Topology Learning of unknown Networked Linear Dynamical System excited by Cyclostationary inputs

Harish Doddi\textsuperscript{a}, Deepjyoti Deka\textsuperscript{b}, Murti V. Salapaka\textsuperscript{c}

\textsuperscript{a}Department of Mechanical Engineering, University of Minnesota Twin Cities
\textsuperscript{b}Los Alamos National Laboratory
\textsuperscript{c}Department of Electrical and Computer Science Engineering, University of Minnesota Twin Cities

Abstract

Topology learning of networked dynamical systems is an important problem with implications to optimal control, decision-making over networks, cybersecurity and safety. The majority of prior work in consistent topology estimation relies on dynamical systems excited by temporally uncorrelated processes. In this article, we present a novel algorithm for guaranteed topology learning of networks that are excited by temporally (colored) cyclostationary processes, which encompasses a wide range of temporal correlation including wide-sense stationarity. Furthermore, unlike prior work, the framework applies to linear dynamic system with complex valued dependencies, and leverages group lasso regularization for effective learning of the network structure. In the second part of the article, we analyze conditions for consistent topology learning for bidirected tree networks when a subset of the network is unobserved. Here, the full topology along with unobserved nodes are recovered from observed node’s time-series alone. Our theoretical contributions are validated on simulated data as well as on real-world climate data.

Key words: Structure Learning, Group-Lasso Regularization, Latent Nodes, Cyclostationary process

1 INTRODUCTION

Network representations often form an integral part in modeling the behavior of complex systems; examples include power grids [9], thermal management systems [34], neuronal networks in the brain [2], climate networks [17], social networks [27] and the finance networks [8]. Network representations constitute multiple agents which interact amongst each other constrained by a graph topology. Significant insights on the temporal and spatial evolution of the system dynamics can be gleaned using the abstractions of a network which can be employed for efficient resource management, control of assets and fault diagnosis. For instance, in epidemiology, understanding the spatial evolution of the virus spread is of interest for detecting the initial source, predicting the future impact, implementing control measures, and to limit or eradicate the virus. An important application that will be visited later in this article is about ocean currents, which are bulk movement of water from one region to another in the ocean and has a network representation. The regions are considered as nodes of the network and the current direction is represented as a network edge. Ocean currents affect the Earth’s climate, navigation, rescue operations in ocean, coastal climate and economy.

Approaches for learning the dependency structure of a network of dynamical systems is broadly classified into two categories: passive and active approaches. In the active learning approach, planned interventions/alterations are made to the network and the constituent agents, and the effects of the changes introduced are studied to identify network structure [22]. Passive learning, also termed non-invasive learning, learns the interaction structure without interfering with the functioning of the system [13, 25]. Among others, time-series data from agents can be analyzed for exact structure learning (topology) using multivariate Wiener filtering [20, 31] and directed information graphs [26]. Note, our article focuses on learning the topology or the exact interaction graph among agents of dynamical system, and not restricted to learning the moral graph
The transfer function $h_{ij} \in \mathbb{C}$ is the $i^{th}$ nodal time-series, $h_{ij}[k]$ is the complex-valued interdependencies, and $e_{i}(t)$ is an temporally correlated exogenous input (unmeasured). The transfer function $h_{ij}(z) := Z[h_{ij}(k)]$ with $h_{ij}(z) = 0$, $i, j \in \{1, \cdots , m\}$. The set of $\{x_{i}(k), e_{i}(k)\}_{i = 1}^{m}$ are JWSCS of period $T$ [33]. We refer to (1) as a networked Linear Dynamical System (LDS) and assume it is stable. We emphasize that the model (1) is applicable to systems where linearized models around an operating point suffice to provide fidelity to reality. Section 2 of [31] provides examples of (1). However, the models considered in [11, 33] considered real-valued interdependencies, while it is complex-valued in our article. We will use $z$ and $\omega$ interchangeably where $z = e^{i\omega}$, $\omega \in [0, 2\pi)$.

The generative graph $G = (V, \mathcal{E})$ of (1) is given by a vertex set $V = \{1, \ldots , m\}$ and a directed edge set $\mathcal{E} = \{(i, j) \in V^2 : e_{ij} \neq 0\}$.
{(i,j)|h_{ij}(z) \neq 0, i,j \in V}. Node j is (i) a parent of i if (i,j) \in \mathcal{E}, (ii) a child of i if (j,i) \in \mathcal{E}, and (iii) a spouse of i if \exists k \in V \setminus \{i,j\} such that (k,j), (k,i) \in \mathcal{E}. The children set of j \in V is denoted by \mathcal{C}_G(j), the parent set by \mathcal{P}_G(j), and the spouse set by \mathcal{K}_G(j). The topology of \mathcal{G} is an undirected graph \mathcal{G}_T = (V, \mathcal{E}_T), where \mathcal{E}_T := \{(i,j) \in V \times V | i \in \mathcal{C}_G(j) \cup \mathcal{P}_G(j)\}. The moral graph of \mathcal{G} is \mathcal{G}_M := (V, \mathcal{E}_M), where \mathcal{E}_M := \{(i,j) \in V \times V | i \in \mathcal{C}_G(j) \cup \mathcal{P}_G(j) \cup \mathcal{K}_G(j)\}. Nodes i, j are in a kin relationship if i \in \mathcal{C}_G(j) \cup \mathcal{P}_G(j) \cup \mathcal{K}_G(j). Note that \mathcal{E}_T \subset \mathcal{E}_M, and \mathcal{E}_M \setminus \mathcal{E}_T is a set of strict spouses in \mathcal{G}. An undirected path between i and j in \mathcal{G}_T is defined as a collection of distinct nodes \{\pi_1, \ldots, \pi_l\} such that \{(i, \pi_1), (\pi_1, \pi_2), \ldots, (\pi_l, j)\} \subset \mathcal{E}_T. The degree of node i in \mathcal{G}_M is \left|\mathcal{N}_G(i)\right|. Two nodes i, j \in V are said to be strict spouses if j \in \mathcal{P}_G(C_G(i)) but i \notin \mathcal{N}_G(j). For a cyclostationary process x_{ij}(t) of period T, the T-dimensional vector WSS lifted process is \mathbf{X}_i(k) := \left[x(kT), \ldots, x(kT + T - 1)\right]' for all k \in \mathbb{Z}. The networked LDS (1) is written in terms of vector WSS processes by lifting \{x_{ij}(k)\}_{i,j} as below:

\[ X_i(k) = \sum_{j=1}^{m} (H_{ij} \star X_j)(k) + E_i(k), \quad \text{where} \]

\[ E_i(k) = \left[ e_1(kT), e_1(kT + 1), \ldots, e_1(kT + T - 1) \right]' \quad \text{and} \quad H_{ij}(z) = D(z)h_{ij}(z), \quad z = \left[z^0, z^{-1}, \ldots, z^{-(T-1)}\right]' \quad \text{and} \quad s(z) = \left[z^0, z^{-1}, \ldots, z^{-(T-1)}\right]' \quad \text{and} \quad s(z) = \left[z^0, z^{-1}, \ldots, z^{-(T-1)}\right]' \]

The z-transform of (2) is \mathbf{X}_i(z) = \sum_{j=1}^{m} H_{ij}(z)X_j(z) + E_i(z). The proof of (2) is given in Lemma 15 in Appendix A. By stacking the vector processes of (2) in z-domain, we get:

\[ z - \text{domain: } \mathbf{X}(z) = \mathcal{H}(z) \mathbf{X}(z) + \mathbf{E}(z), \quad \text{where} \quad \forall i, j \in V, \quad \text{the } (i,j) \text{th block of } \mathcal{H}(z) \text{ is } H_{ij}(z), \quad \mathbf{X} = [X_1, \ldots, X_m]' \quad \text{and} \quad \mathbf{E} = [E_1, \ldots, E_m]' \quad \text{and} \quad \text{since cyclostationary processes } \{e_i(k)\}_{i=1}^{m} \text{ are mutually uncorrelated, it follows that } \mathbf{H}(z) = \text{block diagonal}. \quad \text{Assume we have } (3) \text{ is well-posed and topologically detectable, that is, } \Phi^{-1}(\omega) > 0 \quad \text{and} \quad \mathbb{I}_{m \times m} - \mathbf{H}(\omega) \text{ is invertible almost surely for } \omega \in [0, 2\pi), \quad i = 1, \ldots, m. \] It follows from (2), a directed edge from i to node j exists in \mathcal{G} if and only if H_{ij} \neq 0 in (2). Thus, LDG for the lifted vector WSS processes is identical to the LDG of (1) [33]; we provide an algorithm to learn the \mathcal{G}_T using the lift vector WSS processes.

3 Learning \mathcal{G}_T for a fully observed network

Given \{x_{ij}(k) \in \mathbb{C}, k \in \mathbb{Z}\}, for all i \in V of networked LDS (H(z), E) described by (3) which is well-posed and topologically detectable, determine \mathcal{G}_T(V, \mathcal{E}_T). Directed loops in \mathcal{G}(V, E) are admissible. We provide the topological learning of \mathcal{G}_T in terms of properties of inverse power spectral density of X in (3) and the multivariate Wiener filtering. The learning of \mathcal{E}_T involves the reconstructing \mathcal{E}_M and then eliminating the strict spouse edges (\mathcal{E}_M \setminus \mathcal{E}_T) from \mathcal{E}_M. The reconstruction of \mathcal{G}_M(V, \mathcal{E}_M) follows from the support structure of \Phi^{-1}(\omega), as shown in Section III of [11]. From (13) in [11], it follows that,

\[ B_j' \Phi^{-1}_X B_i = -\Phi^{-1}_E H_{ij} - H_{ij}' \Phi^{-1}_E + \sum_{k=1}^{m} (H_{kj})' \Phi^{-1}_E (H_{ki}). \quad (4) \]

If \[ B'_j \Phi^{-1}_X B_i \neq 0, \] then i \in \mathcal{C}_G(j) \cup \mathcal{P}_G(j) \cup \mathcal{K}_G(j) (kin relationship). However, the converse is not guaranteed and such cases are pathological (see [11,33]). Consider an edge set \mathcal{E}_M := \{(i,j)|B_j' \Phi^{-1}_X B_i \neq 0\}. Thus, \mathcal{E}_M = \mathcal{E}_M. However, the number of spurious edges in \mathcal{E}_M, that is, \mathcal{E}_M \setminus \mathcal{E}_T, can be substantial.

3.1 Identifying strict spouses in \mathcal{G}(V, \mathcal{E})

To provide a sufficient condition for identifying strict spouses in \mathcal{G}, we make the following assumption on the number of strict spouses in the generative graph.

Assumption 1 In the generative graph \mathcal{G} of the networked LDS of (1), for any two strict spouses i and j, the set of all common children \{k|k \in \mathcal{C}_G(j) \cap \mathcal{C}_G(i)\} has a cardinality of at most 1.

Assumption 1 is not necessary when all the interdependencies in (1) are strictly real-valued, a scenario presented in [11]. The above assumption is satisfied by a large class of directed graphs, such as with (i) tree topologies (undirected connected graph with no cycles), (ii) loopy topologies, with every loop has a size greater than four. Such networks are common in infrastructure networks such as power and gas grids, as the practical loop size is much greater than four. If Assumption 1 is not satisfied, then the reconstructed topology would contain spurious edges that are restricted to within two-hop distance from a node in \mathcal{G}.

Theorem 2 Consider a well-posed and topologically detectable networked LDS described by (1), with its equivalent representation (H(z), E) as in (3), its associated LDG \mathcal{G}, topology \mathcal{G}_T and satisfying Assumption 1. Let i, j \in V, are strict spouses in \mathcal{G}(V, \mathcal{E}). Suppose \mathcal{E}_ji \in \mathbb{C}^{T \times 1} be the eigenvalues of \[ B_j' \Phi^{-1}_X (\omega) B_i. \] Then, \mathcal{E}_ji is a constant, independent of \omega, for l \in \{1, 2, \ldots, T\} and for all \omega \in [0, 2\pi).

PROOF. Consider j \in \mathcal{P}_G(C_G(i)) and i \notin \mathcal{N}_G(j), then H_{ij}(\omega) = H_{ji}(\omega) = 0_{T \times T} for all \omega \in [0, 2\pi). Using (4),
For identifying strict spouses in generative graphs when multiple common children between strict spouses are allowed, that is, Assumption 1 is relaxed, we make the following assumption to develop a condition for identifying the strict spouses.

**Assumption 3** For $h_{ki}(\omega), h_{kj}(\omega) \neq 0$, then $\frac{h_{kj}(\omega)}{h_{ki}(\omega)}$ is a constant for all $\omega \in [0, 2\pi)$.

Define $a_{kij} := \frac{h_{ki}(\omega)}{h_{kj}(\omega)}$ for $h_{ki}(\omega), h_{kj}(\omega) \neq 0$. From Assumption 3, it follows that for strict spouses $i, j \in \mathcal{V}$, the $a_{kij}$ is a constant for all $\omega \in [0, 2\pi)$. Assumption 3 is satisfied by a large class of dynamical systems, including but not restricted to the generative models considered in [10, 11, 20, 31–33]. From the term $a_{kij} = \frac{\Phi_{\omega}^{-1}(\omega) H_{kj}(\omega)}{\Phi_{\omega}^{-1}(\omega) H_{ki}(\omega)}$ in (4), it follows that Assumption 3 holds trivially for a network with $\{a_{kij} \in \mathbb{R}\}$, a scenario considered in [10, 11, 20, 31–33]. An illustration is shown in Appendix B. However, for complex-valued interdependencies in the generative model, Assumption 3 is a sufficient condition for identifying strict spouse edges without restricting the structure of the generative graph. Moreover, if the network comprises of WSS processes (time-period $T = 1$), the $\Phi_{\omega}^{-1}(\omega)(j, i)$ is a scalar quantity (not a matrix). Then, for strict spouses $i, j \in \mathcal{V}$, $\bigcap_{\omega \in [0, 2\pi)} \Phi_{\omega}^{-1}(\omega)(j, i)$ is a constant $\theta$ for all $\omega \in [0, 2\pi)$, a result presented in [10, 31, 32].

**Theorem 4** Consider a well-posed and topologically detectable networked LDS described by (1), its representation $(\mathcal{H}(z), E)$ as in (3), associated LDG $\mathcal{G}'_{\mathcal{T}}$, and topology $\mathcal{G}_{\mathcal{T}}$, while satisfying Assumption 3. Suppose, for $i, j \in \mathcal{V}$, $a_{kij} = \frac{h_{ki}(\omega)}{h_{kj}(\omega)}$ holds for all $k_1, k_2 \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j), k_1 \neq k_2$. Let $\mathcal{E}_{ji} \in \mathbb{C}^{T \times 1}$ be the eigenvalues of $B'_i \Phi_{\omega}^{-1}(\omega) B_i$, where $X(k)$ is the output of the networked LDS (3). Then, $\mathcal{E}_{ji}(l)$ is a constant $c_{ji}$ for all $l \in \{1, 2, \cdots, T\}$, $\omega \in [0, 2\pi]$, if and only if $j \in \mathcal{P}_{G}(C_{G}(i))$ but $i \notin \mathcal{N}_{G}(j)$, that is, $i, j$ are strict spouses.

**PROOF.** Suppose $a_{kij} = r_k e^{i\theta_k}$ for all $k \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j)$. Under the stated assumption, we have $a_{kij} = \frac{h_{ki}(\omega)}{h_{kj}(\omega)}$ for any distinct $k_1, k_2 \in \{k \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j)\}$. Then, $\theta_k$ is independent of $k$. Let $\theta_{kij} = \theta$ for all $k \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j)$. For strict spouses $i, j$, we have,

$$B'_i \Phi_{\omega}^{-1}(\omega) B_i = \sum_{k \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j)} [a_{kij}] h_{ki}(\omega)[D(\omega)\Phi_{\omega}^{-1}(\omega) D(\omega)]$$

$$= \sum_{k \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j)} r_k e^{i\theta_k} [D(\omega)\Phi_{\omega}^{-1}(\omega) D(\omega)]$$

$$= e^{i\theta} \sum_{k \in \mathcal{C}_{G}(i) \cap \mathcal{C}_{G}(j)} r_k [D(\omega)\Phi_{\omega}^{-1}(\omega) D(\omega)]$$

$$= e^{i\theta} \times \text{Positive Definite Hermitian.}$$

Hence, $\mathcal{E}_{ji}(l) = \theta$, $\forall l \in \{1, 2, \cdots, T\}$.

Here, $c_{ji} := \theta$. The converse of the Theorem 4 holds, except for pathological cases. The proof is similar to Theorem 17. Therefore, the consequences of Theorem 2 hold even though Assumption 1 is violated.

**3.3 Connection between multivariate Wiener filter and Power spectral density**

The connection between inverse power spectral density and the multivariate Wiener filtering for topology learning of cyclostationary processes has been explored in [11, 33]. Using (12) from [11], we have,

$$B'_i \Phi_{\omega}^{-1}(\omega) B_i = \mathcal{E}_{ji}(\omega),$$

where $\mathcal{E}_{ji}(k) = X_j(k) - \sum_{l=1}^{m} e_{ij}(l) W_{jl} X_i(l)$. Using (6) in (4), the moral graph $\tilde{\mathcal{G}}_{M}$ can be reconstructed using
the condition $\mathbf{W}_j(\omega) \neq 0$ instead of $\mathbf{B}_j^T \Phi^ {-1}(\omega) \mathbf{B}_i \neq 0$. Note that $\Phi^{-1}_{E_1}$ in (6) is positive definite and its eigenvalues are real. From (6), the phase of eigenvalues of $\mathbf{W}_j(\omega)$ is equal to phase of eigenvalues of $\mathbf{B}_j^T \Phi^{-1}(\omega) \mathbf{B}_i + \pi$.

3.4 Regularized Wiener Filter Estimate:

Wiener filter estimation from the data is a necessary step for both fully observability and partially observability of the network. The theoretical results provided in this article are based on the asymptotic Wiener filter estimator. Sparse regression techniques such as Group Lasso regularization [4] is used in the Wiener filter computation for topology reconstruction in the high dimensional setting. So, we propose a block-sparsity Lasso regularized problem to promote the sparsity among the $T \times T$ blocks in the Wiener filter estimate matrix. The regularized Wiener filter can be estimated both in time-domain as well as in frequency domain.

**Time-domain regression:** The following time-domain optimization is proposed for a nodal time-series trajectory $\{X_j(k) \in \mathbb{C}\}$:

$$
\min_{\mathbf{W}_j} \mathbb{E} \left[ \|X_j(k) - \sum_{i=1, i \neq j}^m \sum_{L=-L_1}^{L=L_1} \mathbf{W}_{ji} X_i(k-L) \|_2^2 \right] + \gamma \sum_{i=1, i \neq j}^m \sum_{L=-L_1}^{L=L_1} \| \mathbf{W}_{ij} \|_F^2,
$$

where, $\gamma$ is the regularization parameter, $L_1$ is the maximum lag, and $\| \cdot \|_F$ is the Frobenius norm. Both $\gamma$ and $L_1$ are predetermined constants tuned based on historical data. From the optimal solution of (7), we construct $\hat{\mathbf{W}}_j(\omega) = \sum_{L=-L_1}^{L=L_1} \mathbf{W}_{ji} e^{-j\omega L}$, and use $\hat{\mathbf{W}}_j(\omega)$ as a surrogate to $\hat{\mathbf{W}}_j(\omega)$. In our preliminary work [11], no regularization is employed ($\gamma = 0$), and thus (7) is solved as $T$ independent scalar optimization problems. We remark that it requires relatively small number of samples when (7) is solved with regularization ($\gamma \neq 0$). Note that (7) cannot be solved as $T$ independent scalar optimization problems due to the coupling present in the regularizer term. In Section 5, we demonstrate the performance of reconstructing topology based on (7) for an ocean current series.

**Frequency-domain regression:** Consider a nodal time-series trajectory $\{X_j(k) \in \mathbb{C}\}$, $k \in \{1, \ldots, n \times N\}$ containing $n \times N$ timestamps for a node $j \in \mathcal{V}$. We split the nodal trajectory $\{X_j(k)\}_{k=1}^{n \times N}$ into $n$ sample trajectories each of length $N$. Let the $r^{th}$ sample trajectories is denoted by $\{X_j^r(k) \in \mathbb{C}\}_{k=1}^{N}$, $r \in \{1, \ldots, n\}$. For the $r^{th}$ sample trajectory, define the Discrete Fourier Transform $\text{DFT}_{k=1}^{N}$ as

$$
\hat{X}_j^r = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} X_j^r(k) e^{-j\omega k}, \quad \hat{X}_j^r = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} X_j^r(k) e^{-j\omega k},
$$

where, $X_j^r = [(X_j^1)^r, \ldots, (X_j^{N-1})^r, (X_j^{N+1})^r, \ldots, (X_j^n)^r]^T$, $r \in \{1, n\}$. Construct $\mathcal{Y} = [\mathcal{X}_j^1, \ldots, \mathcal{X}_j^n]^T \in \mathbb{C}^{n \times T}$ and $\mathcal{X} = [\mathcal{X}_j^1, \ldots, \mathcal{X}_j^n]^T \in \mathbb{C}^{n \times (m-1)T}$ respectively. For finite $N$, we propose a regularized version of Wiener filter estimate:

$$
\hat{\mathbf{W}}_j(\omega, \lambda) = \arg \min_{\beta \in \mathbb{C}^m} \frac{1}{2m} \| \mathcal{Y} - \mathcal{X} \beta \|_2^2 + \lambda \sum_{i=1}^{m} \| \beta_i \|_F,
$$

where, $\lambda > 0$ is the regularization parameter, $\| \cdot \|_F$ is the Frobenius norm of a matrix. The entries of the matrix $(\hat{\beta}_i')$ belong to a group and the regularization term in (9) promotes block sparsity in $\mathbf{W}_j$. We demonstrate the applicability of (9) in reconstructing the network of ocean currents from sea surface temperatures (SST).

4 Topology Learning of Cyclostationary time-series with Latent nodes

We focus on exact topology learning when only a subset of nodes in the network are observed (partial observability), $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is bidirected and $\mathcal{G}_T$ is tree, with complex-valued interdependencies. Suppose $\mathcal{V}_0 := \{1, \ldots, m\}$ be the set of observed nodes and $\mathcal{V}_h := \{m+1, \ldots, n\}$ be the set of latent nodes. Given $\{x_i(k), k \in \mathbb{Z}\}$ for all $i \in \mathcal{V}_0 \subset \mathcal{V}$ of networked LDS (\\(\mathbb{H}(z), E)\\) described by (3) which is well-posed and topologically detectable, we determine $\mathcal{G}_T(\mathcal{V}, \mathcal{E}_T)$. WLOG, suppose $\{x_i(k)\}_{i=1}^{m}$ are observed processes and $\{x_i(k)\}_{i=m+1}^{n}$ are the latent processes. The exogenous inputs $\{e_i(k)\}_{i=1}^{n}$ are mutually uncorrelated WSCS of period $T := \text{LCM}(T_1, \ldots, T_n)$.

**Assumption 5** The time-period $T_i$, for all $l \in \{m + 1, \ldots, n\}$, can be written as $\frac{T_i}{n}$ for some $n \in \mathbb{N}$.

By Assumption 5, we have $T = \text{LCM}(T_1, \ldots, T_m, \ldots, T_n) = \text{LCM}(T_1, \ldots, T_n)$. Then, the collection $\{x_i(k), e_i(k)\}_{i=1}^{n}$ are jointly cyclostationary with period $T$, which can be solely computed from observed nodal time-series. Assumption 5 is not required in Section 3 because all the nodes are observable. All processes $\{x_i(k), e_i(k)\}_{i=1}^{n}$ are lifted to a $T$-dimensional vector WSS process $\{X_i(k), E_i(k)\}_{i=1}^{n}$. The network dynamics of $\{X_i(k)\}_{i=1}^{n}$ is

$$
\begin{bmatrix}
X_o(k) \\
X_h(k)
\end{bmatrix} =
\begin{bmatrix}
H_{oo} & H_{oh} \\
H_{ho} & H_{hh}
\end{bmatrix}
\begin{bmatrix}
X_o(k) \\
X_h(k)
\end{bmatrix} +
\begin{bmatrix}
E_o(k) \\
E_h(k)
\end{bmatrix},
$$

\[1\] computed at frequency $\omega = \frac{2\pi l}{N}$, $l \in \{0, \ldots, N-1\}$ unless explicitly mentioned.
where \( X_o = [X_1, \cdots, X_m]^\top \), \( X_h = [X_{m+1}, \cdots, X_n]^\top \),
\[
E_o = [E_1, \cdots, E_m]^\top, E_h = [E_{m+1}, \cdots, E_n]^\top
\]
The dynamics (10) is written in a compact form as,
\[
X(k) = H(z) * X(k) + E(k). \tag{11}
\]
Here, \( H_{oo} \) is \( mT \times mT \) transfer matrix representing the interactions from the observed nodes to all the observed nodes. \( H_{oh} \) is of size \( mT \times (n - m)T \) represents the interactions from the latent nodes to all the observed nodes. \( H_{ho} \) is of size \( (n - m)T \times mT \), and it represents the interaction term from the observed nodes to all the latent nodes. Define \( E = \{(i, j) | i, j \in V, \mathbb{H}(i, j) \neq 0 \} \). We denote the generative graph associated with (11) by \( G(V, E) \) and its topology by \( G_T(V, E_T) \). Further, the restriction of \( G(V, E) \) to the observed nodes is denoted by \( G_o(V_o, E_o) \), where \( E_o = \{(i, j) | i, j \in V_o, (i, j) \in E \} \). The remaining directed edges in \( E \) are associated with hidden nodes, denoted by \( E_h := E \setminus E_o \). The topology \( G_T \) restricted to \( V_o \) is \( G_{T_o}(V_o, E_{T_o}) \), where the undirected edge set \( E_{T_o} := \{(i, j) | i, j \in V_o, (i, j) \in E \} \). Define \( E_{T_h} := E \setminus E_{T_o} \).

Given the nodal observed time-series \( \{x_i(t)\}_{t=0}^m \), we determine the topology \( G_T(V, E_T) \). Our framework for learning \( G_T \) comprises reconstructing (a) the edges between observed nodes \( G_{T_o}(V_o, E_{T_o}) \) (b) find hidden nodes \( (V_h) \) and (c) the connections to hidden nodes and the observed nodes \( (E_{T_h}) \). From (11), it follows that,
\[
\Phi^1_{X} = \begin{bmatrix} \Phi_{oo}(z) & \Phi_{oh}(z) \\ \Phi_{ho}(z) & \Phi_{hh}(z) \end{bmatrix}^{-1} = \begin{bmatrix} J_{oo}(z) & J_{oh}(z) \\ J_{ho}(z) & J_{hh}(z) \end{bmatrix} \tag{12}
\]
Using the matrix inversion lemma [16] in (12), it follows that, \( \Phi^1_{oo} = J_{oo} - J_{oh}J_{hh}^{-1}J_{ho} =: \Gamma + \Delta + \Sigma\), where,
\[
\Gamma = (I - H_{oo})\Phi^{-1}_{hh}(I - H_{ho}), \quad \Delta = H_{ho}\Phi^{-1}_{hh}H_{ho}, \quad \Sigma = -\Psi^{*}\Lambda^{-1}\Psi \tag{13}
\]
where, \( \Lambda = H_{oh}\Phi^{-1}_{hh}H_{oh} + \Phi_{hh}^{-1}\Psi = H_{ho}\Phi^{-1}_{hh}(I - H_{oo}) + \Phi_{hh}^{-1}H_{ho} \).

The following Lemma 6 relates the \( G_T \) to the structure of \( \Phi^1_{oo} \).

**Lemma 6** The following assertions hold

1. For a \( i, j \in V_o \), if (i) there does not exist a \( k \in V_o \setminus \{i, j\} \) such that \( i - k - j \) is in \( G_T \), and (ii) \( i - j \) is not in \( G_T \), then \( B_i^{*} \Delta B_j = 0_{T \times T} \).
2. For a \( i, j \in V_o \), there does not exist a \( l \in V_h \) such that \( i - l - j \) is in \( G_T \), then \( B_i^{*} \Delta B_j = 0_{T \times T} \).

(3) For a \( l_1, l_2 \in V_h \), there does not exist an \( i \in V_o \) such that \( l_1 - i - l_2 \) is in \( G_T \), then \( \Lambda \) is block diagonal and Hermitian.

(4) Suppose in \( G_T \), for \( j \in V_o \) and \( l \in V_h \); (i) \( j - l \) is not present and (ii) there is no path of the form \( j - p - l \) with \( p \in V_o \), then \( B_j^{*} \Psi B_l = 0_{T \times T} \).

(5) Suppose \( \Lambda \) is block diagonal and Hermitian and if in \( G_T \), for \( i, j \in V_o \) and \( l \in V_h \), there are no paths of the form \( i - p - l \) or \( i - l \) and \( j - p^* - l \) or \( j - l \) for any \( p \in V_o \setminus \{i, j\} \), then \( B_i^{*} \Sigma B_j = 0_{T \times T} \).

**PROOF.** The proof is similar to the proof of Lemma 3.1 from [33], but extended to complex-valued cyclostationary processes. Its important to note the salient points (i) for \( i, j \in V_o \), \( H_{ij} := H_{oo}(i, j) \) is a \( T \times T \) transfer matrix and \( H_{ii} = 0 \) (ii) \( \Phi_{E_o}^{-1}, \Phi_{E_h}^{-1}, \Phi_{E_h}^{-1} \) are block diagonal Hermitian matrices of size \( nT \times nT \), \( mT \times mT \) and \( (n - m)T \times (n - m)T \), respectively.

(1) It follows from (13),
\[
B_i^{*} \Lambda B_j = -B_i^{*} \Phi_{E_o}^{-1}H_{ho}B_j + B_i^{*} \Phi_{E_h}^{-1}H_{ho}B_j + B_i^{*} \Phi_{E_h}^{-1}H_{ho}B_j + \sum_{k=1}^{m} H_{kl} \Phi_{E_h}^{-1}H_{ho}.
\tag{14}
\]
Here, \( i, j \in V_o \). If \( (i, j) \notin E_o \), then \( H_{ij} = 0_{T \times T} \). Similarly, if \( (i, j) \notin E_o \), then \( H_{ij} = 0_{T \times T} \). In \( G_o \), if there does not exist \( k \in V_h \setminus \{i, j\} \), such that \( \{(k, i), (k, j)\} \in E_h \), then the third term is \( 0_{T \times T} \).

(2) From (13), it follows that,
\[
B_i^{*} \Delta B_j = \sum_{l \in V_h} [H_{oo}(l, i)]^{*} \Phi_{E_o}^{-1}(l, l)H_{ho}(l, j).
\]
For a given \( i, j \in V_o \), if there does not exist a \( l \in V_h \), such that \( \{(l, i), (l, j)\} \in E_h \).

(3) Consider two distinct hidden nodes \( l_1, l_2 \in V_h \). If there does not exist an \( i \in V_o \), such that \( \{(i, l_1), (i, l_2)\} \in E_h \), then \( H_{oh}(i, l_1) = H_{oh}(i, l_2) = 0_{T \times T} \). Thus, from (13), it follows that \( B_i^{*} \Delta B_j = \sum_{l \in V_h} [H_{oo}(l, i)]^{*} \Phi_{E_o}^{-1}H_{ho}(l, j) \neq 0_{T \times T} \). The diagonal block of \( \Delta \) is given by \( B_i^{*} \Delta B_i = \sum_{l \in V_h} [H_{oo}(l, i)]^{*} \Phi_{E_o}^{-1}H_{ho}(l, i) \neq 0_{T \times T} \). This is Hermitian of size \( T \times T \), thus, \( \Lambda \) is block diagonal Hermitian.

(4) Suppose \( (l, j) \notin E_o \), then \( H_{oh}(l, j) = H_{ho}(l, j) = 0_{T \times T} \). If there does not exist a \( p \in V_o \), such that \( \{(p, j), (p, l)\} \in E \), then \( H_{ho}(p, l) = H_{ho}(p, j) = 0_{T \times T} \). Thus, from (13), the \( (l, j)^{th} \) block of \( \Psi \) is given by \( B_j^{*} \Psi B_l = [H_{oh}(j, l)]^{*} \Phi_{E_o}^{-1} - \sum_{p=1}^{m} [H_{ho}(p, l)]^{*} \Phi_{E_o}^{-1}H_{ho}(p, j) + \Phi_{E_h}^{-1}(l, l)H_{ho}(l, j) = 0 \).
(5) For a given \( i, j \in V_o \), and for any \( l \in V_h \) the following holds (i) \( B'_i \Phi B_j = 0 \) if there does not exist \( i \leftarrow l, i \rightarrow l \) and \( i \rightarrow p \leftarrow l \) for any \( p \in V \setminus i \) in \( \mathcal{E} \).
(ii) \( B'_i \Phi B_j = 0 \) if there does not exist \( j \leftarrow l, j \rightarrow l \) and \( j \rightarrow p \leftarrow l \) for any \( p \in V \setminus j \) in \( \mathcal{E} \). Thus, \( B'_i \Sigma B_j = 0_{T \times T} \).

Since, we are restricting our attention to bidirected networks, we make the following assumption.

**Assumption 7** If \( H_{ij}(z) \neq 0 \), then \( H_{ij}(z) \neq 0 \).

**Assumption 8** The hidden nodes in \( G_T \) are at least four or more hops away from each other.

**Theorem 9** Consider a linear dynamical system with topology \( G_T \) such that Assumptions 7, 5 and 8 hold. Then \( B'_i \Phi B_j = 0 \) for all \( \omega \in [0, 2\pi) \), implies that, \( i \) and \( j \) are within four hops of each other in the graph \( G_T \).

**Proof.** Given that \( B'_i \Phi B_j = 0 \), then either (i) \( B'_i B_j \neq 0 \) or (ii) \( B'_i \Delta B_j \neq 0 \) or (iii) \( B'_i \Sigma B_j \neq 0 \). The proof is by enumerating all the possible cases as follows.

(i) \( B'_i \Gamma B_j \neq 0 \) implies that either \( i \leftarrow j \) or \( i \rightarrow j \) or \( i \rightarrow p \leftarrow j \) exists in \( \mathcal{E}_o \), for some \( p \in V \setminus \{i, j\} \). This is evident from 1) of Lemma 6.

(ii) From 2) of Lemma 6, \( B'_i \Delta B_j \neq 0 \) implies there exists a \( l \in V_h \) such that \( i \rightarrow l \rightarrow \downarrow j \) exists in \( \mathcal{E}_h \).

(iii) From 3), 4) and 5) of Lemma 6, \( B'_i \Sigma B_j \neq 0 \), implies that there exists a hidden node \( l \in V_h \) such that \( l \rightarrow \downarrow \leftarrow j \) exists in \( \mathcal{E}_h \) for some \( p \in V \setminus i \) and (b) either \( j \leftarrow l \) or \( j \rightarrow l \rightarrow \downarrow p \rightarrow l \) exists in \( \mathcal{E} \) for some \( p \in V \setminus j \).

In (i), (ii) and (iii), the nodes \( i \) and \( j \) are within four hops of each other in \( G_T \).

**Remark 10** From proof of Theorem 9, it follows that if an observable node \( i \) is more than 2 hops away from any hidden node, then, for all \( j \in V_o \), i, the following holds: (i) \( B'_i \Delta B_j = B'_i \Sigma B_j = 0 \), and (ii) \( B'_i \Phi B_j = 0 \) implies that \( B'_i B_j = 0 \) (no hidden node contribution).

An instance of the transfer function matrix \( \mathbb{H}(z) \) can be designed so that converse of Theorem 9 does not hold. However, such cases are pathological and the converse of Theorem 9 holds for all practical purposes. Thus, based on Theorem 9, we construct an undirected graph based on the structure of \( \Phi_{oc}^{-1} \) as follows: Initialize \( \mathcal{E}_c \) as \( \{i\} \) and set \( V_o \) as \( \{1, 2, \ldots, m\} \). Construct the undirected edge set \( \mathcal{E}_c := \{i, j\} | B'_i \Phi B_j \neq 0 \). The undirected graph \( G_c := (V_o, \mathcal{E}_c) \) constitutes edges between the nodes that are within four hops of each other in \( G_T \). From Theorem 9, it follows that \( \mathcal{E}_c \subset \mathcal{E}_T \), and hence all the true edges among the observed nodes can be recovered in addition to spurious edges. We use \( \mathcal{G}_c(V_o, \mathcal{E}_c) \) in the following theorems to obtain \( \mathcal{G}_T \). We identify \( \mathcal{E}_c \), number of hidden nodes, and their neighbors solely from \( G_c(V_o, \mathcal{E}_c) \). We will make the following assumption of \( G_T \) for recovering \( \mathcal{E}_c \), number of hidden nodes, and their neighbors from \( G_c \).

**Assumption 11** \( G_T \) is tree, that is for any two nodes \( i, j \in V \), there is a unique path connecting nodes \( i \) and \( j \) in \( G_T \). Further, every hidden node is at least three hops away from all leaf nodes in \( G_T \).

Note that with Assumption 11, \( G_T \) does not posses cycles and hence, the Assumption 1 holds. Thus, Theorem 2 and Theorem 17 from Appendix C holds. The nodes with degree 1 are termed as leaf nodes \( V_l \), while other nodes in \( V \) are referred to as non-leaf nodes \( V_n \). By Assumption 11, \( V_l \in V_o \).

### 4.1 Reconstructing \( G_{T_s} \) from \( G_c \)

Given that \( G_T \) satisfies Assumption 8 and Assumption 11, we propose an algorithm for identifying the leaf and non-leaf observable nodes, and the topology restricted to observable nodes. The phase result of the eigenvalues developed in the previous section is not applicable to identify the topology restricted to observed nodes as we do not know the locations of the hidden nodes. Here we exploit graphical separation and the tree topology to identify the true edges from \( G_c \). Consider an undirected graph \( U(V_o, \mathcal{E}_u) \), and nodes \( i, j \in V_o \). The vertex set \( Z \in V_o \setminus \{i, j\} \) is said to separate nodes \( i \) and \( j \) in \( U \) if all the paths from \( i \) to \( j \) in \( U \) contains at least a node in \( Z \). If \( Z \) separates \( i \) and \( j \) in \( U \), then we say \( \text{sep}(i, Z, j) \).

The following theorem enables us to identify the non-leaf nodes and edges among them from \( G_c \).

**Theorem 12** Consider a networked LDS (11), such that Assumptions 7, 5, 8 and 11 hold. There exist distinct nodes \( a, b, c, d \in V_o \) such that \( \text{sep}(c, \{a, b, d\}) \) holds in \( G_c \) if and only if \((a, b) \in \mathcal{E}_T \) and \( a, b \) are non-leaf nodes.

**Proof.** Suppose \( a - b \) is not an edge in \( G_T \). Let \( p := c - \pi_{h,1} - \pi_1 - \pi_2 - \pi_{h,2} - \pi_3 - \cdots - \pi_m - \pi_{h,j} - d \) be the unique path between \( c \) and \( d \) in \( G_T \) such that \( \{\pi_1, \pi_2, \ldots, \pi_m\} \) are observed nodes and \( \{\pi_{h,1}, \pi_{h,2}, \ldots, \pi_{h,j}\} \) are hidden nodes. There are three possibilities - (i) neither of \( a, b \) belong to \( \{\pi_1, \ldots, \pi_m\} \), (ii) either \( a \) or \( b \) but not both belong to \( \{\pi_1, \ldots, \pi_m\} \) and (iii) both \( a \) and \( b \) belong to \( \{\pi_1, \ldots, \pi_m\} \) with \( a - b \) not being an edge.

(i) If \( a \) and \( b \) do not belong to \( \{\pi_1, \ldots, \pi_m\} \). Then \( c - \pi_1 - \pi_2 - \cdots - \pi_j - a - \pi_1 - \cdots - \pi_m - d \) is a path in \( G_c \) with no intermediate node in the path being \( a \) or \( b \). Thus, \( \text{sep}(c, \{a, b\}, d) \) does not hold, which is a contradiction.

(ii) If \( a \) belongs to \( \{\pi_1, \ldots, \pi_m\} \) but not \( b \). Let \( \pi_{k} = a \). Then \( c - \pi_1 - \pi_2 - \cdots - \pi_j - a - \pi_1 - \cdots - \pi_m - d \) is
a path in $G_e$, which is not separated by \((a, b)\). Thus, \(sep(c, \{a, b\}, d)\) does not hold which is a contradiction.
Similarly, one can arrive at a contradiction for the case $b$ belongs to \(\{\pi_1, \ldots, \pi_m\}\) but not $a$.

(iii) If both $a$ and $b$ belong to \(\{\pi_1, \ldots, \pi_m\}\), and $a - b$ is not in $G_T$. Let $a = \pi_e$ and $b = \pi_f$. Then

(a) $a, b$ are two-hop neighbors through an observed node $\pi_g$.

Let $\pi_d, \pi_l$ be an observed neighbor of $a, b$, respectively and $\pi_g$ be the common unobserved neighbor of $a, b$. Then, $c - \pi_l - \pi_d - \pi_g - \pi_l - \pi_d = d$ is a path in $G_e$, which is not separated by \(\{a, b\}\). Thus, $sep(c, \{a, b\}, d)$ does not hold and is a contradiction.

(b) $a, b$ are two-hop neighbors through an unobserved node $\pi_{h,g}$.

Let $\pi_d, \pi_l$ be an observed neighbor of $a, b$, respectively and $\pi_{h,g}$ be the common unobserved neighbor of $a, b$. Then, $c - \pi_l - \pi_d - \pi_g - \pi_l - \pi_d = d$ is a path in $G_e$, which is not separated by \(\{a, b\}\). Thus, $sep(c, \{a, b\}, d)$ does not hold and is a contradiction.

(c) Let $a, b$ are three hop neighbors with one hidden node($\pi_{h,f}$) and one observed node($\pi_{f}$) in between $a, b$.

Let $\pi_l$ be neighbor of $a$ on the other side of $b$. Similarly, let $\pi_l$ be another neighbor of $b$ on the other direction of $a$. Then, $c - \pi_l - \pi_d - \pi_g - \pi_l - \pi_d = d$ is a path in $G_e$, which is not separated by \(\{a, b\}\). Thus, $sep(c, \{a, b\}, d)$ does not hold and is a contradiction.

(d) Let $a, b$ are four hop neighbors with $\pi_f, \pi_g$ being observed neighbors of $a$ and $b$, respectively and $\pi_{h,f}$ being an unobserved neighbor of $\pi_f, \pi_g$ in $G_T$.

The path in $G_T$ is of the form $c - \pi_l - \pi_d - \pi_g - \pi_l - \pi_d = d$ is a path in $G_e$, which is not separated by \(\{a, b\}\). Thus, $sep(c, \{a, b\}, d)$ does not hold and is a contradiction.

(e) Let $a, b$ be more than four hops away such that $a - \pi_f - \pi_g - \pi_{h,g} - \pi_l - \pi_t - \pi_t - \pi_t - \pi_t - b$. Using the same reasoning as before one can show that a path exists in $G_e$, which does not contain both $a$ and $b$, that is, there exist a path which is not separated by \(\{a, b\}\) in $G_e$. Thus, $sep(c, \{a, b\}, d)$ does not hold in $G_e$ and is a contradiction.

As all cases have been exhausted we conclude that $sep(c, \{a, b\}, d)$ in $G_e$ is not possible, which is a contradiction. Hence, $a - b$ is a true edge in $G_T$. Both $a, b$ have degree at least two as they have at least two neighbors, hence, are non-leaf nodes. This proves the theorem.

The conclusion of Theorem 12 is that, if $a, b$ is a spurious edge between observable non-leaf nodes $a, b \in V_o$, then there exist no $c, d$ different from $a, b$ such that $sep(c, \{a, b\}, d)$ holds in $G_e$. This provides a graph based test to identify the true edges among non-leaf observable nodes from $G_e$. Thus, all the non-leaf nodes $V_{\ell} \subset V_o$, the edges $\{(i, j) | i, j \in V_{\ell}, (i, j) \in G_T\}$ are recovered. Since a leaf node is not hidden by Assumption 11, the set of leaf nodes are given by $V_l := V_o \backslash V_{\ell}$. Suppose $l \in V_l$, then it has a single non-leaf neighbor in $G_T$, since degree of $l$ is one. Based on Assumption 11, the node $l$ is at least three hops away from any hidden node in $G_T$. From Lemma 6 and Remark 10, the spurious edges connected with $l$ in $G_e$ include those up to its two-hop neighbors in $G_T$. Thus, the two-hop neighbors of a leaf node in $G_T$ should be identified and pruned. The following theorem is helpful for identification of two-hop neighbors of a leaf node.

**Theorem 13** Consider a networked LDS described by (11) such that Assumptions 3, 5, 7, 8 and 11 hold. Let $i \in V_l$ and $j \in V_{\ell}$ be a neighbor of $i$ in $G_e$. Suppose $E_{ij} \in \mathbb{C}^{T \times 1}$ be the eigenvalues of $B_i \Phi_1^{\top} B_j$. Then, $f_{E_{ij}}(l)$ is a constant for $l \in \{1, 2, \cdots, T\}$ and all $\omega \in [0, 2\pi)$ if and only if $i, j$ are two-hop neighbors in $G_T$.

**PROOF:** The proof follows directly from Theorem 17.

Based on Theorem 12 and 13, we propose an Algorithm 2 which reconstructs $G_{T_e}$. The Algorithm 2 is shown in Appendix E. As asserted earlier, the two-hop neighbors of a leaf node are identified using Theorem 13 when Assumption 3 holds. When Assumption 3 does not hold, the spurious edge associated with a leaf node cannot be identified, and thus, $E_{T_e} \subset E_T \setminus E_{T_e}$ is the set of two-hop neighbors of the leaf nodes. After learning $G_{T_e}$, we find $V_o$ and $E_{T_e}$ using the next theorem (note that the theorem statement appeared in a preliminary conference paper [30]without proof).

**Theorem 14** Consider a networked LDS described by (11) such that Assumptions 5, 7, 8 and 11 hold. Suppose $T_1, T_2$ are two disconnected components in $G_{T_e}(V_o, E_{T_e})$ with observed nodes $c \in T_1$ and $e \in T_2$. If for all $b \in T_1$, for all $f \in T_2$ where $b \neq c$ and $c \neq f$ are edges in true topology $G_T$, and $b, c, e$ form a clique in $G_e$, then there exists a $d \in V_h$ such that $c - d - e$ is a path in $G_T$.

**PROOF.** Since, $G_T$ satisfies Assumptions 8, it follows that $G_{T_e}(V_o, E_{T_e})$ is a union of disconnected connected components, where each component has at least three nodes. Suppose $T_1, T_2$ are two disconnected components in $G_{T_e}(V_o, E_{T_e})$ with observed nodes $c \in T_1$ and $e \in T_2$. Consider $b \in T_1, b \neq c$ and $f \in T_2, f \neq e$, such that $b \neq c$ and $e \neq f$ exists in $G_T$. From Assumptions 8 and 11, there exists an observable node $a \in T_1, g \in T_2$ such that $a - b - c$ exists and $e \neq g$ exists in $G_T$.  

8
We will show that if there is no $l \in V_l$ such that $c - l - e$ exists in $\mathcal{G}_T$, then $b, c, e, f$ cannot form a clique in $\mathcal{G}_c$. If there is no such $l$, then $a - b - c - d_1 - d - d_2 - e - f - g$ exists in $\mathcal{G}_\pi$ for some $d_1, d_2 \in V_o$ and $d \in V_h$. Node $d$ exists because $a, b, c, e \in \mathcal{F}_1$ and $e, f, g \in \mathcal{F}_2$. From Lemma 6, it is evident that $b, c, d, e$ does not form a clique in $\mathcal{G}_c$, which is a contradiction. Same holds if one $d_1$ or $d_2$ is present. Thus there does not exists $d_1$ and $d_2$, such that $a - b - c - d_1 - d - d_2 - e - f - g$ exists in $\mathcal{G}_T$. Therefore, $b - c - d - e - f$ exists in $\mathcal{G}_T$.

Let $\mathcal{G}_{T_r}$, the graph topology of observed nodes, be reconstructed using Algorithm 2. As alluded earlier, $\mathcal{G}_{T_r}$ is a union of disconnected subgraphs. Let $h$ be the number of disconnected subgraphs in $\mathcal{G}_{T_r}$. In Algorithm 3, shown in Appendix F, for each pair of disconnected subgraphs $\mathcal{F}_i, \mathcal{F}_j$, Theorem 14 is checked to identify if a hidden node exists between them (Step 5). If yes, then a hidden node is inserted and the edges to its neighbors are added. This completes the topology reconstruction of $\tilde{\mathcal{T}} = (V_o, \mathcal{E}_T)$, which is identical to $\mathcal{G}_T(V_o \cup V_h, \mathcal{E}_T)$. Note that Theorem 14 does not require Assumption 3 to hold. Consider $\tilde{\mathcal{T}}$ when Assumption 3 does not hold. The spurious edges associated with the leaf nodes are recovered in $\mathcal{E}_T$, along with the true edges of the leaf nodes. The hidden nodes and their one-hop, two-hop neighborhoods in $\mathcal{G}_T$ are identified exactly. Moreover, if a hidden node is present at a 3-hop distance from a leaf node in $\mathcal{G}_{T_r}$, then the spurious edge associated with that leaf node in $\tilde{\mathcal{T}}$ can be identified from the recovered exact two-hop neighborhood of the hidden node. For a network with directed loops or loopy topology, topology learning with latent nodes for cyclostationary processes is a challenging task and future effort. In the next section, we provide numerical results to validate the theoretical algorithms presented in Section 3 and Section 4.

5 Results

The illustrations presented in this section are based on networks with dynamic links among agents. CVX [14] is used to solve the vector optimization (7) and (9).

Ocean Current Reconstruction from SST data: Ocean currents are bulk movement of water from one region to another in the ocean and has a network representation. The regions are considered as nodes of the network and the current direction is represented as a network edge. The currents are reconstructed using monthly mean COBE-SST2 Sea Surface Temperature alone, downloaded from [1]. The nodes considered for the network reconstruction are spatially distant in the Atlantic region to begin with, and from the SST data of these nodes the edges of the network are reconstructed. The SST monthly mean data for the period 1850 - 2019 consists of 2040 data samples. $X(f)$ the FFT of the SST data at frequencies $f = \frac{2\pi k}{2040}, k = \{1, \ldots, \frac{2040}{2} - 1\}$ shows a large magnitude at 12 months, second highest at 6 months; there are components corresponding to 3 months and lower exists, but their magnitude is small. The time-period of the SST data is estimated to be 12 months, hence $T = 12$. The values of $n, N, \lambda$ in solving (9) are 42, 4, and 1.7, respectively. The threshold used in Algorithm 1 is 0.04. Fig. 1 shows the reconstructed currents along the west and east coast of Atlantic. The reconstructed currents in Fig. 1 can be identified with known ocean currents: gulf stream: 15 → 14 → 12 → 10 → 8 → 17, Benguela current: 7 → 4, Brazilian current: 1 → 3, Malvinas current: 6 → 5, Canary current: 9 → 11 → 13 → 16.

Fig. 1. Current reconstruction using frequency-domain regression presented in Subsection 3.4. Reconstructed currents are: gulf stream: 15 → 14 → 12 → 10 → 8 → 17, Benguela current: 7 → 4, Brazilian current: 1 → 3, Malvinas current: 6 → 5, Canary current: 9 → 11 → 13 → 16.

5.1 Reconstruction of a Test network

We presented consistent topology reconstruction under full and partial observability for a test network containing 50 nodes with 5 latent nodes.

Test Network: Reconstruction under full observability

We consider a generative graph consisting of observable 50 nodes, with topology $\mathcal{G}_T(V, \mathcal{E}_T)$ shown in Fig. 2(a). $\mathcal{E}_T$ includes green colored edges. All the edges (including green colored) are bidirected in the generative graph.
and hence possess directed loops, but satisfy Assumption 1. Note that generative graph is not restricted to bidirected network; the generative graph is allowed to be any directed graph satisfying Assumption 1. The time-series \( x_i(k) \) is available at each node \( i \in \{1, 2, \cdots, 50\} \). The dynamic links \( h_{ji} \) in (1) are chosen to be FIR transfer functions with complex-valued filter coefficients satisfying Assumption 3. The exogenous input \( e_i(k) \) is generated in MATLAB as \( e_i(k) = \cos(\pi k)w_1(k) \), where \( w_1(k) \) is a zero mean wide sense stationary process. \( e_i(k) \) is a cyclostationary process of period \( T_1 = 2 \). The remaining exogenous inputs are mutually uncorrelated wide sense stationary processes, uncorrelated with \( e_i(k) \). Using \( \{e_i(k)\}_{i=1}^{50} \), time-series \( \{x_i(k)\}_{i=1}^{50} \) are generated using (1). Each nodal time-series contains 628400 samples. The simulations are done using MATLAB. Based on the periodogram analysis of the \( \{x_i(k)\}_{i=1}^{50} \), we get \( T = 2 \). Each \( x_i(k) \) is lifted to a vector time-series \( X_i(k) = [x_i(2k), x_i(2k + 1)]^T \) for \( i \in \{1, \cdots, 50\} \). Solve (7) for all \( j = \{1, \cdots, 50\} \) with a predetermined constants \( \gamma = 0.1 \) and \( L_\ast = 3 \). The moral graph (\( \mathcal{E}_M \)) is reconstructed based on the condition \( H_{\infty}(\hat{W}_{ji}) + H_{\infty}(\hat{W}_{ij}) > \tau \), for \( i, j \in \{1, \cdots, 50\} \) and \( \tau = 0.03 \). Fig. 2(b) shows a color map of \( H_{\infty}(\hat{W}_{ji}) + H_{\infty}(\hat{W}_{ij}) \). Some of the spurious edges have very low color intensity and are not identified in 2(b), which reduces the computational effort of pruning them. The values of threshold \( \tau, \gamma \) and \( L_\ast \) are tuned to recover the exact topology when the number of samples per node is \( 6(10)^6 \). An edge \((j,i) \in \mathcal{E}_M \) is considered as spurious (strict spouses) if \( 0.5|\hat{\mathcal{E}}_{ji}(\omega)| + |\hat{\mathcal{E}}_{ij}(\omega)| \) is constant. Here, \( \hat{\mathcal{E}}_{ji}(\omega) \) is the vector of eigenvalues of \( \hat{W}_{ji}(\omega) \). Such edges are pruned. \( \mathcal{G}_T \) is exactly recovered after pruning the spurious edges from \( \mathcal{E}_M \).

**Reconstruction under partial observability:** Consider the generative graph consisting of 50 nodes with topology \( \mathcal{G}_T \) shown in Fig. 2(a) without green colored edges. The red colored nodes \( \{46, 47, 48, 49, 50\} \) are latent. We aim to reconstruct \( \mathcal{G}_T \) using the time-series \( \{x_i(k)\}_{i=1}^{50} \). The efficacy of the algorithms was assessed using comparisons with the truth, determined by the exact expression of power spectral density using the generative model. \( T = 2 \) based on the periodogram analysis of \( \{x_i(k)\}_{i=1}^{50} \). The lifted processes are \( X_i(k) = [x_i(2k), x_i(2k + 1)]^T \) for \( i \in \{1, \cdots, 45\} \). We solve (7) for all \( j = \{1, \cdots, 45\} \) with a predetermined constants \( \gamma = 0.1 \) and \( L_\ast = 3 \). Construct an edge set \( \mathcal{E}_e \) by placing an undirected edge between two distinct nodes \( j \) and \( i \) if \( H_{\infty}(\hat{W}_{ji}) + H_{\infty}(\hat{W}_{ij}) > \tau \), for \( i, j \in \{1, \cdots, 50\} \) and \( \tau = 0.03 \). The values of \( \tau, \gamma \) and \( L_\ast \) are tuned based on large sample size. Fig. 3 shows a color map of \( H_{\infty}(\hat{W}_{ji}) + H_{\infty}(\hat{W}_{ij}) \). Note that \( \mathcal{G}_e \) is not a moral graph of \( \mathcal{G} \). We now apply Algorithm 2 on \( \mathcal{G}_e \) to obtain \( \mathcal{G}_T(\mathcal{V}_e, \mathcal{E}_T) \). \( \mathcal{G}_T \) is identified correctly. Please see Fig. 5 for consistent topology estimation without regularizer.

6 Conclusions

For a network of multiple agents interacting according to a linear dynamic model excited by cyclostationary pro-

---

**Fig. 2.** (a) Generative graph topology \( \mathcal{G}_T(\mathcal{V}_T, \mathcal{E}_T) \): Under full observability, green colored edges are included in \( \mathcal{G}_T \). Under partial observability, red colored nodes are latent, and green colored edges are excluded from \( \mathcal{G}_T \). (b) Reconstruction under full observability: color map of \( H_{\infty}(\hat{W}_{ji}) + H_{\infty}(\hat{W}_{ij}) \) for \( i, j \in \{1, \cdots, 50\} \). Some of the spurious edges have very low color intensity. All edges in \( \mathcal{G}_T \) are identified correctly.

**Fig. 3.** Partial observability: Left figure is the magnitude of model-based Wiener filter (exact values based on generative model). Right figure is the magnitude of regularized Wiener filter obtained by solving (7) with \( \gamma = 0.07 \) and \( L_\ast = 3 \).

**Fig. 4.** Phase plot of eigenvalues (model-based) for various edges \( j - i \). Here, \( \epsilon_{ji} \) ( \( \epsilon_{ij} \)) is the largest component of the vector \( \mathcal{E}_{ji} (\mathcal{E}_{ij}) \).
Fig. 5. Reconstruction without regularization: Left figure is the magnitude of model-based Wiener filter (exact values based on generative model). Right figure is the magnitude of Wiener filter estimate without regularizer ($\gamma = 0$) and $L^* = 3$, computed from 628400 samples.

cesses, an algorithm is presented for reconstructing the topology using nodal time-series data. The learning algorithm provably recovers the true topology in the large sample limit. For bidirected networks with tree topologies under partial observability, an algorithm for topology learning is developed. The algorithm’s performance is demonstrated using test networks, as well as real-time identification of ocean current’s based on sea surface temperature data (COBE-SST2).

References

[1] National oceanic and atmospheric administration, physical sciences laboratory. psl. [https://psl.noaa.gov/data/gridded/data.cobe2.html] Accessed: 2024-07-08.

[2] José-Luis Aguirre, Ramon Brena, and Francisco J Cantu. Multiagent-based knowledge networks. Expert Systems with applications, 20(1):65–75, 2001.

[3] Igor Aizenberg. Why we need complex-valued neural networks? In Complex-Valued Neural Networks with Multi-Valued Neurons, pages 1–53. Springer, 2011.

[4] Francis R Bach. Consistency of the group lasso and multiple kernel learning. Journal of Machine Learning Research, 9(6), 2008.

[5] S. Brettreil and R. Weber. Comparison of two cyclostationary detectors for radio frequency interference mitigation in radio astronomy. Radio Science, 40(5), 2005.

[6] Ewa Broszkiewicz-Suwaj, Andrzej Makagon, Rafal Weron, and Agnieszka Wykomańska. On detecting and modeling periodic correlation in financial data. Physica A: Statistical Mechanics and its Applications, 336(1-2):196–205, 2004.

[7] Venkat Chandrasekaran, Pablo A Parrilo, and Alan S Willsky. Latent variable graphical model selection via convex optimization. In 2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pages 1610–1613. IEEE, 2010.

[8] K Tse Chi, Jing Liu, and Francis CM Lau. A network perspective of the stock market. Journal of Empirical Finance, 17(4):659–667, 2010.

[9] Deepjyoti Deka, Michael Chertkov, and Scott Backhaus. Structure learning in power distribution networks. IEEE Transactions on Control of Network Systems, 2017.

[10] Harish Doddi, Deepjyoti Deka, Saurav Talukdar, and Murti Salapaka. Efficient and passive learning of networked dynamical systems driven by non-white exogenous inputs. In International Conference on Artificial Intelligence and Statistics, pages 9982–9997. PMLR, 2022.

[11] Harish Doddi, Saurav Talukdar, Deepjyoti Deka, and Murti Salapaka. Exact topology learning in a network of cyclostationary processes. In 2019 American Control Conference (ACC), pages 4968–4973. IEEE, 2019.

[12] C. J. Finelli and J. M. Jenkins. A cyclostationary least mean squares algorithm for discrimination of ventricular tachycardia from sinus rhythm. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society Volume 13: 1991, pages 740–741, Oct 1991.

[13] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. Biostatistics, 9(3):432–441, 2008.

[14] Michael Grant and Stephen Boyd. Cvx: Matlab software for disciplined convex programming, version 2.1, 2014.

[15] Akira Hirose. Complex-valued neural networks: Advances and applications, volume 18. John Wiley & Sons, 2013.

[16] Roger A Horn and Charles R Johnson. Matrix analysis. Cambridge university press, 2012.

[17] Kwang-Y. Kim and Gerald R. North. Eofs of harmonizable cyclostationary processes. Journal of the Atmospheric Sciences, 54(19):2416–2427, 1997.

[18] Kwang-Y. Kim, Gerald R. North, and Jianping Huang. Eofs of one-dimensional cyclostationary time series: Computations, examples, and stochastic modeling. Journal of the Atmospheric Sciences, 53(7):1007–1017, 1996.

11
[19] Donatello Materassi and Giacomo Innocenti. Topological identification in networks of dynamical systems. *IEEE Transactions on Automatic Control*, 55(8):1860–1871, 2010.

[20] Donatello Materassi and Murti V Salapaka. On the problem of reconstructing an unknown topology via locality properties of the wiener filter. *IEEE transactions on automatic control*, 57(7):1765–1777, 2012.

[21] Donatello Materassi and Murti V Salapaka. Signal selection for estimation and identification in networks of dynamic systems: a graphical model approach. *IEEE Transactions on Automatic Control*, 65(10):4138–4153, 2019.

[22] Marzieh Nabi-Abdolyousefi. Network identification via node knockout. In *Controllability, Identification, and Randomness in Distributed Systems*, pages 17–29. Springer, 2014.

[23] Antonio Napolitiano and Chad M Spooner. Cyclic spectral analysis of continuous-phase modulated signals. *IEEE Transactions on Signal Processing*, 49(1):30–44, 2001.

[24] Alan V Oppenheim. *Discrete-time signal processing*. Pearson Education India, 1999.

[25] Judea Pearl. *Causality*. Cambridge university press, 2009.

[26] Christopher J Quinn, Negar Kiyavash, and Todd P Coleman. Directed information graphs. *IEEE Transactions on Information Theory*, 61(12):6887–6909, 2015.

[27] John Scott. Social network analysis. *Sociology*, 22(1):109–127, 1988.

[28] Firoozeh Sepehr and Donatello Materassi. Inferring the structure of polytree networks of dynamic systems with hidden nodes. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 4618–4623. IEEE, 2016.

[29] Firoozeh Sepehr and Donatello Materassi. Blind learning of tree network topologies in the presence of hidden nodes. *IEEE Transactions on Automatic Control*, 65(3):1014–1028, 2019.

[30] Saurav Talukdar, Deepjyoti Deka, Michael Chertkov, and Murti Salapaka. Topology learning of radial dynamical systems with latent nodes. In *2018 Annual American Control Conference (ACC)*, pages 1096–1101. IEEE, 2018.

[31] Saurav Talukdar, Deepjyoti Deka, Harish Doddi, Donatello Materassi, Michael Chertkov, and Murti V Salapaka. Physics informed topology learning in networks of linear dynamical systems. *Automatica*, 112:108705, 2020.

[32] Saurav Talukdar, Deepjyoti Deka, Donatello Materassi, and Murti Salapaka. Exact topology reconstruction of radial dynamical systems with applications to distribution system of the power grid. In *2017 American Control Conference (ACC)*, pages 813–818. IEEE, 2017.

[33] Saurav Talukdar, Mangal Prakash, Donatello Materassi, and Murti V Salapaka. Reconstruction of networks of cyclostationary processes. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 783–788. IEEE, 2015.

[34] R Wälchli, T Brunschwiler, B Michel, and D Poulikakos. Combined local microchannel-scale cfd modeling and global chip scale network modeling for electronics cooling design. *International Journal of Heat and Mass Transfer*, 53(5-6):1004–1014, 2010.

[35] Bin Yang and Wenzheng Bao. Complex-valued ordinary differential equation modeling for time series identification. *IEEE Access*, 7:41033–41042, 2019.

[36] ZK Zhu, ZH Feng, and FR Kong. Cyclostationarity analysis for gearbox condition monitoring: approaches and effectiveness. *Mechanical Systems and Signal Processing*, 19(3):467–482, 2005.
A Lemma for 2:

Lemma 15 Consider a collection of cyclostationary processes \( \{x_i(k)\}_{i=1}^{m} \) described by (1), which are jointly WSCS of period \( T \). Consider a lifted process \( X_i(k) = [x_i(kT), \ldots, x_i(kT + T - 1)]' \). The dynamics of \( \{X_i(k)\}_{i=1}^{m} \) is governed by:

\[
X_j(k) = \sum_{i=1, i \neq j}^{m} (H_{ji} * X_i)(k) + E_j(k). \tag{A.1}
\]

The \( z \)-transform is given by,

\[
X_j(z) = \sum_{i=1, i \neq j}^{m} H_{ji}(z)X_i(z) + E_j(z); \text{ where,} \tag{A.2}
\]

\[
E_i(k) = \begin{bmatrix} e_i(kT) & e_i(kT + 1) & \cdots & e_i(kT + T - 1) \end{bmatrix}',
\]

\[
X_i(k) = \begin{bmatrix} x_i(kT) & x_i(kT + 1) & \cdots & x_i(kT + T - 1) \end{bmatrix}',
\]

\[
H_{ji}(z) = D(z\tau)h_{ji}(z\tau), \quad D(z) = \begin{bmatrix} z^0 & z^1 & \cdots & z^{T-1} \\
\vdots & \ddots & \ddots & \vdots \\
 z^{-(T-1)} & \cdots & z^0 \end{bmatrix}. \]

PROOF.

The dynamics of a network of cyclostationary processes are given by,

\[
x_i(z) = \sum_{j=1}^{m} h_{ij}(z)x_j(z) + e_i(z),
\]

\[
x_i(k) = \sum_{j=1}^{m} \sum_{n=-\infty}^{\infty} h_{ij}(n)x_j(k-n) + e_i(k),
\]

\[
x_i(kT) = \sum_{j=1}^{m} \sum_{n=-\infty}^{\infty} h_{ij}(n)x_j(kT-n) + e_i(kT),
\]

\[
x_i(kT + p) = \sum_{j=1}^{m} \sum_{n=-\infty}^{\infty} h_{ij}(n)x_j(kT + p - n) + e_i(kT + p),
\]

where \( k \) is the time index and \( p \in \{0, 1, \ldots, T - 1\} \). Substitute \( n = aT + b \), where \( a \in \mathbb{Z} \) and \( b \) takes the values \( \{p, p-1, p-2, \ldots, p-T+1\} \).
Lift the scalar process $x_i(kT + p) = \sum_{j=1}^{m} \sum_{n=-\infty}^{\infty} h_{ij}(n)x_j(kT + p - n) + e_i(kT + p)$.

Define $H_{ij,p}[a] = h_{ij}(aT + p - t)$. Here, $p$ is the row index ranging from 0 to $T-1$ and $t$ is the column index ranging from 0 to $T-1$ of the filter matrix $H_{ij}$. For the diagonal entries ($p = t$) of $H_{ij}$, $H_{ij,pp}[n] = h_{ij}(nT)$ for $p \in \{0, \ldots, T-1\}$.

$x_i(kT + p) = \sum_{j=1}^{m} \sum_{t=0}^{T-1} \sum_{a=-\infty}^{\infty} H_{ij,pt}[a]x_j([k-a]T + t) + e_i(kT + p)$,

$x_i(kT + p) = \sum_{j=1}^{m} [H_{ij,00} x_j(kT) + \cdots + H_{ij,T-1} x_j(kT + T - 1)] + e_i(kT + p)$.

Lift the scalar process $x_i(k)$ to a vector process $X_i(k)$ by varying $p$ from 0 to $T-1$:

$$X_i(k) = \begin{bmatrix} x_i(kT) \\ x_i(kT + 1) \\ \vdots \\ x_i(kT + T - 1) \end{bmatrix}$$

Iterate the $p$ from 0 to $T-1$ to get the following relation:

$$\begin{bmatrix} x_i(kT) \\ x_i(kT + 1) \\ \vdots \\ x_i(kT + T - 1) \end{bmatrix} = \sum_{j=1}^{m} \begin{bmatrix} H_{ij,00} & H_{ij,01} & \cdots \\ \vdots & \vdots & \ddots \\ H_{ij,T-1,0} & H_{ij,T-1,1} & \cdots \end{bmatrix} \begin{bmatrix} x_j(kT) \\ x_j(kT + 1) \\ \vdots \\ x_j(kT + T - 1) \end{bmatrix} + \begin{bmatrix} e_i(kT) \\ e_i(kT + 1) \\ \vdots \\ e_i(kT + T - 1) \end{bmatrix}.$$
Example 16 Consider the following generative model for the time-series,

\[
X_i(k) = \sum_{j=1}^{m} H_{ij} \ast X_j(k) + E_i(k) \tag{A.3}
\]

Taking the Z transform, we get,

\[
X_i(z) = \sum_{i=1}^{m} H_{ij}(z)X_j(z) + E_i(z) \tag{A.4}
\]

where \( H_{ij,pt} \ast x_j(kT+t) = \sum_{n=-\infty}^{\infty} H_{ij,pt}[a]x_j(kT+t-aT) \) for \( t \in \{0, 1, \ldots, T-1\} \) with \( H_{ij,pt}[n] = h_{ij}(nT+p-t) = h_{ij}(T[n + \frac{p-t}{T}]) \). This implies \( H_{ij}(z) = Z[H_{ij}(k)] = D(z^\frac{1}{T})h_{ij}(z^\frac{1}{T}) \), where \( D(z) = \begin{bmatrix} z^0 & z^1 & \cdots & z^{T-1} \\
\vdots & \ddots & \vdots \\
z^{-(T-1)} & \cdots & z^0 \end{bmatrix} \).

B Illustration

B.1

Example 16 Consider the following generative model for the time-series,

\[
\sum_{n=0}^{l} a_{n,i} \frac{d^n x_i}{dt^n} = \sum_{j=1}^{m} b_{ij}x_j(t) + p_i(t), \tag{B.1}
\]

where, \( a_{n,i}, b_{ij} \) are complex-valued system constants, \( p_i(t) \) is an exogenous input, \( x_i(t) \) is the state of \( i \)-th agent.

Representing (B.1) in the form of (1) using bilinear transform (Tustin’s method [24]), we get \( h_{ij}(z) = \frac{b_{ij}}{S_i(z)} \), \( e_i(z) := \frac{b_i(z)}{S_i(z)} \), where \( \Delta t \) is the sampling period and \( S_i(z) := \sum_{n=1}^{l} a_{n,i}(\frac{2(1-z^{-1})}{\Delta t(1+z^{-1})})^n \). Assumption 3 holds for this example with \( a_{jki} = \frac{b_{jki}}{\Delta t} \) for all \( i, j, k \) such that \( b_{jki}, b_{ki} \neq 0 \) holds in (B.1).

C Pathological cases:

Theorem 17 Consider a well-posed and topologically detectable networked LDS described by (1), with its equivalent representation \((\mathbb{H}(z), E)\) as in (3), with associated graph \( G \), topology \( \mathcal{T} \) and satisfying Assumptions 3 and 1. Suppose \( i \) and \( j \) are such that \( i \in N_G(j) \), and denote the eigenvalues of \( B_i' \Phi X^{-1} \omega \mathcal{B}_i \), as \( E_{ij} \in \mathbb{C}^{T \times 1} \). The set of system parameters \( \{a_{jki}, \forall k \in V \setminus i, j\} \), such that \( \mathcal{E}_{ij}(l) \) is a constant for \( l \in \{1, 2, \ldots, T\} \) and all \( \omega \in [0, 2\pi] \), has measure zero.

PROOF. Using (4) and proof of Theorem 2, we have

\[
B_i' \Phi X^{-1} \omega \mathcal{B}_i = -b_{ij} S_i(\frac{\omega}{T}) \Phi E_j^{-1} \omega D(\frac{\omega}{T}) - b_j S_i(\frac{\omega}{T}) D^*(\frac{\omega}{T}) \Phi E_j^{-1} \omega + (b_{jki} b_{ki}) [D^*(\frac{\omega}{T}) \Phi E_k^{-1} \omega D(\frac{\omega}{T})].
\]

Suppose \( A, B \) and \( C \) correspond to first, second and third term in the above equation respectively. Given that the phase response of the eigenvalues of \( B_i' \Phi X^{-1} \omega \mathcal{B}_i \) is a constant and is denoted by \( c_{ij} \). The eigenvalues of \( C \) has a constant phase of \( \mathcal{P}_{b_{jki} b_{ki}} \). Thus, the phase of the eigenvalues of \( A + B \) is another constant \( \alpha = \mathcal{P}_{b_{jki} b_{ki}} \). Thus, \( e^{-\alpha} [A + B] \) has to be Hermitian and has real eigenvalues. It follows from the property of Hermitian that,
\[
e^{-j\omega T} [-b_j S_j^* (\frac{\omega}{T}) \Phi^{-1}_j (\omega) D(\frac{\omega}{T}) - b_j S_j (\frac{\omega}{T}) D^* (\frac{\omega}{T}) \Phi^{-1}_j (\omega)] \\
e^{-j\omega T} [-b_i S_i (\frac{\omega}{T}) D^* (\frac{\omega}{T}) \Phi^{-1}_i (\omega) - b_i S_i^* (\frac{\omega}{T}) \Phi^{-1}_i (\omega) D(\frac{\omega}{T})].
\]

The set of system parameters which satisfies the above equality constraint for all \( \omega \in [0, 2\pi] \) has a zero measure.

## D Algorithm 1

**Algorithm 1** Learning Algorithm for reconstructing the topology of LDG with cyclostationary inputs

**Input:** Nodal time-series \( x_i(k) \) for each node \( i \in \{1, 2, ..., m\} \) which is WSCS. Thresholds \( \rho, \tau \). Frequency points \( \Omega \).

**Output:** Reconstructed Topology \((V, \bar{E}_T)\)

1. Compute the periodogram of \( x_i(k) \) to determine the period \( T_i \). Determine \( T = LCM\{T_1, \cdots, T_m\} \). Lift each \( x_i(k) \) to vector process \( X_i(k) \) of size \( T \). Define \( X(k) = [X_1(k), \cdots, X_m(k)]' \)
2. Edge set \( \bar{E}_M \leftarrow \{\} \)
3. for all \( l, p \in \{1, 2, ..., m\}, l \neq p \) do
4. \hspace{1cm} if \( H_\infty [W_{lp}] \neq 0 \) then
5. \hspace{2cm} \( \bar{E}_M \leftarrow \bar{E}_M \cup \{(l,p)\} \)
6. \hspace{1cm} end if
7. end for
8. Edge set \( \bar{E}_T \leftarrow \bar{E}_M \)
9. for all \( (p, l) \in \bar{E}_M \) do
10. \hspace{1cm} Compute \( \{E_{pl}(t)\}_{t=1}^T = eig\{W_{pl}(\omega)\} \)
11. \hspace{2cm} if \( E_{pl}(t) \) is constant \( \forall \omega \in [0, 2\pi], \forall t \) then
12. \hspace{3cm} \( \bar{E}_T \leftarrow \bar{E}_T \setminus \{(p, l)\} \)
13. \hspace{2cm} end if
14. end for
E  Algorithm 2

Algorithm 2 Learning $\mathcal{G}_{T_0}(V_o, \mathcal{E}_{T_0})$

Input: $\mathcal{G}_e = (V_o, \mathcal{E}_e)$ generated from structure of $\Phi_{oo}$
Output: $T = (V_o, \mathcal{E}_T)$, $V_i$ and $V_{nl}$.

1: Edge set $\mathcal{E}_T \leftarrow \{}$
2: $V_{nl} \leftarrow \{}$
3: for all edge $a - b$ in $\mathcal{E}_e$ do
4:  if $Z := \{a, b\}$ there exist $I \neq \{\phi\}$ and $J \neq \{\phi\}$ such that $\text{sep}(I, Z, J)$ holds in $G_c$ then
5:    $V_{nl} \leftarrow V_{nl} \cup \{a, b\}$, $\mathcal{E}_T \leftarrow \mathcal{E}_T \cup \{(a, b)\}$
6:  end if
7: end for
8: $V_t \leftarrow V_o - V_{nl}$
9: for all $a \in V_t, b \in V_{nl}$ with $(a, b) \in \mathcal{E}_e$ do
10:  Compute $E_{ab} = \text{eig}[\Phi_{oo}^{-1}(a, b)]$
11:  if $E_{ab}[t]$ is not a constant $\forall \omega \in [0, 2\pi)$, for $t \in \{1, \cdots, T\}$ then
12:    $\mathcal{E}_T \leftarrow \mathcal{E}_T \cup \{(a, b)\}$
13: end if
14: end for

F  Algorithm 3

Algorithm 3 Reconstructing $V_h$ and $\mathcal{E}_{T_h}$

Input: $\mathcal{T} = (V_o, \mathcal{E}_{\mathcal{T}}) = \bigcup_{j=1}^{h} \mathcal{T}_j$
Output: $\tilde{T} = (V_{\tilde{T}}, \mathcal{E}_{\tilde{T}})$.

1: Node set $V_{\tilde{T}} \leftarrow V_o$, edge set $\mathcal{E}_{\tilde{T}} \leftarrow \mathcal{E}_{\mathcal{T}}$
2: $h \leftarrow \text{Number of disjoint subgraphs in } \mathcal{G}_{T_0}$
3: for all $j \in \{1, 2, ..., h\}$ do
4:  for all $i \in \{j + 1, ..., h\}$ do
5:    if there exist a pair of nodes $a, b$ such that $a \in \mathcal{T}_j$ and $b \in \mathcal{T}_i$ such that all their neighbors in $\mathcal{T}$ are connected in $\mathcal{G}_c$ then
6:      $V_{\tilde{T}} \leftarrow V_{\tilde{T}} \cup l_i$
7:      $\mathcal{E}_{\tilde{T}} \leftarrow \mathcal{E}_{\tilde{T}} \cup \{(a, l_j), (l_j, b)\}$
8:    end if
9:  end for
10: end for
11: Merge hidden nodes that are neighbors of the same observed node.