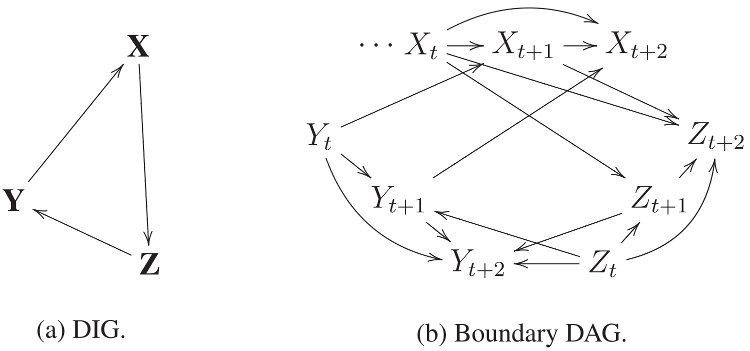


Figure 4: The DIG and its underlying variable dependences.

Let V be a network of dependent variables and σ be an ordering $\{v_1, \ldots, v_m\}$ of the elements in V. The boundary strata of this network relative to σ is an ordered set of subsets of V, $(B_1, B_2 \ldots)$, such that each B_i is a Markov boundary of v_i with respect to the set $V_i := \{v_1, \ldots, v_{i-1}\}$; that is B_i is a minimal set satisfying $B_i \subseteq V_i$ and v_i is independent of $V_i \setminus B_i$ given B_i. The DAG created by designating each B_i as parents of vertex v_i is called a boundary DAG of this network relative to σ. By Verma and Pearl (1988), boundary DAGs are Bayesian networks (minimal independence maps under d-separation).

A simple observation is that due to the nature of random processes there already exists an ordering among the underlying variables, which is time. Hence, if $\underline{\mathbf{X}}$ is a set of random processes that satisfies assumption 1 with the corresponding minimal generative model graph $\overrightarrow{G}$, then one can define a unique boundary DAG for the underlying variables relative to time ordering. Notice that the boundary DAG relative to time ordering is unique since there are no simultaneous influences between variables and, therefore, any causal ordering results in the same DAG. Now, by the definition of a minimal generative model graph, the Markov boundary of the tth variable in process $\mathbf{X}_i$ contains $X_{j,t'}, t' < t$ if and only if $\mathbf{X}_j$ is a parent of $\mathbf{X}_i$ in $\overrightarrow{G}$ or equivalently by theorem 1 in the corresponding DIG. For example, in Figure 4, Y_t is in the Markov boundary of X_{t+1}; hence, $\mathbf{Y}$ must be a parent of $\mathbf{X}$ in the corresponding DIG.

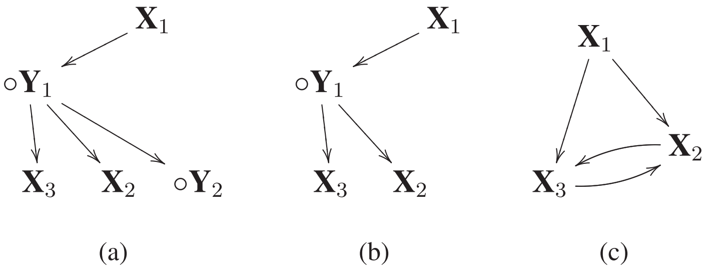


Figure 5: The DIGs of example 3. (a) The DIT corresponding to $P_{\underline{\mathbf{X}}}$. (b) The DIT corresponding to $P_{\underline{\mathbf{X}}\setminus\{\mathbf{Y}_2\}}$. (c) The DIG corresponding to $P_{\underline{\mathbf{O}}}$. Latent nodes are indicated by circles.

4 Minimal Latent Tree

Consider a set of random processes $\underline{\mathbf{X}}$ for which the directed information graph is a polytree $\overrightarrow{T} = (V, \overrightarrow{E})$, abbreviated as DIT. Denote $\underline{\mathbf{O}} = \{\mathbf{X}_1, \ldots, \mathbf{X}_m\}$ as the set of observable processes, and their corresponding nodes in the DIT is denoted by O. Likewise, denote $\underline{\mathbf{L}} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_k\}$ as the set of latent processes, and their corresponding nodes are denoted by L. Briefly, $\underline{\mathbf{X}} = \underline{\mathbf{O}} \cup \underline{\mathbf{L}}$ is the set of random processes, and $V = O \cup L$ is their corresponding nodes in the DIT.

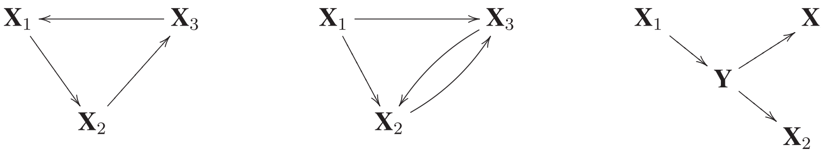
A probability distribution $P_{\underline{\mathbf{O}}}$ is called *polytree decomposable* if there exists a joint distribution of the form $P_{\underline{\mathbf{O}}\cup\underline{\mathbf{L}}}$ that satisfies assumption 1 and its corresponding DIG is a polytree. In this case, $P_{\underline{\mathbf{O}}\cup\underline{\mathbf{L}}}$ is called a polytree extension of $P_{\underline{\mathbf{O}}}$.

Example 3. Consider an array of five random processes $\underline{\mathbf{X}} = (\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \mathbf{Y}_1, \mathbf{Y}_2)$ with the joint dynamics

$$\underline{\mathbf{X}}(t) = \underline{\mathbf{X}}(t-1)\mathbf{A} + \underline{\mathbf{X}}(t-2)\mathbf{B} + \underline{\mathbf{W}}(t),$$

where $\underline{\mathbf{X}}(t)$ is the row vector $(X_{1,t}, X_{2,t}, X_{3,t}, Y_{1,t}, Y_{2,t})$, and $\mathbf{A}$ and $\mathbf{B}$ are 5×5 real matrices such that their nonzero entries are $\mathbf{A}(4,2)$, $\mathbf{A}(1,4)$, $\mathbf{A}(4,5)$, and $\mathbf{B}(4,3)$ and they are all equal to one. $\underline{\mathbf{W}}$ is a set of five jointly independent random processes. Figure 5a illustrates the corresponding DIG of the whole system. Figures 5b and 5c are obtained by marginalizing over $\mathbf{Y}_2$ and $\{\mathbf{Y}_1, \mathbf{Y}_2\}$, respectively. Since there exists at least one joint distribution such that its corresponding DIG has a polytree structure, $P_{\underline{\mathbf{O}}}$ is polytree decomposable, where $O = \{\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3\}$.

A latent node $h \in L$ is called redundant if the DIG corresponding to the joint distribution of observed and latent nodes excluding $\mathbf{Y}_h$, $(P_{\underline{\mathbf{O}},\underline{\mathbf{L}}\setminus\{\mathbf{Y}_h\}})$ remains a forest, that is, a collection of polytrees. For instance, in example 3,



(a) The DIG of equation 4.1

(b) The DIG of equation 4.2

(c) The corresponding DIT of equation 4.2

Figure 6: Directed information graphs of example 4.2.

Y_2 is a redundant hidden node. A latent directed information polytree (LDIT) is called minimal if it has no redundant hidden nodes (Pearl, 1988).[5] The polytree in Figure 3b is minimal.

Assumption 2. *We assume that the joint distribution of the set of observed processes is polytree decomposable.*

The next example demonstrates cases in which one is polytree-decomposable and the other is not.

Example 4. Consider a set of three observable processes $\underline{\mathbf{X}}$ comprising a physical, dynamical system, such that the evolution of the processes over time satisfies the following stochastic equations:

$$\begin{aligned} X_1(t) &= X_3(t-1)/3 + V_1(t), \\ X_2(t) &= X_1(t-1)/2 + V_2(t), \\ X_3(t) &= X_2(t-1)/2 + V_3(t), \end{aligned} \tag{4.1}$$

where (V_1, V_2, V_3) are three exogenous, independent processes. Figure 6a demonstrates the corresponding DIG. For this small example, by checking all possible sets of auxiliary variables, we can conclude that there is no set of auxiliary variables $\underline{\mathbf{L}}$ such that $P_{\underline{\mathbf{X}}\cup\underline{\mathbf{L}}}$ both satisfies assumption 1 and its corresponding DIG is a polytree. Now, consider the following discrete-time dynamical system with the corresponding DIG shown in Figure 6b:

$$\begin{aligned} X_1(t) &= V_1(t), \\ X_2(t) &= X_1(t-2)/2 + V_4(t-1)/2 + V_2(t), \\ X_3(t) &= X_1(t-2)/3 + V_4(t-1)/3 + V_3(t), \end{aligned} \tag{4.2}$$

[5] A redundant hidden node in Pearl (1988) is defined as a hidden node that the joint distribution without it remains a tree instead of a forest.

where (V_1, V_2, V_3, V_4) are exogenous, independent processes. By defining $Y(t) := X_1(t-1) + V_4(t)$, we can obtain a DIT as shown in Figure 6c.

4.1. Some Properties of a Minimal LDIT. This section presents some properties of the DIT and the minimal LDIT, which will be used in section 5 for structure learning.

Lemma 1. *Let* $\overrightarrow{T} = (V, \overrightarrow{E})$ *be the DIT corresponding to the joint distribution of a collection of random processes* $\underline{\mathbf{X}}$*. Let* $\mathbf{X} \in \underline{\mathbf{X}}$ *and* $\mathcal{A}_1$ *and* $\mathcal{A}_2$ *be two disjoint subsets of the parents of* $\mathbf{X}$*, that is,* $\mathcal{PA}(\mathbf{X})$*. Then* $\underline{\mathbf{X}}_{\mathcal{A}_1}$ *and* $\underline{\mathbf{X}}_{\mathcal{A}_2}$ *are independent.*

Proof. See appendix B.

Lemma 2. *In a minimal LDIT, all hidden nodes have at least two children.*

Proof. See appendix C.

Lemma 3. *Consider a collection of random processes* $\underline{\mathbf{X}}$ *with a DIT* $T = (V, \overrightarrow{E})$*. If there is a directed path from j to i of length d, that is, there is a sequence of nodes* $(i_1, \ldots, i_{d-1})$ *where j is the parent of* i_1*,* i_k *is the parent of* i_{k+1} *for* $(1 \leq k \leq d-2)$*, and* i_{d-1} *is the parent of i, then*

$$D(P_{\mathbf{X}_i|\mathbf{X}_j} \,||\, P_{\mathbf{X}_i||_d\mathbf{X}_j}) = 0\,. \tag{4.3}$$

Proof. See appendix D.

Lemma 3 implies that by walking along the path between two random process $\mathbf{X}_i$ and $\mathbf{X}_j$, each time we pass a node, the time dependency between $\mathbf{X}_i$ and $\mathbf{X}_j$ is shifted by at least one unit. In the next sections, we will see that these time delays will help us recover the structure of a minimal LDIT. Time delays have also been used for infernce tasks in network forensic applications such as traffic analysis (Shmatikov & Wang, 2006; Kiyavash & Coleman, 2009; Kadloor, Kiyavash, & Venkitasubramaniam, 2012a, 2012b).

Lemma 4. *Suppose there exist two disjoint directed paths from* $\mathbf{W}$ *to* $\mathbf{X}$ *and* $\mathbf{Y}$ *in a minimal LDIT. Then*

$$D\left(P_{\mathbf{X},\mathbf{Y},\mathbf{W}} || P_{\mathbf{W}} P_{\mathbf{X}||\mathbf{W}} P_{\mathbf{Y}||\mathbf{W}}\right) = 0. \tag{4.4}$$

Proof. See appendix E.

Lemma 5. *In a minimal LDIT, if the root ancestors of two nodes are disjoint, then they are independent.*[6]

Proof. See appendix F.

Another property that plays an essential role in learning the latent structure is what we call *sibling resemblance*.

Definition 6. *A collection of random processes $\underline{\mathbf{X}}$ with a corresponding minimal LDIT, $\overrightarrow{T} = (V, \overrightarrow{E})$, satisfies sibling resemblance property, if for every pair $(\mathbf{X}_i, \mathbf{X}_j)$, $(i \neq j)$, of sibling with common parent $\mathbf{X}_k$, the following property holds: If there exists a time s such that $I(X_{i,1}^{s}; \mathbf{X}_k) > 0$, then $I(X_{i,s}; \mathbf{X}_j | X_{i,1}^{s-1}) > 0$*

This property simply states that in a minimal LDIT, the information inherited from a node to its children is not independent. Many dynamical systems such as autoregressive models satisfy this property. The next example illustrates the importance of this property for learning latent polytrees.

Example 5. Consider a minimum LDIT with two observable and one latent random processes denoted by $\underline{\mathbf{X}} = \{\mathbf{X}_1, \mathbf{X}_2, \mathbf{Y}_1\}$. Let $X_{1,t+1} = 2Y_{1,2t-1} + \epsilon_{1,t+1}$ and $X_{2,t+1} = -Y_{1,2t} + \epsilon_{2,t+1}$, where $\epsilon_{1,t}$, $\epsilon_{2,t}$, and $\mathbf{Y}_1$ are jointly independent. The corresponding DIG of this system is $\mathbf{X}_1 \leftarrow \mathbf{Y}_1 \rightarrow \mathbf{X}_2$. Suppose that $\{Y_{1,2t}\}$ and $\{Y_{1,2t-1}\}$, that is, the even and odd subprocesses of $\mathbf{Y}_1$ are independent. In this case, $\mathbf{X}_1$ and $\mathbf{X}_2$ are independent, and recovering the structure of the system given $\{\mathbf{X}_1, \mathbf{X}_2\}$ is impossible. This system does not satisfy the sibling resemblance property since $\mathbf{X}_1$ and $\mathbf{X}_2$ are siblings with $\mathbf{Y}_1$ as their common parent and $I(X_1^2; \mathbf{Y}_1) > 0$, $(s = 2)$, but $I(X_{1,2}; \mathbf{X}_2 | X_{1,1}) = 0$.

4.2. Presence of Simultaneous Influences.

Excluding simultaneous influences helps us write equation 3.1, which consequently leads to the definition of generative model graphs in section 3.2. Now the question is, What if there were in fact simultaneous influences?

In this section, we show that if there are simultaneous influences between processes, the corresponding DIG is not a polytree and hence cannot be recovered by the proposed method in this article. To make the statement rigorous, we need to modify the definition of the DIG by using the original Kramer's causal conditioning that allows for simultaneous influences. For $\mathcal{K} \subseteq -\{j\}$, define

$$\widetilde{P}_{\mathbf{X}_j \| \underline{\mathbf{X}}_{\mathcal{K}}} := \prod_{t=1}^{n} P_{X_{j,t} | X_{j,1}^{t-1}, \underline{X}_{\mathcal{K},1}^{t}}$$

[6] The set of roots that are ancestors of a given node in a directed tree is called root ancestors of that node.

and the modified conditional directed information as

$$\widetilde{I}\left(\mathbf{X}_j \to \mathbf{X}_i || \underline{\mathbf{X}}_{\mathcal{K}}\right) := \mathbb{E}_{P_{\underline{\mathbf{X}}_{\mathcal{K}\cup\{i,j\}}}} \left[\log \frac{d\widetilde{P}_{\mathbf{X}_i || \mathbf{X}_j, \underline{\mathbf{X}}_{\mathcal{K}}}}{d\widetilde{P}_{\mathbf{X}_i || \underline{\mathbf{X}}_{\mathcal{K}}}} \right].$$

Using the above measure, we are able to define the modified directed information graph (MDIG) that captures the simultaneous effects as such: there is an arrow from node i to node j for $i, j \in \{1, \ldots, m\}$ in the MDIG if and only if

$$\widetilde{I}\left(\mathbf{X}_i \to \mathbf{X}_j || \, \underline{\mathbf{X}}_{-\{i,j\}}\right) > 0 \, .$$

Theorem 3. *Let $\overrightarrow{T}$ to be an MDIG over a set of random processes $\underline{\mathbf{X}}$, which is a polytree, and let $\mathcal{PA}(\mathbf{X})$ to be the parent set of $\mathbf{X}$ in $\overrightarrow{T}$. Then*

$$D\left(\widetilde{P}_{\mathbf{X}|| \, \underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{X})}} || P_{\mathbf{X}|| \, \underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{X})}}\right) = 0.$$

Proof. See appendix G.

A consequence of the above result is that the corresponding DIG of a system with simultaneous influences is not a polytree. This is because when the corresponding MDIG of a dynamical system is a polytree, based on the above result, all the simultaneous influences can be dropped.

5. Recovery of Latent Trees

A simple observation about a directed tree is that each pair of nodes that are the descendants of the same root has a unique common ancestor. In this section, we define a notion of distance on a polytree in order to determine the distance of each pair of nodes to their common ancestor, if it exists. Moreover, we will show that given these distances for a subset of nodes, the graph is uniquely recoverable.

Definition 7. *Given a polytree $\overrightarrow{T} = (V, \overrightarrow{E})$ with the root set $\mathcal{R}$, every function $\gamma : V \times V \to \mathbb{R}$ that satisfies the following criterion is called a discrepancy on $\overrightarrow{T}$. γ assigns a real number to the path from v_1 to the common ancestor of v_1 and v_2, such that:*

1. *$\gamma(v_1, v_2) = 0$ iff either v_1 is the ancestor of v_2 or $v_1 = v_2$.*
2. *If the common ancestor of v_1 and v_2 is the same as the common ancestor of v_1 and v_3, then*

$$\gamma(v_1, v_2) = \gamma(v_1, v_3).$$

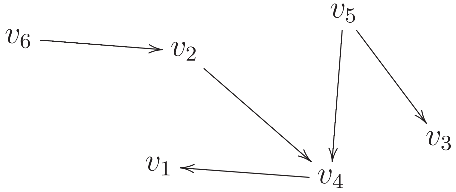


Figure 7: The directed tree of example 6.

3. *If the common ancestor of v_1 and v_2 is on the path from the common ancestor of v_1 and v_3 to v_1, then*

$$\gamma(v_1, v_2) < \gamma(v_1, v_3).$$

4. $\gamma(v_1, v_2) < 0$ *iff v_1 and v_2 have no common ancestor.*

The image of such these functions can be presented by the discrepancy matrix:

$$\Gamma_V := \left[\gamma_r(v_i, v_j)\right], \quad v_i, v_j \in V.$$

Note that for a given tree, the discrepancy matrix is not unique. Any function that satisfies the conditions in definition 7 is a valid discrepancy measure.

Example 6. Consider the polytree depicted in Figure 7 with roots $\{v_5, v_6\}$ and the following discrepancy matrix:

$$\Gamma_V = \begin{pmatrix} 0 & 2 & 3 & 1 & 3 & 4 \\ 0 & 0 & -2 & 0 & -1 & 1 \\ 1 & -3 & 0 & 1 & 1 & -3 \\ 0 & 1 & 2 & 0 & 2 & 3 \\ 0 & -1 & 0 & 0 & 0 & -2 \\ 0 & 0 & -1 & 0 & -1 & 0 \end{pmatrix}.$$

For instance, looking at the third row, this particular discrepancy function assigns 1 to the path from v_3 to its common ancestor with v_1, that is, v_5. Since v_2 and v_3 have no common ancestor, $\Gamma_V(3, 2) < 0$.

We prove that the discrepancy matrix suffices to uniquely learn the topology of a polytree $\overrightarrow{T} = (V, \overrightarrow{E})$. We present an algorithm that learns the structure of a polytree given the discrepancies between all the pairs of observed nodes.

Definition 8. *In a polytree $\overrightarrow{T} = (V, \overrightarrow{E})$, we call a subset $L \subset V$ learnable if every node $v \in L$ has at least two outgoing arrows. We call $O := V \setminus L$ the set of observed nodes.*

Algorithm 1 : **Separation**(Γ_O).

```
Input : $\Gamma_O$
Output : $O_1, ..., O_{|\mathcal{R}|}$
$M \leftarrow \emptyset, i \leftarrow 1$
while $O \setminus M \neq \emptyset$ do
    Choose $v$ in $O \setminus M$
    Find all $\mathcal{C} \subseteq O$ such that $v \in \mathcal{C}$ and
    for all $(u, w) \in \mathcal{C} \times \mathcal{C}, \gamma(u, w) \geq 0$.
    $O_i \leftarrow$maximal $\mathcal{C}$
    Return $O_i$
    $M \leftarrow M \cup O_i$
    $i \leftarrow i + 1$
end while
```

For example, $\{v_5\}$ is a learnable subset of the polytree shown in Figure 7. From definition 8, if L is a learnable subset of a polytree, then all the leaves belong to $O = V \setminus L$.

Theorem 4. *Let $\overrightarrow{T} = (V, \overrightarrow{E})$ be a polytree with the root set $\mathcal{R}$, and let $L \subseteq V$ be a learnable subset. Then the existence of a discrepancy matrix Γ_O for $O = V \setminus L$ suffices for learning $\overrightarrow{T}$.*

Proof. See appendix H.

Next, inspired by the steps in the proof of theorem 4, we present an algorithm for structure learning of polytrees.

5.1. Structure Recovery Algorithm.

Here, we introduce an algorithm to learn a latent polytree. The rationale of the algorithm follows the three main steps of proof of theorem 4 as described in the following, The first step is to discover the number of roots $|\mathcal{R}|$ of the underlying polytree and all their descendants in the set of observed nodes (O) given the discrepancy matrix Γ_O. This can be done by fixing a node $v \in O$ and finding a maximal subset of O containing v in which every pair of nodes has positive discrepancy (see algorithm 1). The next step is to recover the underlying

Algorithm 2 : **Tree**(O).

```
$Input: \ \Gamma_O$
$Output: \ \overrightarrow{T} = (V, \overrightarrow{E})$
For all $v \in O$
$B_v \leftarrow \arg\min_{u \in O \setminus \{v\}} \gamma(v, u)$
if $B_v = O \setminus \{v\} \ \forall v \in O$ then
    if $\exists w \in O : \min_{u \in O \setminus \{w\}} \gamma(w, u) = 0$ then
        $\overrightarrow{T}$ is a star graph with $w$ as the root in the center.
    else
        $\overrightarrow{T}$ is a star graph with a hidden node as the root in the center.
    end if
else
    Choose $w$ such that $B_w \neq O \setminus \{w\}$
    $\overrightarrow{T}' \leftarrow$Tree$(B_w \cup \{w\})$
    $\overrightarrow{T}'' \leftarrow$Tree$(O \setminus B_w)$
    Substitute $w$ in $\overrightarrow{T}''$ by another node, say $h$.
    $\overrightarrow{T} \leftarrow \overrightarrow{T}' \oplus \overrightarrow{T}''(h)$
end if
```

tree for every root $r \in \mathcal{R}$ given its discovered descendants in the set O. This can be done using the recursive approach summarized in algorithm 2. The last step is to merge the recovered trees from the previous step to recover the underlying polytree. This too is possible, since if two recovered trees are connected, their common subgraph is also a tree; thus, it can be learned using algorithm 2. Algorithm 3 describes the required steps.

Next, we present our algorithm that learns a polytree given a discrepancy matrix on its observed nodes using the three main steps noted. A simple

Algorithm 3 : **Polytree**(Γ_O).

```
$Input: \ \Gamma_O$
$Output: \ \overrightarrow{T} = (V, \overrightarrow{E})$
**Separation**$(\Gamma_O)$.
$\overrightarrow{T} \leftarrow$**Tree**$(O_1)$
$\mathcal{S} \leftarrow O_1, \mathcal{I} \leftarrow \{1\}$
**while** $\mathcal{I} \neq \{1, 2, ..., |\mathcal{R}|\}$ **do**
    Find $i \in \{1, 2, ..., |\mathcal{R}|\} \setminus \mathcal{I}$ such that $O_i \cap \mathcal{S} \neq \emptyset$
    $\overrightarrow{T}_{sub} \leftarrow$**Tree**$(\mathcal{S} \cap O_i)$
    $\overrightarrow{T}_i \leftarrow$**Tree**$(O_i)$
    $\overrightarrow{T} \leftarrow \overrightarrow{T} \circ \overrightarrow{T}_i|_{\overrightarrow{T}_{sub}}$
    $\mathcal{S} \leftarrow \mathcal{S} \cup O_i$
    $\mathcal{I} \leftarrow \mathcal{I} \cup \{i\}$
**end while**
```

example that illustrates the algorithm is also provided. First, we need the following definition:

Definition 9. *A tree merger is an operator that takes two directed trees* $\overrightarrow{T}_1$, $\overrightarrow{T}_2$ *and a given subtree of both of them, say,* $\overrightarrow{T}_3$*, and merges them at* $\overrightarrow{T}_3$*. We denote this operation by* $\overrightarrow{T}_1 \circ \overrightarrow{T}_2|_{\overrightarrow{T}_3}$ *.*

Figure 8 depicts one such tree merger.

Polytree (Γ_O) presents an algorithm for learning the polytree $\overrightarrow{T}(V, \overrightarrow{E})$ with the root set $\mathcal{R}$ given the discrepancy matrix Γ_O on its observed nodes O. First, it calls the subroutine **Separation**(Γ_O), which finds subsets O_is, where $O = \cup_i O_i$ such that each subset corresponds to observed nodes in a directed tree with a single root. Each of these single rooted subtrees can be learned by algorithm **Tree**(O). To complete the task, algorithm **Polytree**(Γ_O) must connect these subtrees to recover the original polytree. This is done by using the fact that if a polytree $\overrightarrow{T}$ and a directed tree $\overrightarrow{T}_i$ have an intersection, their

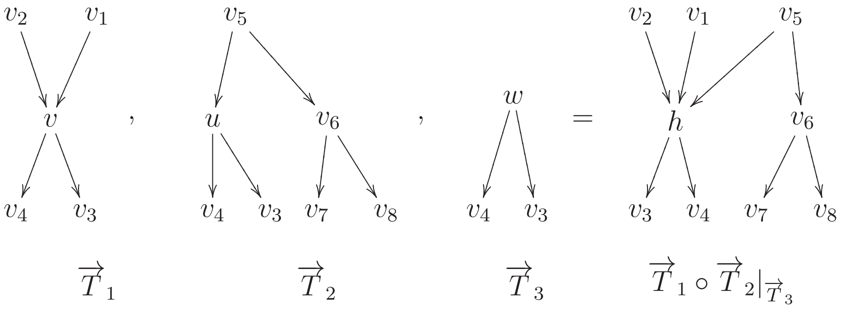


Figure 8: An example that illustrates the merger operator between two directed trees.

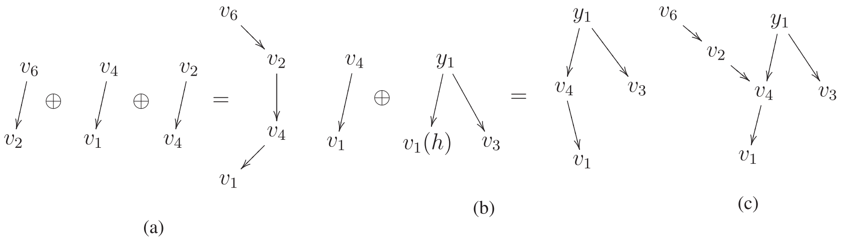


Figure 9: (a) The steps and outputs of **Tree**($\{v_1, v_2, v_4, v_6\}$). (b) The steps and outputs of **Tree**($\{v_1, v_3, v_4\}$). (c) Merging the first two directed trees by sharing their common subtree, which is obtained by **Tree**($\{v_1, v_4\}$).

intersection will be a directed tree. Thus, it also could be learned by algorithm **Tree**(O). After learning the intersection part, algorithm **Polytree**(Γ_O) uses the tree merger operator defined in definition 9 to connect these together. In Algorithm **Tree**(O), $\overrightarrow{T}_1 \oplus \overrightarrow{T}_2(h)$ is an operator that connects a directed tree $\overrightarrow{T}_1 = (V_1, \overrightarrow{E}_1)$ with root r_1 to a polytree $\overrightarrow{T}_2 = (V_2, \overrightarrow{E}_2)$ given a leaf of $\overrightarrow{T}_2$, h, by simply substituting h in $\overrightarrow{T}_2$ by $\overrightarrow{T}_1$. More precisely,

$$\overrightarrow{T}_1 \oplus \overrightarrow{T}_2(h) := \left(V_1 \cup V_2 \setminus \{h\}, \overrightarrow{E}\right),$$

where

$$\overrightarrow{E} = \overrightarrow{E}_1 \cup \{(\mathcal{PA}_2(h), r_1)\} \cup \overrightarrow{E}_2 \setminus \{(\mathcal{PA}_2(h), h)\},$$

and $\mathcal{PA}_2(h)$ is given by equation 3.10, and it represents the set of parents of h in $\overrightarrow{T}_2$. Figure 9b depicts an example.

Example 7. Consider the polytree in example 6. Assume $O = \{v_1, v_2, v_3, v_4, v_6\}$. Then, by the definition $V \setminus O = \{v_5\}$ is a learnable subset. Given the discrepancy matrix

$$\Gamma_O = \begin{pmatrix} 0 & 2 & 3 & 1 & 4 \\ 0 & 0 & -2 & 0 & 1 \\ 1 & -3 & 0 & 1 & -3 \\ 0 & 1 & 2 & 0 & 3 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix},$$

algorithm 3 calls **Separation** to find all subtrees with single roots, which are $O_1 = \{v_1, v_2, v_4, v_6\}$ and $O_2 = \{v_1, v_3, v_4\}$. As seen in Figure 7, the subtrees induced by O_1 and O_2 each have a single root.

Subsequently, algorithm 3 calls **Tree** to build the subtrees. Figures 9a and 9b illustrate these subtrees. For instance, the subtree in Figure 9a is obtained as follows: Algorithm 2 computes B_{v_i}s for $i \in \{1, 2, 4, 6\}$ at step 4. Since $B_{v_2} = \{v_1, v_4\} \neq O_1 \setminus \{v_2\}$, the condition in step 5 is not satisfied and algorithm 2 will jump to step 12 and choose w to be v_2. In steps 13 and 14, the algorithm recursively calls itself, but this time given $\{v_1, v_2, v_4\}$ and $\{v_2, v_6\}$, respectively. Since the subtree induced by $\{v_2, v_6\}$ is a star, it will be constructed in steps 5 to 10. On the other hand, the subtree induced by $\{v_1, v_2, v_4\}$ is not a star. It is learned by breaking it into two stars, as shown in Figure 9a.

Finally, algorithm 3 must reconnect the subtrees depicted in Figures 9a and 9b. To do so, it finds the common subtree between them at steps 8 and 9, and it merges the trees in Figures 9a and 9b together at step 11. The final result is shown in Figure 9c.

6. Discrepancy Measure for Latent Directed Information Trees

In this section, we establish a discrepancy measure for learning minimal directed information trees. Recall that lemma 3 states that the lag between random processes grows by walking along the directed paths in a minimal DIT. This allows us to have the following definition in such graphs:

Definition 10. *For any pair of random processes $(\mathbf{X}_j, \mathbf{X}_k) \in \underline{\mathbf{O}} \times \underline{\mathbf{O}}$, we define the directed measure from $\mathbf{X}_j$ to $\mathbf{X}_k$ denoted by $\gamma(\mathbf{X}_j, \mathbf{X}_k)$ as follows. If $I(\mathbf{X}_k; \mathbf{X}_j) = 0$, then $\gamma(\mathbf{X}_j, \mathbf{X}_k) = -1$, and*

$$\gamma(\mathbf{X}_j, \mathbf{X}_k) := \begin{cases} \max\limits_{d \geq 0} \left\{ d \,:\, I\left(X_{j,1}^d; \mathbf{X}_k\right) = 0 \right\} & j \neq k \\ 0 & j = k. \end{cases} \tag{6.1}$$

Note that $I\left(X_{j,1}^0; \mathbf{X}_k\right) = 0$.

Theorem 5. *Let $\underline{\mathbf{X}} = \underline{\mathbf{O}} \cup \underline{\mathbf{L}}$ be a collection of random processes that form a minimal LDIT, $\overrightarrow{T} = (V, \overrightarrow{E})$, where $V = O \cup L$. If $\underline{\mathbf{X}}$ satisfies assumptions 1, 2, and the sibling resemblance property, then the directed measure defined above is an admissible discrepancy and L is a learnable subset.*

Proof. See appendix I.

7. Sample Complexity for Empirical Estimator

This section studies the complexity of the proposed algorithm to recover the minimal LDIT given N independent and identically distributed (i.i.d.) samples of the observed random processes, $\{\underline{\mathbf{O}}^{(1)}, \ldots, \underline{\mathbf{O}}^{(N)}\}$, where $\underline{\mathbf{O}}^{(q)} = \{\mathbf{X}_1^{(q)}, \ldots, \mathbf{X}_m^{(q)}\}$ denotes the qth sample from all the m processes. $\mathbf{X}_i^{(q)} \in \mathcal{X}^n$ for each i. Consider the case that the alphabet set $\mathcal{X}$ is finite. In order to learn the minimal LDIT, we need to estimate the directed measures introduced in the previous section between all pairs of observed processes. To do so, first we estimate the joint distributions for each pair $(\mathbf{X}_i, \mathbf{X}_j)$ using the empirical estimator defined as

$$\widehat{P}_{\mathbf{X}_i,\mathbf{X}_j}(\mathbf{x}_i, \mathbf{x}_j) := \frac{1}{N} \sum_{q=1}^{N} \mathbb{I}_{\left\{(\mathbf{x}_i,\mathbf{x}_j)=(\mathbf{X}_i^{(q)},\mathbf{X}_j^{(q)})\right\}}, \tag{7.1}$$

where $(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{X}^n \times \mathcal{X}^n$ and $\mathbb{I}$ is the indicator function. Using the empirical distribution of equation 7.1, we can compute the empirical entropies and, consequently, the empirical mutual information.

Lemma 6. *Given N i.i.d. samples of two random processes, $\mathbf{X}_1 \in |\mathcal{X}|^{d_1}$ and $\mathbf{X}_2 \in |\mathcal{X}|^{d_2}$, $d_1, d_2 \leq n$, we have*

$$\mathbb{P}\left(|I(\mathbf{X}_1; \mathbf{X}_2) - \widehat{I}(\mathbf{X}_1; \mathbf{X}_2)| \geq \epsilon\right) \leq 6|\chi|^{2n}\, e^{-N\xi_n(\epsilon)},$$

where

$$\xi_n(\epsilon) = 2\,exp\left(\frac{2\,log\,\frac{\epsilon}{3|\chi|^{2n}}}{log\frac{\epsilon}{3|\chi|^{2n}} - 1}\,log\frac{\epsilon}{3|\chi|^{2n}\,log\frac{3|\chi|^{2n}}{\epsilon}}\right), \tag{7.2}$$

and $\xi_n(\epsilon) > 0$.

Proof. See appendix J.

As long as there exists an estimator for the mutual information $\widehat{I}(\cdot\ \ \cdot)$, such as the empirical estimator in equation 7.1, we can estimate the directed measure, equation 6.1, from $\mathbf{X}_i$ to $\mathbf{X}_j$ by estimating $\widehat{I}(\mathbf{X}_j; X_{i,1}^{d})$ for $d = 1, \ldots, n$.

After choosing an appropriate threshold $\rho > 0$, our estimate of directed measure will be the smallest d for which $\widehat{I}(\mathbf{X}_j; X_{i,1}^d) > \rho$:

$$\widehat{\gamma}(\mathbf{X}_i, \mathbf{X}_j) := \min\{d : \widehat{I}(\mathbf{X}_j; X_{i,1}^d) > \rho\}. \tag{7.3}$$

Theoretically, the best possible threshold is

$$\rho^* := \min_{i \neq j} \left\{ I(X_{i,1}^{\gamma(\mathbf{X}_i, \mathbf{X}_j)+1}; \mathbf{X}_j) \right\}. \tag{7.4}$$

The next theorem presents a concentration bound for our estimate.

Theorem 6. *Given N i.i.d. samples of two random processes* $\mathbf{X}_1$ *and* $\mathbf{X}_2$ *each of length n, and threshold* $0 < \rho \leq \rho^*$ *in equation 7.4, we have*

$$\mathbb{P}\left(\gamma(\mathbf{X}_1, \mathbf{X}_2) \neq \widehat{\gamma}(\mathbf{X}_1, \mathbf{X}_2)\right) \leq 6n|\chi|^{2n}\, e^{-N\xi_n(\rho)},$$

where $\xi_n(\cdot)$ *is given in equation 7.2.*

Proof. Using definition 7.4, one can show $\{\gamma(\mathbf{X}_1, \mathbf{X}_2) \neq \widehat{\gamma}(\mathbf{X}_1, \mathbf{X}_2)\} \subseteq \bigcup_{k=1}^{n}\{|I_k - \widehat{I}_k| \geq \rho\}$, where

$$I_k := I(X_{1,1}^k; \mathbf{X}_2),\ \widehat{I}_k := \widehat{I}(X_{1,1}^k; \mathbf{X}_2).$$

Applying the union bound and lemma 6 concludes the proof.

Most of the practical dynamical systems have finite memory, that is, they have finite Markov order. In such scenarios, the sample complexity reduces extensively. More precisely, consider a dynamical system with finite Markov order p; then in order to estimate $I(X_{i,1}^d; \mathbf{X}_j)$, it suffices to estimate the estimating mutual information between two random processes each of length at most $p+1$. This is true because for a process $\mathbf{X}_j$ of length n and finite Markov order p, we have

$$H(\mathbf{X}_j) = \sum_{t=1}^{n} H(X_{j,t}|X_{j,t-p}^{t-1}) = \sum_{t=1}^{n} H(X_{j,t-p}^{t}) - H(X_{j,t-p}^{t-1}). \tag{7.5}$$

Using the result of lemma 6, theorem 6, and equation 7.5, we obtain the following sample complexity for a network with finite Markov order:

Corollary 1. *Given N i.i.d. samples of two random processes* $\mathbf{X}_1$ *and* $\mathbf{X}_2$*, each of length n with finite Markov order p, and threshold* $0 < \rho \leq \rho^*$ *in equation 7.4, we have*

$$\mathbb{P}\left(\gamma(\mathbf{X}_1, \mathbf{X}_2) \neq \widehat{\gamma}(\mathbf{X}_1, \mathbf{X}_2)\right) \leq 6n^2|\chi|^{2p+2}\, e^{-N\xi_{p+1}(\rho/n)}.$$

Let $\widehat{\overrightarrow{T}}_N = (\widehat{V}_N, \widehat{\overrightarrow{E}}_N)$ denote the reconstructed polytree using the empirical directed measures, equation 7.3, given N i.i.d. samples from the observable processes and assume that the true minimal LDIT was $\overrightarrow{T} = (V, \overrightarrow{E})$. Define the error event as

$$\{\overrightarrow{T} \not\equiv \widehat{\overrightarrow{T}}_N\} := \{V \neq \widehat{V}_N\} \cup \{\overrightarrow{E} \neq \widehat{\overrightarrow{E}}_N\}.$$

That is, an error occurs in the reconstruction algorithm if the set of constructed nodes and edges is not precisely that of the true polytree $\overrightarrow{T}$.

Corollary 2. *Consider a minimal LDIT* $\underline{\mathbf{X}} = \underline{\mathbf{O}} \cup \underline{\mathbf{L}}$ *consisting of m observable nodes. Given N i.i.d. samples from each of the observable processes,*

$$\mathbb{P}\left(\overrightarrow{T} \not\equiv \widehat{\overrightarrow{T}}_N\right) \leq 12\binom{m}{2} n|\chi|^{2n}\, e^{-N\xi_n(\rho)},$$

where $0 < \rho \leq \rho^*$ *and* $\xi_n(\cdot)$ *is given in equation 7.1.*

Proof. Theorem 4 states that given the discrepancies between all pairs of observed nodes, $\overrightarrow{T}$ is recoverable. Since there are m such nodes, $2\binom{m}{2}$ directed measures need to be estimated. Theorem 6 and union bound establish the result.

8. Experimental Results

In this section, we present some experimental results for both synthetic linear system and nonlinear system, and a real data set.

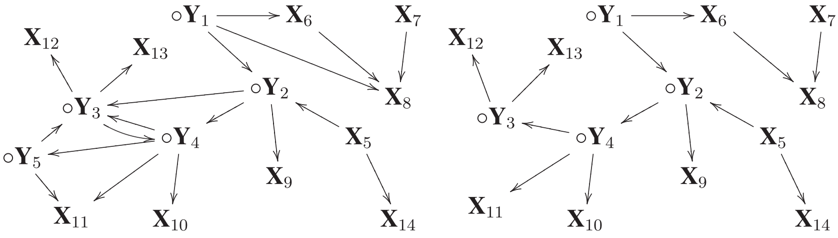
8.1. Autoregressive Model.

We simulated a network of 14 processes corresponding to a polytree with three roots in which four processes were latent. We observed $N \in \{2000, 4000\}$ i.i.d. samples from every observed process, each of length $n = 20$. They were modeled as zero-mean multivariate normal autoregressive time series such that $\underline{\mathbf{Z}}_t = \sum_{i=1}^{3} \mathbf{A}_i \underline{\mathbf{Z}}_{t-i} + \underline{\mathbf{W}}_t$, where $\underline{\mathbf{Z}}_t, \underline{\mathbf{W}}_t \in \mathbb{R}^{14}$, and $\mathbf{A}_i \in \mathbb{R}^{14\times 14}$. $\mathbf{W}_i$s were generated i.i.d. gaussian with mean zero and variance one. The nonzero entries of $\mathbf{A}_i$s are given in Table 1. The first four processes of $\underline{\mathbf{Z}}$ denoted by $(\mathbf{Y}_1, \ldots, \mathbf{Y}_4)$ were the latent ones.

Mutual information between two jointly gaussian random processes $\mathbf{X}$ and $\mathbf{Y}$ is given by (Cover & Thomas, 2012) $I(\mathbf{X}; \mathbf{Y}) = -0.5 \log \frac{|\Sigma_{\mathbf{X},\mathbf{Y}}|}{|\Sigma_{\mathbf{X}}||\Sigma_{\mathbf{Y}}|}$, where $\Sigma_{\mathbf{X}}$ is the covariance matrix of process $\mathbf{X}$, and $\Sigma_{\mathbf{X},\mathbf{Y}}$ is the covariance matrix of $(\mathbf{X}, \mathbf{Y})$. Hence, we were able to estimate the discrepancies equation 7.3, by estimating the covariance matrices between the observed processes. Figures 10a and 10b illustrate the recovered structure for $N = 2000$ and $N = 4000$, respectively.

Table 1: Nonzero Coefficients of the AR Model.

$\mathbf{A}_1$	$A_1(1,1)=1, A_1(2,1)=1, A_1(2,2)=0.5, A_1(2,5)=\sqrt{2}/2, A_1(3,4)=1,$ $A_1(5,5)=1, A_1(6,1)=-2, A_1(8,8)=1, A_1(8,7)=0.1, A_1(10,10)=0.3,$ $A_1(12,12)=\sqrt{2}, A_1(13,13)=-0.2, A_1(13,3)=-1, A_1(14,5)=0.2.$
$\mathbf{A}_2$	$A_2(3,3)=-1, A_2(5,5)=0.2, A_2(7,7)=\sqrt{2}, A_2(8,8)=1, A_2(8,7)=0.2,$ $A_2(9,9)=3, A_2(9,2)=2.5, A_2(10,4)=-1, A_2(11,11)=1, A_2(12,3)=-\sqrt{2}.$
$\mathbf{A}_3$	$A_3(4,2)=\sqrt{3}, A_3(6,6)=1, A_3(8,6)=0.6, A_3(11,4)=-2.$



(a) $N = 2000$

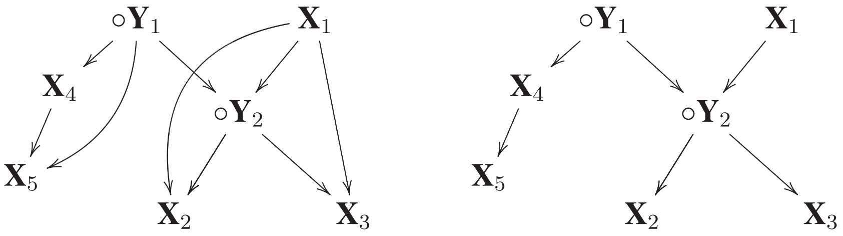
(b) $N = 4000$

Figure 10: Recovered polytree of the AR model. Latent nodes are indicated by circles.

To compute each directed measure pair γ_{jk}, we estimated quantities $f_{j,k}(d) := \widehat{I}(X^d_{j,1}; \mathbf{X}_k)$, for $1 \le d \le 20$ using the above expression for the mutual information. If for all $1 \le d \le 20$, $f_{j,k}(d)$ is less than τ, a sufficiently small threshold (in this example, $\tau = 0.05$), we set the directed measure from j to k, $\gamma_{j,k}$ to -1. Otherwise it is set to equal a value d^*, where d^* is the first value at which $f_{j,k}(d)$ makes a significant jump. That is, $f_{j,k}(d^*)$ is greater than its preceding values $\{f_{j,k}(i), i < d^*\}$. This means ρ in section 7 was set to equal $f_{j,k}(d^* - 1)$.

We see cycles for small number of samples because of estimation errors. When the number of samples is not sufficient to estimate the entries of the discrepancy matrix correctly, the resulting discrepancy matrix will violate some constraints in definition 7, particularly constraint 3.2, which will enforce the algorithm to add cycles in order to be consistent with the estimated discrepancy matrix.

8.2. A Nonlinear Model. We simulated a network of seven processes, which formed a polytree with two roots in which two processes were latent. Denoting the latent processes with Y and the observed ones with X, the



(a) $N = 10^3$ (b) $N = 10^4$

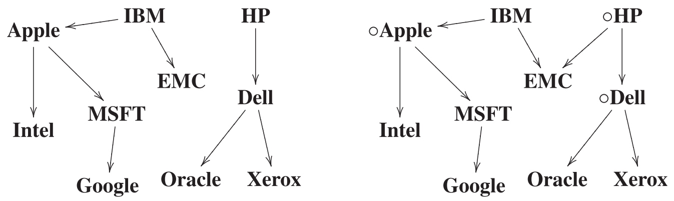
Figure 11: Recovered polytree of the nonlinear model. Latent nodes are indicated by circles.

model is expressed as

$$
\begin{aligned}
Y_1(t) &= Y_1(t-3) + 0.1Y_1(t-2)^2 + \zeta_1(t),\\
X_1(t) &= X_1(t-1)^2/\sqrt{2} - 0.1|\zeta_2(t)|,\\
Y_2(t) &= Y_2(t-1) - X_1(t-1) + 1.5\sqrt{|Y_1(t-1)|} + \zeta_3(t),\\
X_2(t) &= -2Y_2(t-1) + 0.3\sqrt{|X_2(t-3)|^3} + \zeta_4(t),\\
X_3(t) &= 2X_3(t-2) - 0.2Y_2(t-1) + \zeta_5(t),\\
X_4(t) &= X_4(t-1) + \sqrt{|2X_4(t-2)|} - Y_1(t-1)\\
&\qquad + 2Y_1(t-2) + 0.7\log|Y_1(t-3)| + \zeta_6(t),\\
X_5(t) &= 3X_5(t-2) + 2.5X_4(t-2) + \zeta_7(t),
\end{aligned}
$$

where ζ_is were generated i.i.d. gaussian with mean zero and variance one. The observed variables $\{X_1, \ldots, X_5\}$ were each of length $n = 20$, and $N \in \{10^3, 10^4\}$ number of samples from each of them was collected. The directed measures were estimated using equation 7.3 and the mutual information was estimated using the 1-nearest neighbour method in Kraskov, Stögbauer, and Grassberger (2004). The same thresholding procedure of section 8.1 was used to decide whether the estimated mutual information are zero or positive. The recovered networks are depicted in Figure 11.

8.3. Market Analysis. As an example of how our approach may discover causal structure in real-world data, we analyzed the causal relationship between stock prices of 10 technology companies on the New York Stock Exchange sourced from Google Finance for 20 market days (March 3, 2008–March 28, 2008). In this simulation, we assumed that the underlying causal structure did not change during the sampling period. Furthermore, we assumed that influences took a business day to propagate among the



(a) The approximated DI structure of the market data.

(b) Polytree of the market data.

Figure 12: The polytree of the market data. Latent nodes are indicated by circles in panel b.

stocks. Hence, the difference between, t and $t+1$ is one business day. To obtain i.i.d. samples, the price of each stock was sampled every 2 minutes during a business day. This amounted to $N = 200$ number of i.i.d. samples for each stock and $n = 20$.

For this experiment, we used the Black-Scholes model (Black & Scholes, 1973) for the market, in which the stock prices are modeled using a set of coupled stochastic partial differential equations. This model allows modeling the logarithm of the stocks prices as an autoregressive model (Marček, 1998). Thus, the directed measures were estimated similar to section 8.1 from the logarithm of the stock's prices.

Since the underlying true DIG of these 10 companies is not necessarily a polytree, we first approximated the DIG graph of the network by the best-directed tree, where "best" is in the sense of minimizing the Kullback-Leibler (KL) divergence between the true joint and the one resulting from the directed tree approximation. Quinn et al. (2013) showed that the optimal approximate directed tree maximizes the sum of pair-wise directed information terms. Thus, to obtain the best tree approximation, we estimated the pair-wise directed information and found the maximum spanning tree. As depicted in Figure 12a, the approximation identified two disjoint trees. In order to obtain a polytree, we connected the two subtrees by the arrow with maximum directed information weight between the nodes of the two subtrees. This edge was (HP,EMC), as shown in Figure 12b.

HP and IBM are the roots in polytree depicted in Figure 12b. This suggests that they had significant influences on the other companies' stock prices during 2008. In fact, Gartner, Inc. had ranked IBM as the worldwide share leader in the enterprise portal software market based on total software revenue.[7] Furthermore, HP was the global PC market share leader during the same period, followed by Dell Inc.[8] Another observation is the detected

[7] IBM, https://www-03.ibm.com/press/us/en/pressrelease/24507.wss.

[8] Gartner, http://www.gartner.com/newsroom/id/856712.

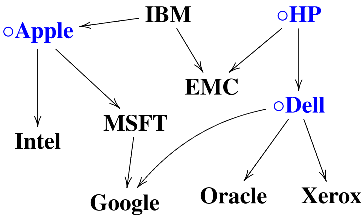


Figure 13: Recovered polytree of the market.

influence of Apple on Intel and Microsoft. Although Apple had begun using Intel processors in its products in 2006, it was only in 2008 that it released MacBook Air and upgraded the processors of MacBook and MacBook Pro to Intel core 2 Duo Penryn, causesing Intels stock price to increase. The arrow from Apple to Microsoft might be a result of the following phenomenon: during 2007–2008, Apple's Mac OS X posted its biggest gain, while the Windows OS market share dove below 90% for the first time.[9]

To test out the latent learning algorithm, we removed the data for Apple, HP, and Dell in the polytree of Figure 12b and ran our algorithm with the data from the remaining seven companies. We used the same thresholding procedure of section 8.1 to obtain the directed measures. The estimated discrepancy matrix is given in equation A.1 and the recovered polytree is shown in Figure 13. The algorithm successfully recovered the hidden nodes, but it added one spurious edge. As a result, the recovered structure is not a polytree. This could be predicted by investigating the estimated discrepancy matrix in equation 8.1. Since entries {(In,Or),(Go,Or),(Ms,Or),(Go,Xr),(Or,Go),(Or,Ib),(Xr,Go),(Xr,In),(Xr,Ms)} are positive, when they should have been −1 due to the fact that these pairs have no common ancestor in Figure 12b. The reason for this may be due to estimation error resulting from an insufficiency of the number of samples or the fact that the true underlying graph is not a polytree.

$$\Gamma_V = \begin{array}{c} \\ Em \\ Go \\ In \\ Ms \\ Ib \\ Or \\ Xr \end{array} \begin{array}{c} \begin{array}{ccccccc} Em & Go & In & Ms & Ib & Or & Xr \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 3 & 0 & 2 & 1 & 3 & 1 & 1 \\ 2 & 1 & 0 & 1 & 2 & 1 & -1 \\ 2 & 0 & 1 & 0 & 2 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 2 & 1 & -1 & -1 & 1 & 0 & 1 \\ 2 & 1 & 1 & 1 & -1 & 1 & 0 \end{pmatrix} \end{array}. \tag{8.1}$$

[9] http://www.computerworld.com/article/2529379/microsoft-windows/windows-market-share-dives-below-90–for-first-time.html.

9. Conclusion

This work presents a new approach for learning latent polytrees when a discrepancy measure is available for the observed nodes. This procedure may be applied to learning latent directed information polytrees from the samples of observed processes. Our algorithms produces a matrix of integer values based on the samples and uses the elements of the matrix to discover the hidden nodes and the connections between the hidden and observed nodes.

Appendix A: Proof of Theorem 2

Suppose Z c-separates U from w in a DIG. Then we need to show

$$I\left(\underline{\mathbf{X}}_U \to \ \mathbf{X}_w || \underline{\mathbf{X}}_Z\right) = 0.$$

Let $\mathcal{A} := \mathcal{PA}(\underline{\mathbf{X}}_w) \setminus Z$ be the parent set of w except the ones that are already in Z. By the definition of DIG, we have

$$I\left(\underline{\mathbf{X}}_U \to \ \mathbf{X}_w || \underline{\mathbf{X}}_{\mathcal{A}}, \underline{\mathbf{X}}_Z\right) = 0. \tag{A.1}$$

If for any t,

$$D\left(P_{\underline{X}^t_{\mathcal{A},1} |\ \underline{X}^t_{U\cup\{w\}\cup Z,1}} \ ||\ P_{\underline{X}^t_{\mathcal{A},1} |\ \underline{X}^t_{\{w\}\cup Z,1}}\right) = 0. \tag{A.2}$$

Then equations A.2 and A.1 will imply the result. In order to show equation A.2, we use the d-separation criterion for the corresponding boundary DAG introduced in section 3.3. Notice that every path from a node in U and a node in $\mathcal{A}$ contains at least a node in $Z \cup \{w\}$ with an outgoing arrow, which implies that every path in the corresponding boundary DAG between $\underline{X}^t_{\mathcal{A},1}$ and $\underline{X}^t_{U,1}$ is d-separated by $\underline{X}^t_{\{w\}\cup Z,1}$. Consequently, equation A.2 holds.

Appendix B: Proof of Lemma 1

We consider two cases. First, if $\mathcal{A}_1, \mathcal{A}_2 \subset \mathcal{R}$ (the root set). Using the chain rule, we have

$$\begin{aligned} P_{\underline{\mathbf{X}}} &= \prod_{t=1}^{n} P_{\underline{X}_t | \underline{\mathbf{X}}_1^{t-1}} \\ &= \prod_{t=1}^{n} \prod_{a\in\mathcal{A}_1} \prod_{b\in\mathcal{A}_2} P_{X_{a,t} | \underline{\mathbf{X}}_1^{t-1}, \underline{\mathbf{X}}_{\mathcal{S}(a),t}} P_{X_{b,t} | \underline{\mathbf{X}}_1^{t-1}, \underline{\mathbf{X}}_{\mathcal{S}(b),t}} P_{\underline{X}_{-\mathcal{A}_1 \cup \mathcal{A}_2, t} | \underline{\mathbf{X}}_1^{t-1}}, \end{aligned} \tag{B.1}$$

where for every x, $\mathcal{S}(x) \subseteq -\{x\}$ such that the above equation holds. Note that if we consider no simultaneous influences, then $\mathcal{S}(x) = \emptyset$ for every x. By the definition of DI, we also have

$$D(P_{\mathbf{X}_a||\underline{\mathbf{X}}_{-\{a\}}} \,||\, P_{\mathbf{X}_a}) = 0, \ \forall a \in \mathcal{A}_1 \cup \mathcal{A}_2.$$

Combining equations B.1 implies

$$P_{\underline{\mathbf{X}}} = P_{\underline{\mathbf{X}}_{\mathcal{A}_1}} P_{\underline{\mathbf{X}}_{\mathcal{A}_2}} \prod_{t=1}^{n} P_{\underline{X}_{-\mathcal{A}_1\cup\mathcal{A}_2,t}|\underline{\mathbf{X}}_1^{t-1}}$$

On the other hand, again using chain rule, we have $P_{\underline{\mathbf{X}}} = P_{\underline{\mathbf{X}}_{\mathcal{A}_1,\mathcal{A}_2}} P_{\underline{\mathbf{X}}\setminus(\mathcal{A}_1\cup\mathcal{A}_2)|\mathcal{A}_1,\mathcal{A}_2}$. The equivalence between the two last equations and the positivity assumption implies that $\underline{\mathbf{X}}_{\mathcal{A}_1}$ and $\underline{\mathbf{X}}_{\mathcal{A}_2}$ are independent.

Otherwise—the second case—let $\mathcal{B}_1$ and $\mathcal{B}_2$ be the set of all parents of $\mathcal{A}_1$ and $\mathcal{A}_2$, respectively. Since the system has a tree structure, then $\mathcal{B}_1 \cap \mathcal{B}_2 = \emptyset$. Similar to the previous case, one can obtain

$$D\left(P_{\underline{\mathbf{X}}_{\mathcal{A}_1}|\underline{\mathbf{X}}_{\mathcal{A}_2\cup\mathcal{B}_1\cup\mathcal{B}_2}} ||P_{\underline{\mathbf{X}}_{\mathcal{A}_1}||\underline{\mathbf{X}}_{\mathcal{B}_1}}\right) = 0.$$

Therefore, $\underline{\mathbf{X}}_{\mathcal{A}_1}$ and $\underline{\mathbf{X}}_{\mathcal{A}_2}$ are independent if $\underline{\mathbf{X}}_{\mathcal{B}_1}$ and $\underline{\mathbf{X}}_{\mathcal{B}_2}$ are independent. By continuing the same procedure, we will end up with two disjoint subsets, $\mathcal{R}_1$ and $\mathcal{R}_2$, of the root set $\mathcal{R}$, such that $\mathcal{R}_i$ is the set of ancestors of $\mathcal{A}_i$. Since $\underline{\mathbf{X}}_{\mathcal{R}_1}$ and $\underline{\mathbf{X}}_{\mathcal{R}_2}$ are independent, $\underline{\mathbf{X}}_{\mathcal{A}_1}$ and $\underline{\mathbf{X}}_{\mathcal{A}_2}$ become independent.

Appendix C: Proof of Lemma 2

Suppose $\mathbf{Y}_h$ is a hidden node in a minimal LDIT with no outgoing edges, and let $\{\mathbf{X}_1, \ldots, \mathbf{X}_s\}$ be its parents. Since $\mathbf{Y}_h$ has no descendant, by marginalizing over $\mathbf{Y}_h$, we obtain s disjoint subtrees. This is a contradiction with the minimality assumption. Now suppose there exists a latent node, $\mathbf{Y}$, in a minimal LDIT with k parents $\underline{\mathbf{X}}_{\mathcal{K}} := \{\mathbf{X}_1, \ldots, \mathbf{X}_k\}$ and one child $\mathbf{X}_0$. From the definition of a generative model graph,

$$\begin{aligned} D(P_{\mathbf{X}_0|\mathbf{Y},\underline{\mathbf{X}}_{\mathcal{K}}}||P_{\mathbf{X}_0||\mathbf{Y}}) &= 0, \\ D(P_{\mathbf{Y}|\underline{\mathbf{X}}_{\mathcal{K}}}||P_{\mathbf{Y}||\underline{\mathbf{X}}_{\mathcal{K}}}) &= 0. \end{aligned} \tag{C.1}$$

By the chain rule,

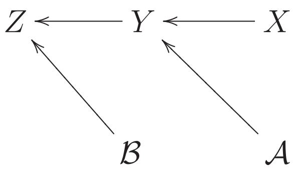


Figure 14: DIG in lemma 3. $\mathcal{A} \cup \{\mathbf{X}\}$ is the parent set of $\mathbf{Y}$, and $\mathcal{B} \cup \{\mathbf{Y}\}$ is the parent set of $\mathbf{Z}$.

$$P_{X^t_{0,1}|\underline{\mathbf{X}}_{\mathcal{K}}} = \sum_{Y_1^{t-1}} P_{X^t_{0,1}|Y_1^{t-1},\underline{\mathbf{X}}_{\mathcal{K}}} P_{Y_1^{t-1}|\underline{\mathbf{X}}_{\mathcal{K}}}. \tag{C.2}$$

From equations C.1 and C.2, we have $D(P_{\mathbf{X}_0|\underline{\mathbf{X}}_{\mathcal{K}}} || P_{\mathbf{X}_0||\underline{\mathbf{X}}_{\mathcal{K}}}) = 0$.

Appendix D: Proof of Lemma 3

It suffices to prove the lemma for $d = 2$, as the case for larger d can be proved by induction. Consider the case where $d = 2$ ($\mathbf{X} \to \mathbf{Y} \to \mathbf{Z}$). Let $\mathcal{A} = \underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{Y})} \setminus \{\mathbf{X}\}$ and $\mathcal{B} = \underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{Z})} \setminus \{\mathbf{Y}\}$ to be the set of parents of $\mathbf{Y}$ and $\mathbf{Z}$ excluding $\mathbf{X}$ and $\mathbf{Y}$, respectively, as shown in Figure 14. First, we show that

$$D(P_{Z_t|Z_1^{t-1},\mathbf{X}} || P_{Z_t|Z_1^{t-1},X_1^{t-2}}) = 0, \ \forall t \leq n. \tag{D.1}$$

Note that if equation D.1 holds, then by multiplying all terms for $t = 1, \ldots, n$, we obtain

$$P_{\mathbf{Z}|\mathbf{X}} = \prod_{t=1}^{n} P_{Z_t|Z_1^{t-1},X_1^{t-2}},$$

which proves our claim. By the chain rule for any t, we have

$$P_{Z_1^t|\mathbf{X}} = \textstyle\sum_{\mathcal{B}_1^{t-1} Y_1^{t-1}} P_{Z_t|Z_1^{t-1},Y_1^{t-1},\mathcal{B}_1^{t-1},\mathbf{X}} P_{Z_1^{t-1}|\mathcal{B}_1^{t-1},Y_1^{t-1},\mathbf{X}} \\ P_{\mathcal{B}_1^{t-1}|Y_1^{t-1},\mathbf{X}} P_{Y_1^{t-1}|\mathbf{X}}. \tag{D.2}$$

Theorem 1, lemma 1, and the definition of generative model imply the following equalities:

$$\begin{aligned} P_{\mathbf{Z}|\mathbf{Y},\mathcal{B},\mathbf{X},\mathcal{A}} &= P_{\mathbf{Z}||\mathbf{Y},\mathcal{B}} = P_{\mathbf{Z}||\mathbf{Y},\mathcal{B},\mathbf{X},\mathcal{A}}, \\ P_{\mathcal{B}|\mathbf{Y},\mathbf{X},\mathcal{A}} &= P_{\mathcal{B}} = P_{\mathcal{B}||\mathbf{X},\mathcal{A},\mathbf{Z}}, \\ P_{\mathbf{Y}|\mathbf{X},\mathcal{A}} &= P_{\mathbf{Y}||\mathbf{X},\mathcal{A}} = P_{\mathbf{Y}||\mathbf{X},\mathcal{A},\mathbf{Z},\mathcal{B}}. \end{aligned} \tag{D.3}$$

The above equalities imply

$$\begin{aligned}P_{Z_t|Z_1^{t-1},Y_1^{t-1},\mathcal{B}_1^{t-1},\mathbf{X}}&=P_{Z_t|Z_1^{t-1},Y_1^{t-1},\mathcal{B}_1^{t-1},X_1^{t-2}},\\ P_{Z_1^{t-1}|Y_1^{t-1},\mathcal{B}_1^{t-1},\mathbf{X}}&=P_{Z_1^{t-1}|Y_1^{t-1},\mathcal{B}_1^{t-1},X_1^{t-2}},\\ P_{\mathcal{B}_1^{t-1}|Y_1^{t-1},\mathbf{X}}&=P_{\mathcal{B}_1^{t-1}|Y_1^{t-1},X_1^{t-2}}.\end{aligned}\tag{D.4}$$

Moreover, one can obtain the following equation using the chain rule, lemma 5, and the equalities in equation D.3:

$$\begin{aligned}P_{Y_1^{t-1}|\mathbf{X}}&=\sum_{\mathcal{A}_1^{t-2}}P_{Y_1^{t-1}|\mathcal{A}_1^{t-2},\mathbf{X}}P_{\mathcal{A}_1^{t-2}|\mathbf{X}}\\ &=\sum_{\mathcal{A}_1^{t-2}}P_{Y_1^{t-1}|\mathcal{A}_1^{t-2},X_1^{t-2}}P_{\mathcal{A}_1^{t-2}|X^{t-2}}=P_{Y_1^{t-1}|X_1^{t-2}}.\end{aligned}\tag{D.5}$$

Substituting equations D.4 and D.5 into the right-hand side of equation D.2 proves our claim.

Appendix E: Proof of Lemma 4

It suffices to show

$$D(P_{\mathbf{Y}|\mathbf{W},\mathbf{X}}||P_{\mathbf{Y}||\mathbf{W}})=0.\tag{E.1}$$

Suppose the length of the path from **W** to **Y** is d. We will prove equation E.1 by induction on d. For $d=1$, define $\mathcal{A}:=\underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{Y})}\setminus\{\mathbf{W}\}$. In this case, similar to the proof of lemma 3, the following equalities hold:

$$\begin{aligned}D(P_{\mathbf{Y}|\mathcal{A},\mathbf{W},\mathbf{X}}||P_{\mathbf{Y}||\mathcal{A},\mathbf{W}})&=0,\\ D(P_{\mathcal{A}|\mathbf{W},\mathbf{X}}||P_{\mathcal{A}})&=0.\end{aligned}\tag{E.2}$$

From the chain rule,

$$P_{Y_1^t|\mathbf{W},\mathbf{X}}=\sum_{\mathcal{A}}P_{Y_t|Y_1^{t-1},\mathcal{A},\mathbf{W},\mathbf{X}}P_{Y_1^{t-1}|\mathcal{A},\mathbf{W},\mathbf{X}}P_{\mathcal{A}|\mathbf{W},\mathbf{X}}.$$

Then, by applying equation E.2 to the above equation, we obtain equation E.1.

Assume that equation E.1 holds for paths of length $d<k$. In order to prove the case $d=k$, let **Z** be the parent of **Y** on the path from **W** to **Y**, and $\mathcal{B}:=\underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{Y})}\setminus\{\mathbf{Z}\}$. The path from **W** to **Z** is of length $k-1$, so by induction hypothesis, we have

$$D(P_{\mathbf{Z}|\mathbf{W},\mathbf{X}}||P_{\mathbf{Z}||\mathbf{W}})=0. \tag{E.3}$$

Moreover, by the definition of the generative model graph and theorem 1,

$$\begin{aligned}&D(P_{\mathbf{Y}|\mathcal{B},\mathbf{Z},\mathbf{W},\mathbf{X}}||P_{\mathbf{Y}||\mathcal{B},\mathbf{Z}})=0,\\&\qquad D(P_{\mathcal{B}|\mathbf{Z},\mathbf{W},\mathbf{X}}||P_{\mathcal{B}})=0.\end{aligned} \tag{E.4}$$

The chain rule implies

$$P_{Y^t|\mathbf{W},\mathbf{X}}=\sum_{\mathcal{B},\mathbf{Z}}P_{Y^t|\mathcal{B},\mathbf{Z},\mathbf{W},\mathbf{X}}P_{\mathcal{B}|\mathbf{Z},\mathbf{W},\mathbf{X}}P_{\mathbf{Z}|\mathbf{W},\mathbf{X}}.$$

Applying equations E.3 and E.4 to the above equation proves the claim.

Appendix F: Proof of Lemma 5

Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be two disjoint subsets of the root set $\mathcal{R}$ in a minimal LDIT. Furthermore, assume $\mathcal{R}_1$ and $\mathcal{R}_2$ are root ancestors for nodes $\mathbf{X}$ and $\mathbf{Y}$, respectively. Denote all the nodes on the paths from $\mathcal{R}_1$ to $\mathbf{X}$ by $\mathcal{A}$. It is easy to check that if a node belongs to $\mathcal{A}$, so do all of its parents. Therefore, $\underline{\mathbf{X}}_{\mathcal{PA}(\mathbf{X})}\subseteq\mathcal{A}$, where $\mathcal{PA}(\mathbf{X})$ is the parent set of $\mathbf{X}$. Similarly, we denote all the nodes on the paths from $\mathcal{R}_2$ to $\mathbf{Y}$ by $\mathcal{B}$. By the definition of generative model, we obtain

$$P_{\mathbf{X},\mathbf{Y},\mathcal{R}_1,\mathcal{R}_2,\mathcal{A},\mathcal{B}}=P_{\mathcal{R}_1}P_{\mathcal{R}_2}\Psi_{\mathcal{A},\mathcal{R}_1}\Phi_{\mathcal{B},\mathcal{R}_2}P_{\mathbf{X}||\mathcal{PA}(\mathbf{X})}P_{\mathbf{Y}||\mathcal{PA}(\mathbf{Y})}, \tag{F.1}$$

where Ψ and Φ represent the terms including the causal conditioned distributions of all processes on the paths from $\mathcal{A}_1$ to $\mathbf{X}$, and from $\mathcal{A}_2$ to $\mathbf{Y}$, respectively. From the chain rule, we obtain

$$\begin{aligned}P_{\mathbf{X},\mathbf{Y},\mathcal{R}_1,\mathcal{R}_2,\mathcal{A},\mathcal{B}}=&P_{\mathcal{R}_1,\mathcal{R}_2}P_{\mathcal{A}|\mathcal{R}_1,\mathcal{R}_2}P_{\mathcal{B}|\mathcal{A},\mathcal{R}_1,\mathcal{R}_2}\\&P_{\mathbf{X}|\mathcal{B},\mathcal{A},\mathcal{R}_1,\mathcal{R}_2}P_{\mathbf{Y}|\mathbf{X},\mathcal{B},\mathcal{A},\mathcal{R}_1,\mathcal{R}_2}.\end{aligned} \tag{F.2}$$

The equivalence between equations F.1 and F.2, and the positivity assumption imply that $\mathbf{X}$ and $\mathbf{Y}$ are independent whenever $\mathcal{PA}(\mathbf{X})$ and $\mathcal{PA}(\mathbf{Y})$ are independent. Continuing the same procedure, we can show that $\mathbf{X}$ and $\mathbf{Y}$ are independent if $\mathcal{R}_1$ and $\mathcal{R}_2$ are independent.

Appendix G: Proof of Theorem 3

The proof consists of two parts. First, we show that if $\mathcal{PA}_i$ is the parent set of $\mathbf{X}_i$ in an MDIG, then

$$D\left(P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i\},1}} \,||\, P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},\underline{\mathbf{X}}^{t}_{\mathcal{PA}_i,1}}\right) = 0. \tag{G.1}$$

To do so, we use the definition of MDIG in section 4.2. Let $\mathcal{R} = -\{i\} \setminus \mathcal{PA}_i$ be the set of all nodes except i and its parents. Since there is no arrow in MDIG from $\mathcal{R}$ to $\mathbf{X}_i$, we have

$$\widetilde{I}(\mathbf{X}_r \to \mathbf{X}_i || \underline{\mathbf{X}}_{-\{i,r\}}) = 0, \;\; \forall r \in \mathcal{R}.$$

The positivity assumption, together with the above equalities, imply

$$D\left(P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i\},1}} \,||\, P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},\bigcap_{r\in\mathcal{R}}\underline{\mathbf{X}}^{t}_{-\{i,r\},1}}\right) = 0.$$

Noticing that $\bigcap_{r\in\mathcal{R}} \underline{\mathbf{X}}^{t}_{-\{i,r\},1} = \underline{\mathbf{X}}^{t}_{\mathcal{PA}_i,1}$, one can establish equation G.1. Next we show that if there is an arrow from $\mathbf{X}_j$ to $\mathbf{X}_i$ in an MDIG with polytree structure (e.g., $\widetilde{I}(\mathbf{X}_j \to \mathbf{X}_i || \underline{\mathbf{X}}_{-\{i,j\}}) > 0$), then

$$D\left(P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i\},1}} \,||\, P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},X_{j,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i,j\},1}}\right) = 0. \tag{G.2}$$

In words, given the past of $\mathbf{X}_j$ is enough for predicting the $X_{i,t}$. To prove equation G.2, we use the fact that the graph is a polytree, and thus if there is an arrow from $\mathbf{X}_j$ to $\mathbf{X}_i$, there will be no arrow in the opposite direction:

$$\widetilde{I}(\mathbf{X}_i \to \mathbf{X}_j || \underline{\mathbf{X}}_{-\{i,j\}}) = 0.$$

Consequently,

$$D\left(P_{\mathrm{X}_{j,t}|X_{j,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{j\},1}} \,||\, P_{\mathrm{X}_{j,t}|X_{j,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i,j\},1}}\right) = 0.$$

On the other hand, the chain rule implies

$$P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i\},1}} = P_{\mathrm{X}_{j,t}|X_{j,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{j\},1}} \frac{P_{\mathrm{X}_{i,t}|X_{i,1}^{t-1},X_{j,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i,j\},1}}}{P_{\mathrm{X}_{j,t}|X_{j,1}^{t-1},X_{i,1}^{t-1},\underline{\mathbf{X}}^{t}_{-\{i,j\},1}}}.$$

Combining the last two equations will imply equation G.2.

Appendix H: Proof of Theorem 4

First, we prove that Γ_O suffices to learn $\overrightarrow{T}$ when $\mathcal{R} = \{r\}$. The proof is by induction on $|O|$. The base case, $|O| = 1$, is trivial, since by definition 8, L

must be empty and $\overrightarrow{T}$ is the single node. Suppose a tree $\overrightarrow{T} = (V, \overrightarrow{E})$ can be recovered given any learnable subset L such that $|O| \leq k - 1$. For the case that $|O| = k$, let $v \in O$ and $B_v := \arg\min_{u \in O \setminus \{v\}} \gamma_r(v, u)$. Note that in a single root tree, all the discrepancies must be nonnegative. We claim that $\overrightarrow{T}$ is a star with a root in the center if and only if $B_v = O \setminus \{v\}$ for all $v \in O$. If $\overrightarrow{T}$ is a star, then clearly $B_v = O \setminus \{v\}$ for all $v \in O$. The other direction is proved by arguing that if $\overrightarrow{T}$ is not a star, then there exists a directed path of length two, and because L is learnable, one can find a node on this path such that $B_v \neq O \setminus \{v\}$.

If there exists $v \in O$ such that $B_v \neq O \setminus \{v\}$, and $\min_{u \in O \setminus \{v\}} \gamma_r(v, u) = 0$, then all the nodes in B_v are the descendants of v. In this case by induction hypothesis, the subtree of $\overrightarrow{T}$ containing v and all its descendants, is recoverable by $B_v \cup \{v\}$, as well as the rest of the tree by $O \setminus B_v$. Similarly for the case $\min_{u \in O \setminus \{v\}} \gamma_r(v, u) > 0$.

We show that if $|\mathcal{R}| > 1$, learning $\overrightarrow{T}$ can be done by learning $|\mathcal{R}|$ single rooted trees, separately.

For $v \in O$, let M_v be a maximal subset of O containing v such that for every $u, w \in M_v$, $\gamma(u, w) \geq 0$. Clearly, if w belongs to M_v, so does all its descendants, which are also in O.

Denote the minimal induced polytree of $\overrightarrow{T}$ containing M_v by $\overrightarrow{T}|_{M_v} = (V', \overrightarrow{E}')$. Note that from the maximality of M_v, $O \cap V' \subseteq M_v$. First, we show that $V' \setminus M_v$ is a learnable subset in $\overrightarrow{T}|_{M_v}$; all nodes with out-degree at most one in $\overrightarrow{T}|_{M_v}$ belong to M_v. All leaves in $\overrightarrow{T}|_{M_v}$ belong to M_v; otherwise, they can be eliminated from $\overrightarrow{T}|_{M_v}$, and it is a contradiction with the minimality assumption on $\overrightarrow{T}|_{M_v}$. Let $u' \in V' \setminus M_v$ be a node with out-degree one in $\overrightarrow{T}|_{M_v}$. Since $O \cap V' \subseteq M_v$, $u' \in L$. If the out-degree of u' is also one in $\overrightarrow{T}$, then we have a contradiction with the learnability assumption of L. Hence, there exists at least one descendant of u' in O that does not belong to $\overrightarrow{T}|_{M_v}$, in which case we have a contradiction with the maximality of M_v.

Next, we claim that $\overrightarrow{T}|_{M_v}$ has only one root from the root set $\mathcal{R}$. Suppose $\overrightarrow{T}|_{M_v}$ has more than one root. Since a tree has no cycles, there must exist at least two nodes with degree one (either a root with degree one or a leaf) with no common ancestor in $\overrightarrow{T}|_{M_v}$, which contradicts the definition of M_v.

The final step is to prove that these single rooted subtrees can be merged uniquely. This can be done by observing that if two single-rooted trees $\overrightarrow{T}_1 = (V_1, \overrightarrow{E}_1)$ and $\overrightarrow{T}_2 = (V_2, \overrightarrow{E}_2)$ have an intersection in $\overrightarrow{T}$, then that intersection is also a single-rooted tree that can be learned from $O \cap V_1 \cap V_2$.

Appendix I: Proof of Theorem 5

To show this we prove that the directed measure in equation 6.1 is a discrepancy measure on T. First, it is important to note that by lemma 2, the set of hidden nodes is a learnable subset in a minimal LDIT. The rest of the proof verifies that directed measure in equation 6.1 satisfies the properties of a discrepancy measure introduced in definition 7.

1. From definition 10, $\gamma(\mathbf{X}, \mathbf{X}) = 0$. Suppose $\mathbf{X}$ is an ancestor of $\mathbf{Y}$. By the sibling resemblance property, since $\mathbf{X}$ is the, common ancestor of $\mathbf{X}$ and $\mathbf{Y}$ and $I(X_1; \mathbf{X}) > 0$, then $I(X_1; \mathbf{Y}) > 0$. In other words $\gamma(\mathbf{X}, \mathbf{Y}) = 0$.

2. This property is also a consequence of the sibling resemblance property. Let $\mathbf{W}$ be the common ancestor of $\mathbf{X}$ and $\mathbf{Y}$. If $\gamma(\mathbf{X}, \mathbf{W}) = d$, then by using lemma 4, we obtain $I(X_1^d; \mathbf{Y}) = 0$, which implies $\gamma(\mathbf{X}, \mathbf{Y}) \geq d$. On the other hand, since $I(X_1^{d+1}; \mathbf{W}) > 0$ and $I(\mathbf{Y}; \mathbf{W}) > 0$, by sibling resemblance property, we obtain $I(X_{d+1}; \mathbf{Y}|X_1^d) > 0$, which implies $\gamma(\mathbf{X}, \mathbf{Y}) = \gamma(\mathbf{X}, \mathbf{W}) = d$.

3. This is shown by proving that for a given path $\mathbf{X} \to \mathbf{Y} \to \mathbf{Z}$ in a minimal LDIT, if $\gamma(\mathbf{Y}, \mathbf{X}) = l$ and $\gamma(\mathbf{Z}, \mathbf{Y}) = d$, then $\gamma(\mathbf{Z}, \mathbf{X}) > \max\{l, d\}$.

First, we prove $\gamma(\mathbf{Z}, \mathbf{X}) > d$. It suffices to show

$$I(Z_{d+1}; \mathbf{X}|Z_1^d) = 0. \tag{I.1}$$

Using the chain rule, we obtain

$$P_{Z_{d+1}|Z_1^d, \mathbf{X}} = \sum_{Y_1} P_{Z_{d+1}|Z_1^d, Y_1, \mathbf{X}} P_{Z_1^d|Y_1, \mathbf{X}} \frac{P_{Y_1|\mathbf{X}}}{P_{Z_1^d|\mathbf{X}}}. \tag{I.2}$$

Since $\gamma(\mathbf{Z}, \mathbf{Y}) = d$, $\mathbf{Y}$ is an ancestor of $\mathbf{Z}$ and by using the same argument as in the proof of lemma 3, we obtain

$$D(P_{Z_1^d|\mathbf{Y}, \mathbf{X}}||P_{Z_1^d}) = 0 \;\; D(P_{Y_1|\mathbf{X}}||P_{Y_1}) = 0, \tag{I.3}$$

$$D(P_{Z_{d+1}|Z_1^d, Y_1, \mathbf{X}}||P_{Z_{d+1}|Z_1^d, Y_1}) = 0. \tag{I.4}$$

Finally, the claim follows by substituting equations I.3 and I.4 into the right-hand side of equation I.2. The statement $\gamma(\mathbf{Z}, \mathbf{X}) > l$ may be proven by showing $I(Z_{l+1}; \mathbf{X}|Z_1^l) = 0$,

$$P_{Z_{l+1}|Z_1^l, \mathbf{X}} = \sum_{Y_1^l} P_{Z_{l+1}|Z_1^l, Y_1^l, \mathbf{X}} P_{Z_1^l|Y_1^l, \mathbf{X}} \frac{P_{Y_1^l|\mathbf{X}}}{\sum_{Y_1^{\prime l-1}} P_{Z_1^l|Y_1^{\prime l-1}, \mathbf{X}} P_{Y_1^{\prime l-1}|\mathbf{X}}},$$

since $\gamma(\mathbf{Y}, \mathbf{X}) = l$, and using the same argument as above, one can prove the claim.

4. This property is a direct consequence of lemma 5 and definition 10.

Appendix J: Proof of Lemma 6

First we prove the following lemma which will be used in the proof of lemma 6.

Lemma 7. *Let $1 \leq a/x$ and $x \geq 0$. For any $0 < \lambda < 1$, $x \log \frac{a}{x}$ is bounded from above by $\frac{a^{\lambda} x^{1-\lambda}}{\lambda}$.*

Proof. Since $1 \leq a/x$, then $\log \left(\frac{a}{x}\right)^{\lambda} \leq \left(\frac{a}{x}\right)^{\lambda}$, for any $0 < \lambda < 1$. Hence, $\lambda x \log \frac{a}{x} \leq a^{\lambda} x^{1-\lambda}$.

Proof of Lemma 6. Using McDiarmid's inequality (McDiarmid, 1989) and the union bound for the empirical estimator, equation 7.1, we obtain

$$\mathbb{P}\left(\max_{(\mathbf{x}_1,\mathbf{x}_2)\in|X|^{d_1+d_2}} |P_{\mathbf{X}_1,\mathbf{X}_2}(\mathbf{x}_1,\mathbf{x}_2) - \widehat{P}_{\mathbf{X}_1,\mathbf{X}_2}(\mathbf{x}_1,\mathbf{x}_2)| \geq \delta \right)$$
$$\leq 2|\mathcal{X}|^{d_1+d_2} e^{-2N\delta^2} \leq 2|\mathcal{X}|^{2n} e^{-2N\delta^2}. \tag{J.1}$$

For simplicity, denote $(\mathbf{X}_1, \mathbf{X}_2)$ by $\mathbf{Z}$. From $||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1 \leq |\mathcal{X}|^{2n} \max_{\mathbf{Z}} |P_{\mathbf{Z}}(\mathbf{Z}) - \widehat{P}_{\mathbf{Z}}(\mathbf{Z})|$ and equation J.1, we obtain

$$\mathbb{P}\left(||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1 \geq |\mathcal{X}|^{2n}\delta \right) \leq 2|\mathcal{X}|^{2n} e^{-2N\delta^2}. \tag{J.2}$$

Using an ℓ_1-norm bound on entropy (Cover & Thomas, 2012), if $||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1 < 0.5$, then

$$|H(\mathbf{Z}) - \widehat{H}(\mathbf{Z})| \leq ||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1 \log \frac{|\mathcal{X}|^{d_1+d_2}}{||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1}.$$

Applying lemma 7, we have

$$|H(\mathbf{Z}) - \widehat{H}(\mathbf{Z})| \leq \frac{1}{\lambda} ||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1^{1-\lambda} |\mathcal{X}|^{\lambda(d_1+d_2)}. \tag{J.3}$$

Therefore,

$$\mathbb{P}\left(|H(\mathbf{Z}) - \widehat{H}(\mathbf{Z})| \geq \epsilon \right) \leq \mathbb{P}\left(||P_{\mathbf{Z}} - \widehat{P}_{\mathbf{Z}}||_1^{1-\lambda} \geq \frac{\lambda\epsilon}{|\mathcal{X}|^{\lambda(d_1+d_2)}} \right).$$

From equation J.2, we have

$$\mathbb{P}\left(|H(\mathbf{Z}) - \widehat{H}(\mathbf{Z})| \geq \epsilon \right) \leq 2|\mathcal{X}|^{2n} \exp\left(-2N \left(\frac{\lambda\epsilon}{|\mathcal{X}|^{2n}} \right)^{\frac{2}{1-\lambda}} \right). \tag{J.4}$$

Using the definition of mutual information,

$$I(\mathbf{X}_1; \mathbf{X}_2) = H(\mathbf{X}_1) + H(\mathbf{X}_2) - H(\mathbf{X}_1, \mathbf{X}_2),$$

we obtain

$$\mathbb{P}\left(|I(\mathbf{X}_1; \mathbf{X}_2) - \widehat{I}(\mathbf{X}_1; \mathbf{X}_2)| \geq \epsilon\right) \leq \mathbb{P}\left(|H(\mathbf{X}_1) - \widehat{H}(\mathbf{X}_1)| \geq \epsilon/3\right) +$$
$$\mathbb{P}\left(|H(\mathbf{X}_2) - \widehat{H}(\mathbf{X}_2)| \geq \epsilon/3\right) + \mathbb{P}\left(|H(\mathbf{X}_1, \mathbf{X}_2) - \widehat{H}(\mathbf{X}_1, \mathbf{X}_2)| \geq \epsilon/3\right).$$

Applying the upper bound in equation J.4 to the above inequality will conclude the lemma. It remains only to choose λ to minimize the right-hand side of equation J.4. We choose $\lambda = 1/\log(\frac{3|\mathcal{X}|^{2n}}{\epsilon})$.

Acknowledgments

We thank the anonymous referees for their helpful comments on an earlier version of this letter. This work was supported in part by NSF CCF 10-54937-CAREER, NSF Center for Science of Information (CCF-0939370), and MURI grant ARMY W911NF-15-1-0479.

References

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

Anandkumar, A., Chaudhuri, K., Hsu, D., Kakade, S. M., Song, L., & Zhang, T. (2011). Spectral methods for learning multivariate latent tree structure. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira, & Q. Weinberger (Eds.), *Advances in neural information processing systems 24*, (pp. 2025–2033). Red Hook, NY: Curran.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, *81*, 637–654.

Boyen, X., Friedman, N., & Koller, D. (1999). Discovering the hidden structure of complex dynamic systems. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence* (pp. 91–100). San Mateo, CA: Morgan Kaufmann.

Chandrasekaran, V., Parrilo, P. A., & Willsky, A. S. (2010). Latent variable graphical model selection via convex optimization. In *Proceedings of the 2010 48th Annual Allerton Conference on Communication, Control, and Computing* (pp. 1610–1613). Piscataway, NJ: IEE.

Chávez, M., Martinerie, J., & Le Van Quyen, M. (2003). Statistical assessment of nonlinear causality: Application to epileptic EEG signals. *Journal of Neuroscience Methods*, *124*(2), 113–128.

Chen, X., Hero, A. O., & Savarese, S. (2012). Multimodal video indexing and retrieval using directed information. *IEEE Transactions on Multimedia*, *14*(1), 3–16.

Choi, M. J., Tan, V. Y., Anandkumar, A., & Willsky, A. S. (2011). Learning latent tree graphical models. *Journal of Machine Learning Research*, *12*, 1771–1812.

Chow, C., & Liu, C. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, *14*(3), 462–467.

Cover, T. M., & Thomas, J. A. (2012). *Elements of information theory*. Hoboken, NJ: Wiley.

Dahlhaus, R. (2000). Graphical interaction models for multivariate time series 1. *Metrika*, *51*(2), 157–172.

Dahlhaus, R., & Eichler, M. (2003). *Causality and graphical models in time series analysis*. In P. J. Green, N. L. Hjort, & S. Richardson (Eds.), *Highly structured stochastic systems* (pp. 115–137). New York: Oxford University Press.

de Campos, L. M. (1994). *Independency relationships in singly connected networks*. (DESCAI Technical Report 960204). Grande: University of Grande.

Eichler, M. (2007). Granger causality and path diagrams for multivariate time series. *Journal of Econometrics*, *137*(2), 334–353.

Elidan, G., Friedman, N., & Chickering, D. M. (2005). Learning hidden variable networks: The information bottleneck approach. *Journal of Machine Learning Research*, *6*(1), 81–127.

Elidan, G., Nachman, I., & Friedman, N. (2007). "Ideal parent" structure learning for continuous variable bayesian networks. *Journal of Machine Learning Research*, *8*(8), 1799–1833.

Erdos, P. L., Steel, M. A., Székely, L., & Warnow, T. J. (1999). A few logs suffice to build (almost) all trees: Part II. *Theoretical Computer Science*, *221*(1), 77–118.

Farris, J. S. (1972). Estimating phylogenetic trees from distance matrices. *American Naturalist*, *106*, 645–668.

Geiger, P., Zhang, K., Schölkopf, B., Gong, M., & Janzing, D. (2015). Causal inference by identification of vector autoregressive processes with hidden components. In *Proceedings of the 32nd International Conference on Machine Learning* (pp. 1917–1925). N.p.: International Machine Learning Society.

Gourévitch, B., & Eggermont, J. J. (2007). Evaluating information transfer between auditory cortical neurons. *Journal of Neurophysiology*, *97*(3), 2533–2543.

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, *37*, 424–438.

Hyvärinen, A., & Smith, S. M. (2013). Pairwise likelihood ratios for estimation of nongaussian structural equation models. *Journal of Machine Learning Research*, *14*(1), 111–152.

Ishteva, M., Park, H., & Song, L. (2013). Unfolding latent tree structures using 4th order tensors. In *Proceedings of the International Conference on Machine Learning*. N.p.: International Machine Learning Society.

Jalali, A., & Sanghavi, S. (2012). Learning the dependence graph of time series with latent factors. In *Proceedings of the International Conference on Machine Learning*. N.p.: International Machine Learning Society.

Jiang, T., Kearney, P., & Li, M. (2001). A polynomial time approximation scheme for inferring evolutionary trees from quartet topologies and its application. *SIAM Journal on Computing*, *30*(6), 1942–1961.

John, K. S., Warnow, T., Moret, B. M., & Vawter, L. (2003). Performance study of phylogenetic methods: (Unweighted) quartet methods and neighbor-joining. *Journal of Algorithms*, *48*(1), 173–193.

Kadloor, S., Kiyavash, N., & Venkitasubramaniam, P. (2012a). Mitigating timing based information leakage in shared schedulers. In *INFOCOM, 2012 Proceedings of the IEEE* (pp. 1044–1052). Piscataway, NJ: IEEE.

Kadloor, S., Kiyavash, N., & Venkitasubramaniam, P. (2012b). Scheduling with privacy constraints. In *Proceedings of the 2012 IEEE Information Theory Workshop* (pp. 40–44). Piscataway, NJ: IEEE.

Kim, S., Putrino, D., Ghosh, S., & Brown, E. N. (2011). A Granger causality measure for point process models of ensemble neural spiking activity. *PLoS Computational Biology*, *7*(3), e1001110.

Kiyavash, N., & Coleman, T. (2009). Covert timing channels codes for communication over interactive traffic. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 1485–1488). Piscataway, NJ: IEEE.

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: Principles and techniques*. Cambridge, MA: MIT Press.

Kramer, G. (1998). *Directed information for channels with feedback*. Ph.D. dissertation, University of Manitoba, Canada.

Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical Review E*, *69*(6), 066138.

Lazarsfeld, P. F., & Henry, N. W. (1968). *Latent structure analysis*. Boston: Houghton Mifflin.

Liu, Y., & Aviyente, S. (2010). Information theoretic approach to quantify causal neural interactions from EEG. In *Proceedings of the Forty-Fourth Asilomar Conference on Signals, Systems and Computers* (pp. 1380–1384). Piscataway, NJ: IEEE.

Lodish, H., Berk, A., Zipursky, S. L., Matsudaira, P., Baltimore, D., & Darnell, J. (2000). *Molecular cell biology*. New York: Freeman.

Marček, D. (1998). Stock price prediction using autoregressive models and signal processing procedures. In *Proceedings of the 16th Conference MME* (vol. 98, pp. 114–121).

Marko, H. (1973). The bidirectional communication theory: A generalization of information theory. *IEEE Transactions on Communications*, *21*(12), 1345–1351.

Massey, J. (1990). Causality, feedback and directed information. In *Proc. Int. Symp. Inf. Theory Applic.* (pp. 303–305). Piscataway, NJ: IEEE.

McDiarmid, C. (1989). On the method of bounded differences. *Surveys in Combinatorics*, *141*(1), 148–188.

Messaouda, O., Oommen, J. B., & Matwin, S. (2003). Enhancing caching in distributed databases using intelligent polytree representations. In Y. Xiang (Ed.), *Advances in artificial intelligence* (pp. 498–504). New York: Springer.

Murphy, K. P. (2002). *Dynamic Bayesian networks: Representation, inference and learning*. Ph.D. dissertation, University of California.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo, CA: Morgan Kaufmann.

Pearl, J. (2000). *Causality: Models, reasoning and inference*. Cambridge: Cambridge University Press.

Quinn, C. J., Coleman, T. P., Kiyavash, N., & Hatsopoulos, N. G. (2011). Estimating the directed information to infer causal relationships in ensemble neural spike train recordings. *Journal of Computational Neuroscience*, *30*(1), 17–44.

Quinn, C. J., Kiyavash, N., & Coleman, T. P. (2011). Equivalence between minimal generative model graphs and directed information graphs. In *Proceedings of the 2011 IEEE International Symposium on Information Theory* (pp. 293–297). Piscataway, NJ: IEEE.

Quinn, C. J., Kiyavash, N., & Coleman, T. P. (2013). Efficient methods to compute optimal tree approximations of directed information graphs. *IEEE Transactions on Signal Processing*, *61*(12), 3173–3182.

Quinn, C., Kiyavash, N., & Coleman, T. P. (2015). Directed information graphs. *IEEE Transactions on Information Theory*, *61*(12), 6887–6909.

Rao, A., Hero III, A. O., States, D. J., & Engel, J. D. (2007). Motif discovery in tissue-specific regulatory sequences using directed information. *EURASIP Journal on Bioinformatics and Systems Biology*, *2007*, 3.

Rebane, G., & Pearl, J. (1987). The recovery of causal poly-trees from statistical data. In *Proceedings of the Third Conference on Uncertainty in Artificial Intelligence*. Amsterdam: Elsevier.

Rissanen, J., & Wax, M. (1987). Measures of mutual and causal dependence between two time series. *IEEE Transactions on Information Theory*, *33*(4), 598–601.

Runge, J. (2014). *Detecting and quantifying causality from time series of complex systems*. Ph.D. dissertation, Humboldt-Universität zu Berlin.

Saitou, N., & Nei, M. (1987). The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, *4*(4), 406–425.

Sattath, S., & Tversky, A. (1977). Additive similarity trees. *Psychometrika*, *42*(3), 319–345.

Schreiber, T. (2000). Measuring information transfer. *Physical Review Letters*, *85*(2), 461.

Schum, D. A. (1994). *The evidential foundations of probabilistic reasoning*. Evanston, IL: Northwestern University Press.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., & Kerminen, A. (2006). A linear nongaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, *7*, 2003–2030.

Shmatikov, V., & Wang, M.-H. (2006). Timing analysis in low-latency mix networks: Attacks and defenses. In *Computer Security–ESORICS 2006* (pp. 18–33). New York: Springer.

Spirtes, P., Meek, C., & Richardson, T. (1995). Causal inference in the presence of latent variables and selection bias. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence* (pp. 499–506). San Mateo, CA: Morgan Kaufmann.

Steel, M. (1992). The complexity of reconstructing trees from qualitative characters and subtrees. *Journal of Classification*, *9*(1), 91–116.

Sucar, L. E., Pérez-Brito, J., Ruiz-Suárez, J. C., & Morales, E. (1997). Learning structure from data and its application to ozone prediction. *Applied Intelligence*, *7*(4), 327–338.

Verma, T. S., & Pearl, J. (1988). Causal networks: Semantics and expressiveness. In *Proceedings of the Fourth Conference on Uncertainty in Artificial Intelligence*. Amsterdam: Elsevier.

Verma, T., & Pearl, J. (1991). Equivalence and synthesis of causal models. In *Proceedings of the Sixh Annual Conference on Uncertainty in Artificial Intelligence*, *6* (p. 255). Amsterdam: Elsevier.

Zaveri, M. S., & Hammerstrom, D. (2010). CMOL/CMOS implementations of Bayesian polytree inference: Digital and mixed-signal architectures and performance/price. *IEEE Transactions on Nanotechnology*, *9*(2), 194–211.

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

Zhang, N. L., & Kocka, T. (2004). Efficient learning of hierarchical latent class models. In *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence* (pp. 585–593). Piscataway, NJ: IEEE.

Received November 9, 2015; accepted April 21, 2016.