IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 August 2026, accepted 26 October 2026, date of publication 2 November 2026, date of current version 6 November 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3749496

RESEARCH ARTICLE

Conditional Independence Graph Learning From Dependent Data via Time-Delay Embedding

JITENDRA K. TUGNAIT, **(Life Fellow, IEEE)**
Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849, USA
e-mail: tugnajk@auburn.edu

This work was supported by the National Science Foundation under Grant CCF-2308473.

ABSTRACT We consider the problem of inferring the conditional independence graph (CIG) of a high-dimensional stationary multivariate Gaussian time series which could be short-range dependent (SRD) or long-range dependent (LRD). In a time series graph, each component of the vector series is represented by a distinct node, and associations between components are represented by edges between the corresponding nodes. We consider a time-domain approach based on time-delay embedding, casting the problem as one of multi-attribute graph estimation for random vectors where a vector is associated with each node of the graph. For an embedding dimension d, at each node the associated $(d+1)$-dimensional random vector consists of a time series component and its d delayed copies, with integer delays up to $d \geq 1$. Graph estimation from dependent data has been investigated in the past using frequency-domain methods whose analysis is restricted to SRD data. We consider a penalized negative pseudo log-likelihood function with convex lasso or non-convex regularizations to estimate the CIG. Non-convex regularization functions are log-sum, smoothly clipped absolute deviation (SCAD) and minimax concave penalty (MCP). We exploit an existing alternating direction method of multipliers (ADMM) approach to optimize the objective function. We establish sufficient conditions in a high-dimensional setting for consistency, local convexity when using non-convex penalties, and graph recovery, for both SRD and LRD series. We illustrate our approach with both synthetic and real-data examples. For a p-dimensional time series, the computational complexity of the ADMM algorithm is $\mathcal{O}((d+1)^3 p^3)$. Based on the numerical examples, a heuristic is suggested and illustrated for selection of d which, in theory, should tend to infinity to infer the time series graph.

INDEX TERMS Sparse graph learning, time series graphs, undirected graphs, multi-attribute graphs, long-range dependence.

I. INTRODUCTION

Graphical models are used to display and explore conditional independence structure in the analysis of multivariate data [1], [2], [3], [4]. In a time series Gaussian graphical model (TSGGM), the conditional statistical dependency structure among p component series $\{x_i(t)\}$, $i \in [p]$, of stationary, zero-mean, $p-$dimensional Gaussian time series $\{\boldsymbol{x}(t),\ t \in \mathcal{Z}\}$, is represented using an undirected graph $\mathcal{G} = (V, \mathcal{E})$ with a set of p vertices (nodes) $V = \{1, 2, \cdots, p\} = [p]$, and a corresponding set of (undirected) edges $\mathcal{E} \subseteq [p] \times [p]$. There is no edge between nodes i and j of the conditional independence graph (CIG) $\mathcal{G}$, (i.e., $\{i,j\} \notin \mathcal{E}$), if and only if (iff) $x_i(t)$ and $x_j(t)$ are conditionally independent given the remaining p-2 scalar series $x_\ell(t)$, $\ell \in V\backslash\{i,j\}$. This is a generalization of the CIG for random vectors where $\{i,j\} \notin \mathcal{E}$ iff $[\boldsymbol{\Omega}]_{ij} = 0$ where $[\boldsymbol{\Omega}]_{ij}$ denotes the (i,j)th component of $\boldsymbol{\Omega} = (E\{\boldsymbol{x}(t)\boldsymbol{x}^\top(t)\})^{-1}$ [3], [4], [5].

Graphical models were originally developed for random vectors [6, p. 234] and are useful in a wide variety of applications [3], [6], [7]. Graphical modeling of stationary time series originated with [8], followed by [5]. Denote the power spectral density (PSD) matrix of $\{\boldsymbol{x}(t)\}$ by $\boldsymbol{S}_x(f)$, where $\boldsymbol{S}_x(f) = \sum_{\tau=-\infty}^{\infty} \boldsymbol{R}_x(\tau) e^{-\iota 2\pi f \tau}$, $\boldsymbol{R}_x(\tau) = E\{\boldsymbol{x}(t+\tau)\boldsymbol{x}^T(t)\}$ and $\iota = \sqrt{-1}$. In [5] it was shown that conditional independence of two time series components given all other

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

© 2026 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

components of the time series, is encoded by zeros in the inverse PSD, that is, $\{i,j\} \notin \mathcal{E}$ iff the (i,j)th element of $\boldsymbol{S}_x^{-1}(f)$, $[\boldsymbol{S}_x^{-1}(f)]_{ij} = 0$ for every f; this requires $\boldsymbol{S}_x^{-1}(f)$ to exist for all f. Hence one can use estimated inverse PSD of observed time series to infer the associated graph. In [5] the low-dimensional case is addressed. Nonparametric approaches for graphical modeling of real-valued time series in high-dimensional settings (p is large and/or sample size n is of the order of p) have been formulated in the form of penalized log-likelihood in frequency-domain in [9], [10], [11], and [12].

In this paper we consider a time-domain approach to graph structure estimation for stationary Gaussian multivariate time series, using a time-delay embedding approach. The name delay embedding originates from [13] in the context of chaos detection where such embeddings reveal the dynamic structure of the underlying system, and such embeddings have been employed in other contexts, e.g., [14] and [15]. We cast the problem as one of multi-attribute graph estimation for random vectors where a vector is associated with each node of the graph. At each node, the associated random vector consists of a time series component and its time-delayed copies. In particular, given the ith component $x_i(t)$ of series $\{\boldsymbol{x}(t)\}$, we associate vector series $\boldsymbol{z}_i(t) = [x_i(t) \;\; x_i(t-1) \;\; \cdots \;\; x_i(t-d)]^\top \in \mathbb{R}^{d+1}$ with node i of the graph. Let $\boldsymbol{y}(t) = [\boldsymbol{z}_1(t) \;\; \boldsymbol{z}_2(t) \;\; \cdots \;\; \boldsymbol{z}_p(t)]^\top \in \mathbb{R}^{p(d+1)}$. We show that for "large" d, the precision matrix $(E\{\boldsymbol{y}(t)\boldsymbol{y}^\top(t)\})^{-1}$ of the augmented sequence $\boldsymbol{y}(t)$ captures the CIG of time-dependent $\{\boldsymbol{x}(t)\}$ (in the sense of [5]).

The delay embedding approach also allows us to analyze time series with long-range dependence, not analyzed in the frequency-domain approaches of [9], [10], and [11]. A (zero-mean) univariate stationary time series $\{y(t)\}$ is said to have long-range dependence (LRD), or long-memory, if its autocorrelation function $r_y(\tau) = E\{y(t+\tau)y(t)\}$ satisfies $\sum_{\tau=-\infty}^{\infty} |r_y(\tau)| = \infty$, and short-range dependence (SRD), or short-memory, otherwise [16], [17], [18], [19], [20], [21], and [22]. A (zero-mean) multivariate stationary time series $\{\boldsymbol{x}(t)\}$ (p-dimensional) is said to have LRD if its autocorrelation function $\boldsymbol{R}_x(\tau) = E\{\boldsymbol{x}(t+\tau)\boldsymbol{x}^\top(t)\}$ has the structure $[\boldsymbol{R}_x(\tau)]_{jk} = [\tilde{\boldsymbol{R}}(\tau)]_{jk}\tau^{d_j+d_k-1}$ where $[\boldsymbol{A}]_{jk}$ denotes (j,k)th component of $\boldsymbol{A}$, $d_j \in (0, 0.5)$, $j = 1, 2, \cdots, p$, and $\tilde{\boldsymbol{R}}(\tau) \in \mathbb{R}^{p\times p}$ satisfies $\lim_{\tau\to\infty} \tilde{\boldsymbol{R}}(\tau) = \bar{\boldsymbol{R}}$ [21]. Otherwise, $\{\boldsymbol{x}(t)\}$ is said to have SRD. Note that in the multivariate case, there is no universal characterization of LRD properties [21], [22].

A. RELATED WORK

There appears to be no prior reported work on any nonparametric time-domain approach to TSGGM estimation in high-dimensional settings from dependent data. Past work on TSGGM estimation from dependent data has focused on frequency-domain approaches. Prior work on graphical modeling for single-attribute dependent time series in low-dimensional settings includes [5], [23], [24], [25], and [26] where some nonparametric frequency-domain test statistics are used to test whether $\{i,j\} \in \mathcal{E}$. These approaches do not scale to high dimensions where p is comparable to or larger than the sample size n.

Nonparametric approaches in high-dimensional settings have been formulated in the form of lasso-penalized log-likelihood in frequency-domain in [9] and [10]. A sparse-group non-convex log-sum penalty is investigated in [11] and [12] to regularize the problem considered in [10], motivated by [27]. The analysis of the approaches of [9], [10], [11], and [12] is restricted to SRD processes. As an alternative to nonparametric modeling of time series, parametric graphical models utilizing (Gaussian) vector autoregressive (AR) process models of $\boldsymbol{x}(t)$ have been proposed in [6], [28], [29], and [30] and autoregressive moving average (ARMA) process (and related) models may be found in [31], [32], and [33], but these approaches are suitable only for low-dimensional settings.

There has been extensive prior work on both single attribute and multi-attribute graph estimation for random vectors based on estimation of the underlying precision matrix, and their analysis is based on the assumption of i.i.d. data [3], [4], [7], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45]. Analysis of estimation of the precision matrix and the corresponding graph under dependent data has been investigated in [46] using tail bounds derived using Hanson-Wright inequality [47]. Ref. [46] considers both SRD and LRD stationary processes.

In this paper we consider estimation of TSGGM for both SRD and LRD data whereas [46] considers the more restrictive precision matrix based graph estimation. That is, given dependent data $\{\boldsymbol{x}(t)\}$, in [46], the graph estimation is based on $\boldsymbol{\Omega} = (E\{\boldsymbol{x}(t)\boldsymbol{x}^\top(t)\})^{-1}$, not the inverse PSD $\boldsymbol{S}_x^{-1}(f)$ or an equivalent statistic, but analysis of the precision matrix estimator $\hat{\boldsymbol{\Omega}}$ accounts for temporal dependence of data. In this paper we account for temporal dependence of data both in graph estimation and in analysis. For some tail bounds in this paper we follow the time series dependence model of [48] which is formulated in terms of an infinite-order causal moving average model, unlike that of [46] which is stated in terms of the data covariance function.

In several applications such as functional magnetic resonance imaging (fMRI) signal analysis or financial time series analysis, the underlying temporal data is not i.i.d. In [30, Sec. 5.2] AR models are fitted to a financial time series (international stock market data) to infer the underlying time series graphical model. Analysis of some resting state fMRI data in [46] shows significant time-dependence. The data set analyzed in [46, Sec. 2.2] consists of a single subject 1190 temporal brain images, each image with 907 functional brain nodes. This data passed stationarity, Gaussianity and linearity tests [46, Sec. 2.2], implying that stationary Gaussian time series assumption used in this paper would be appropriate. Furthermore, analysis reported in [46, Sec. 2.2] determined that some fMRI series components are LRD, where, for large τ and some $C > 0$, $|E\{x_i(t+\tau)x_i(t)\}| = C|\tau|^{-0.3}$ (which corresponds to exponents

$\gamma = 0.65$ and $d_i = 0.35$ in Secs. III-B and III-B2, with $\gamma \in (0.5, 1]$ and $d_i \in (0, 0.5)$ implying long-range dependence). Therefore, consideration of long-range dependence in this paper is well-motivated.

Multi-attribute graph learning problem has been addressed in [36] and [42] with lasso regularization and in [43] with non-convex log-sum and SCAD regularizations, except that [36], [42], and [43] address estimation from i.i.d. data whereas in this paper we have dependent data. We use the log-likelihood of [36], [42], and [43] as our pseudo log-likelihood for dependent data. This allows us to optimize it as in [43] but analysis of its properties is different since now we have dependent data which could be SRD or LRD.

B. OUR CONTRIBUTIONS

In this paper we propose a nonparametric time-domain approach to TSGGM learning from a high-dimensional stationary multivariate Gaussian time series which could be short-range dependent (SRD) or long-range dependent (LRD). We consider a time-domain approach based on time-delay embedding, casting the problem as one of multi-attribute graph estimation for random vectors where a vector is associated with each node of the graph. We consider a penalized negative pseudo log-likelihood function with convex lasso or non-convex regularizations to estimate the CIG. Non-convex regularization functions are log-sum [27], smoothly clipped absolute deviation (SCAD) [34], [49], and minimax concave penalty (MCP) [35]. It is well-known that use of non-convex penalties can yield more accurate results compared to the convex lasso penalty. We exploit an existing alternating direction method of multipliers (ADMM) approach from [43] to optimize the objective function. We establish sufficient conditions in a high-dimensional setting for consistency, local convexity when using non-convex penalties, and graph recovery, for both SRD and LRD series. We illustrate our approach with both synthetic and real-data examples where comparisons with some existing approaches are provided.

Preliminary versions of this paper appear in the conference papers [50], [51]. The main idea of multi-attribute formulation appears in [50] with lasso penalty, and a consistency result is stated therein for SRD processes with no proof. In [51] a proof of consistency is given for LRD processes for lasso-penalized loss function and there are no simulation or real data results. In [50] and [51] non-convex penalties are not addressed. Furthermore, in [50] and [51] the sufficient conditions for consistency for lasso penalty are stated in terms of the covariance of the multi-attribute process $\{\boldsymbol{y}(t)\}$ whereas in this paper we also formulate them in terms of the original process $\{\boldsymbol{x}(t)\}$; see Remarks 4 and 5 in Sec. V. There are no real data results in [50] and [51], and the synthetic data results in this paper are more comprehensive compared to [50].

C. NOTATION

For a set V, $|V|$ denotes its cardinality. Given a real a, $(a)_+ = \max(0, a)$. Given $\boldsymbol{A} \in \mathbb{C}^{p\times p}$, we use $\phi_{\min}(\boldsymbol{A})$, $\phi_{\max}(\boldsymbol{A})$, $|\boldsymbol{A}|$ and $\mathrm{tr}(\boldsymbol{A})$ to denote the minimum eigenvalue, maximum eigenvalue, determinant and trace of $\boldsymbol{A}$, respectively, and $\boldsymbol{A} \succ \boldsymbol{0}$ and $\boldsymbol{A} \succeq \boldsymbol{0}$ denote that $\boldsymbol{A}$ is positive-definite and positive semi-definite, respectively. Given $\boldsymbol{B} \in \mathbb{C}^{p\times m}$, $[\boldsymbol{B}]_{ij}$ denotes the (i,j)-th element of $\boldsymbol{B}$, and so does B_{ij}, and $\boldsymbol{I}_q$ denotes the $q \times q$ identity matrix. The symbol $\otimes$ denotes the matrix Kronecker product. The superscripts $*$ and H denote the complex conjugate and conjugate transpose operations, respectively.

For $\boldsymbol{B} \in \mathbb{C}^{p\times q}$, we define the operator norm, the Frobenius norm, the vectorized ℓ_1 norm and element-wise maximum, respectively, as $\|\boldsymbol{B}\| = \sqrt{\phi_{\max}(\boldsymbol{B}^H\boldsymbol{B})}$, $\|\boldsymbol{B}\|_F = \sqrt{\mathrm{tr}(\boldsymbol{B}^H\boldsymbol{B})}$, $\|\boldsymbol{B}\|_1 = \sum_{i,j}|B_{ij}|$ and $\|\boldsymbol{B}\|_\infty = \max_{i,j}|B_{ij}|$. Given $\boldsymbol{A} \in \mathbb{C}^{n\times p}$, column vector $\mathrm{vec}(\boldsymbol{A}) \in \mathbb{C}^{np}$ denotes the vectorization of $\boldsymbol{A}$ which stacks the columns of the matrix $\boldsymbol{A}$.

Given a variable vector $\boldsymbol{x}$ or matrix $\boldsymbol{X}$, we use $\boldsymbol{x}^\diamond$ or $\boldsymbol{X}^\diamond$, respectively, to denote their true values. Given a diagonal matrix $\boldsymbol{D} = \mathrm{diag}\{d_1, d_2, \cdots, d_p\} \in \mathbb{C}^p$ and a scalar a, a^D denotes $\mathrm{diag}\{a^{d_1}, a^{d_2}, \cdots, a^{d_p}\}$, similar to [21]. For $\boldsymbol{y}_n \in \mathbb{R}^p$ and $\boldsymbol{x}_n \in \mathbb{R}^q$, $\boldsymbol{x}_n = \mathcal{O}(\boldsymbol{y}_n)$ means there exists $0 < M < \infty$ such that $\|\boldsymbol{x}_n\| \le M\|\boldsymbol{y}_n\| \forall n \ge 1$, and $\boldsymbol{x}_n = o(\boldsymbol{y}_n)$ means $\lim_{n\to\infty} \|\boldsymbol{x}_n\|/\|\boldsymbol{y}_n\| = 0$. The notation $\boldsymbol{y}_n = \mathcal{O}_P(\boldsymbol{x}_n)$ for random vectors $\boldsymbol{y}_n \in \mathbb{R}^p$ and $\boldsymbol{x}_n \in \mathbb{R}^q$ means that for any $\varepsilon > 0$, there exists $0 < M < \infty$ such that $P(\|\boldsymbol{y}_n\| \le M\|\boldsymbol{x}_n\|) \ge 1 - \varepsilon \forall n \ge 1$.

D. OUTLINE

The rest of the paper is organized as follows. Some background material for graphical models for random vectors and for multivariate time series is presented in Sec. II. Time-delay embedding and the resulting multi-attribute graph formulation is discussed in Sec. III. The resulting log-likelihood formulation of the problem is presented in Sec. IV and its optimization via ADMM is presented in Sec. IV-B. A computational complexity analysis of the ADMM algorithm is in Sec. IV-C, and model selection aspect for choosing the regularization parameter in the penalized pseudo log-likelihood objective function is discussed in Sec. IV-D. In Sec. V we provide a theoretical analysis of the proposed approach, resulting in Theorems 1-3. Synthetic data numerical results are presented in Sec. VI and real data results are in Sec. VII. The proofs of the technical results in Sec. V are given in two appendices.

II. GRAPHICAL MODELS

Here we provide some background material for graphical models for random vectors and for multivariate time series which then forms the basis for the proposed time-delay embedding discussed in Sec. III.

A. RANDOM VECTORS

Consider a graph $\mathcal{G} = (V, \mathcal{E})$ with $V = [p]$ and $\mathcal{E} \subseteq V \times V$. Let $\boldsymbol{x} = [x_1\ x_2\ \cdots\ x_p]^\top \in \mathbb{R}^p$ denote a Gaussian random vector that is zero-mean with covariance $\boldsymbol{\Sigma} = E\{\boldsymbol{x}\boldsymbol{x}^\top\} \succ \mathbf{0}$. The conditional independence relationships among x_i's are encoded in $\mathcal{E}$ where edge $\{i,j\} \notin \mathcal{E}$ iff x_i and x_j are conditionally independent given $x_\ell, \ell \in V\backslash\{i,j\}$. Let

$$\boldsymbol{x}_{-ij} = \{x_k\ :\ k \in V\backslash\{i,j\}\} \in \mathbb{R}^{p-2} \tag{1}$$

denote the vector $\boldsymbol{x}$ after deleting x_i and x_j from it. Let $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$ denote the precision matrix. Define

$$e_{i|-ij} = x_i - E\{x_i|\boldsymbol{x}_{-ij}\}\,,\quad e_{j|-ij} = x_j - E\{x_j|\boldsymbol{x}_{-ij}\}\,. \tag{2}$$

Then we have the following equivalence [2]

$$\{i,j\} \notin \mathcal{E} \ \Leftrightarrow\ \Omega_{ij} = 0 \ \Leftrightarrow\ E\{e_{i|-ij}e_{j|-ij}\} = 0\,. \tag{3}$$

B. MULTIVARIATE TIME SERIES GRAPHICAL MODELS

Consider zero-mean stationary Gaussian time series $\boldsymbol{x}(t) \in \mathbb{R}^p$, $t \in \mathbb{Z}$, with PSD $\boldsymbol{S}_x(f)$. The conditional independence relationships among time series components $\{x_i(t)\}$'s are encoded in edge set $\mathcal{E}$ of $\mathcal{G} = (V, \mathcal{E})$ where edge $\{i,j\} \notin \mathcal{E}$ iff $\{x_i(t),\ t \in \mathbb{Z}\}$ and $\{x_j(t),\ t \in \mathbb{Z}\}$ are conditionally independent given the remaining p-2 components

$$\boldsymbol{x}_{-ij,\mathbb{Z}} = \{x_k(t)\ :\ k \in V\backslash\{i,j\},\ t \in \mathbb{Z}\}\,. \tag{4}$$

Define

$$e_{i|-ij}(t) = x_i(t) - E\{x_i(t)|\boldsymbol{x}_{-ij,\mathbb{Z}}\} \tag{5}$$
$$e_{j|-ij}(t) = x_j(t) - E\{x_j(t)|\boldsymbol{x}_{-ij,\mathbb{Z}}\}\,. \tag{6}$$

Then we have the following equivalence [5]

$$\begin{aligned}\{i,j\} \notin \mathcal{E} \ &\Leftrightarrow\ [\boldsymbol{S}_x^{-1}(f)]_{ij} = 0\ \forall f \in [0,1]\\ &\Leftrightarrow\ E\{e_{i|-ij}(t+\tau)e_{j|-ij}(t)\} = 0\ \forall \tau \in \mathbb{Z}\,.\end{aligned} \tag{7}$$

C. MULTI-ATTRIBUTE GRAPHICAL MODELS FOR RANDOM VECTORS

Now consider p jointly Gaussian vectors $\boldsymbol{z}_i \in \mathbb{R}^m$, $i \in [p]$. We associate $\boldsymbol{z}_i$ with the ith node of graph $\mathcal{G} = (V, \mathcal{E})$, $V = [p]$, $\mathcal{E} \subseteq V \times V$. We now have m attributes per node. Now $\{i,j\} \notin \mathcal{E}$ iff vectors $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ are conditionally independent given the remaining p-2 vectors $\{\boldsymbol{z}_\ell\,, \ell \in V\backslash\{i,j\}\}$.

$$\boldsymbol{x} = [\boldsymbol{z}_1^\top\ \boldsymbol{z}_2^\top\ \cdots\ \boldsymbol{z}_p^\top]^\top \in \mathbb{R}^{mp}\,. \tag{8}$$

Let $\boldsymbol{\Omega} = (E\{\boldsymbol{x}\boldsymbol{x}^\top\})^{-1}$ assuming $E\{\boldsymbol{x}\boldsymbol{x}^\top\} \succ \mathbf{0}$. Define the $m \times m$ subblock $\boldsymbol{\Omega}^{(ij)}$ of $\boldsymbol{\Omega}$ as

$$[\boldsymbol{\Omega}^{(ij)}]_{rs} = [\boldsymbol{\Omega}]_{(i-1)m+r,(j-1)m+s}\,,\ r,s \in [m]\,. \tag{9}$$

Let

$$\boldsymbol{z}_{-ij} = \{\boldsymbol{z}_k\ :\ k \in V\backslash\{i,j\}\} \in \mathbb{R}^{m(p-2)} \tag{10}$$

denote the vector $\boldsymbol{x}$ in (8) after deleting vectors $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ from it. Define

$$\boldsymbol{e}_{i|-ij} = \boldsymbol{z}_i - E\{\boldsymbol{z}_i|\boldsymbol{z}_{-ij}\}\,,\quad \boldsymbol{e}_{j|-ij} = \boldsymbol{z}_j - E\{\boldsymbol{z}_j|\boldsymbol{z}_{-ij}\}\,. \tag{11}$$

Then we have the following equivalence [36]

$$\{i,j\} \notin \mathcal{E} \ \Leftrightarrow\ \boldsymbol{\Omega}^{(ij)} = \mathbf{0} \ \Leftrightarrow\ E\{\boldsymbol{e}_{i|-ij}\boldsymbol{e}_{j|-ij}^\top\} = \mathbf{0}\,, \tag{12}$$

where the first equivalence in (12) is given in [36, Sec. 2.1] and the second equivalence is given in [36, Appendix B.3].

III. TIME-DELAY EMBEDDING

Consider stationary, zero-mean, $p-$dimensional multivariate Gaussian time series $\boldsymbol{x}(t)$ with PSD $\boldsymbol{S}_x(f) \succ \mathbf{0}$. We wish to estimate the associated CIG $\mathcal{G} = (V, \mathcal{E})$ as discussed in Sec. II-B, but using a time-domain formulation which is explained in this section. This approach results in a multi-attribute graph estimation problem discussed in [43], except that [43] deals with i.i.d. data whereas here we have dependent data. In Sec. III-B we quantify the dependence structure of $\{\boldsymbol{x}(t)\}$.

A. MULTI-ATTRIBUTE FORMULATION FOR TIME SERIES GRAPHICAL MODELING: TIME-DELAY EMBEDDING

With $\{\boldsymbol{x}(t)\}$ as in Sec. II-B, let $(d \geq 1)$

$$\boldsymbol{z}_i(t) = [x_i(t)\ x_i(t-1)\ \cdots\ x_i(t-d)]^\top \in \mathbb{R}^{d+1} \tag{13}$$
$$\boldsymbol{y}(t) = [\boldsymbol{z}_1^\top(t)\ \boldsymbol{z}_2^\top(t)\ \cdots\ \boldsymbol{z}_p^\top(t)]^\top \in \mathbb{R}^{(d+1)p}\,. \tag{14}$$

Let $\boldsymbol{\Omega}_y = (E\{\boldsymbol{y}(t)\boldsymbol{y}^\top(t)\})^{-1}$. With $m = d+1$, define the $m \times m$ subblock $\boldsymbol{\Omega}_y^{(ij)}$ of $\boldsymbol{\Omega}_y$ as

$$[\boldsymbol{\Omega}_y^{(ij)}]_{rs} = [\boldsymbol{\Omega}_y]_{(i-1)m+r,(j-1)m+s}\,,\ r,s \in [m]\,. \tag{15}$$

Lemma 1: For a stationary zero-mean time series $\{\boldsymbol{x}(t)\}$ as in Sec. II-B, $\{x_i(u),\ t-d \leq u \leq t\}$ and $\{x_j(u),\ t-d \leq u \leq t\}$, $i \neq j$, are conditionally independent given $\{x_\ell(u),\ t-d \leq u \leq t,\ \ell \in V\backslash\{i,j\}\}$ iff $\boldsymbol{\Omega}_y^{(ij)} = \mathbf{0}$. •

Proof: Let

$$\boldsymbol{z}_{-ij}(t) = \{\boldsymbol{z}_k(t)\ :\ k \in V\backslash\{i,j\}\}\,, \tag{16}$$
$$\boldsymbol{e}_{i|-ij}(t) = \boldsymbol{z}_i(t) - E\{\boldsymbol{z}_i(t)|\boldsymbol{z}_{-ij}(t)\}\,, \tag{17}$$
$$\boldsymbol{e}_{j|-ij}(t) = \boldsymbol{z}_j(t) - E\{\boldsymbol{z}_j(t)|\boldsymbol{z}_{-ij}(t)\}\,. \tag{18}$$

Then by multi-attribute graphical modeling (Sec. II-C),

$$\boldsymbol{\Omega}_y^{(ij)} = \mathbf{0} \ \Leftrightarrow\ E\{\boldsymbol{e}_{i|-ij}(t)\boldsymbol{e}_{j|-ij}^\top(t)\} = \mathbf{0}\,. \tag{19}$$

Define $m(p-2)$-vector

$$\tilde{\boldsymbol{x}}_{-ij;t,d} = \{x_k(s)\ :\ k \in V\backslash\{i,j\}\,,\ t-d \leq s \leq t\}\,, \tag{20}$$
$$e_{xi|-ij}(t') = x_i(t') - E\{x_i(t')|\tilde{\boldsymbol{x}}_{-ij;t,d}\}\,, \tag{21}$$
$$e_{xj|-ij}(t') = x_j(t') - E\{x_j(t')|\tilde{\boldsymbol{x}}_{-ij;t,d}\}\,. \tag{22}$$

Notice that $e_{xi|-ij}(t')$ above is an element of $\boldsymbol{e}_{i|-ij}(t)$ defined in (17) for any $t-d \leq t' \leq t$. Then by (12) and (19), we have

$$\begin{aligned}\boldsymbol{\Omega}_y^{(ij)} = \mathbf{0} \ &\Leftrightarrow\ E\{e_{xi|-ij}(t_1)e_{xj|-ij}(t_2)\} = 0,\\ &\text{for}\quad t-d \leq t_1, t_2 \leq t.\end{aligned} \tag{23}$$

This proves the desired result. ■

Remark 1: It follows from (23) that if we let $d \uparrow \infty$, then checking if $\boldsymbol{\Omega}_y^{(ij)} = \mathbf{0}$ to ascertain (19) becomes a surrogate for checking if the last equivalence in (7) holds true for TSGGM estimation without using frequency-domain

methods followed e.g., in [9], [10], and [11]. For "large" d, with $\mathbf{\Omega}_y^{(ij)} = \mathbf{0}$ as surrogate for $[\boldsymbol{S}_x^{-1}(f)]_{ij} = 0\ \forall f \in [0,1]$, in this paper we wish to learn the graph $\mathcal{G} = (V, \mathcal{E})$ where we take

$$\{i,j\} \notin \mathcal{E} \Leftrightarrow \mathbf{\Omega}_y^{(ij)} = \mathbf{0}$$

instead of $\{i,j\} \notin \mathcal{E} \Leftrightarrow [\boldsymbol{S}_x^{-1}(f)]_{ij} = 0\ \forall f \in [0,1]$ (as e.g., in [10]). Time-domain methods allow us to address inference of CIGs for both SRD and LRD sequences quantitatively as we show in this paper, unlike frequency-domain methods of [9], [10], and [11] whose analysis is restricted to SRD sequences. □

Remark 2: By an abuse of notation, let $\mathbf{\Omega}_{y,d}$ denote $\mathbf{\Omega}_y$ with the dependence on d made explicit. If $d_1 < d_2$, then $\mathbf{\Omega}_{y,d_1}^{(ij)}$ is a submatrix of $\mathbf{\Omega}_{y,d_2}^{(ij)}$. By [5, Remark 2.3], $[\boldsymbol{S}_x^{-1}(f)]_{ij} \equiv 0 \Leftrightarrow \lim_{d\to\infty} \|\mathbf{\Omega}_{y,d}^{(ij)}\|_F = 0$, implying that $\|\mathbf{\Omega}_{y,d}^{(ij)}\|_F = 0$ for any $d \geq 0$ and therefore $\{i,j\} \notin \mathcal{E}$. But if $\{i,j\} \in \mathcal{E}$, then $[\boldsymbol{S}_x^{-1}(f)]_{ij} \not\equiv 0$, but $\|\mathbf{\Omega}_{y,d}^{(ij)}\|_F$ could be zero. How large should d be so that the set of edges for which $\mathbf{\Omega}_{y,d}^{(ij)} = \mathbf{0}$ yields the set of edges for which $[\boldsymbol{S}_x^{-1}(f)]_{ij} = 0\ \forall f \in [0,1]$? This question is yet unresolved. We show in Sec. IV-C that the computational complexity of the ADMM optimization algorithm used to estimate $\mathbf{\Omega}_y$ from data is $\mathcal{O}((d+1)^3p^3)$. In Sec. V, under some technical conditions, we provide an upperbound on the Frobenius norm of the error in estimating $\mathbf{\Omega}_y$, and it is $\mathcal{O}((d+1)^2/\sqrt{n-d})$ for SRD and a class of LRD sequences (see Theorem 1 and Remark 6 in Sec. V). Note that the number of unknown parameters in $\mathbf{\Omega}_y$ increase as $\mathcal{O}((d+1)^2)$. As d increases, both computational complexity and the Frobenius error norm in estimating $\mathbf{\Omega}_y$ increase. But from a graph recovery viewpoint, the empirical results based on the synthetic data examples in Sec. VI-A2 and Table 2 show that as one increases d, the graph recovery performance metrics (the F_1 score and the Hamming distance between the true and estimated edge-sets) improve significantly at first, but then they plateau out with increasing d. This observation suggests the heuristic that the user should try several values of d and pick the smallest value, say d_0, beyond which (e.g., for $d = d_0 + 1$) the estimated graph is only marginally different from the estimated graph for $d = d_0$. In practice, since the true graph is unknown, one cannot compute the F_1 score or the Hamming distance. For real data, what constitutes marginal difference would be problem and user dependent. A financial time series based real data results are presented in Sec. VII where Fig. 2 shows the estimated graphs (p =97) for $d = 3$, 4 and 5, and where the strengths of the edge connections, based on normalized $\|\hat{\mathbf{\Omega}}_{y\lambda}^{(k\ell)}\|_F/(\max_{k,\ell:\, k\neq\ell} \|\hat{\mathbf{\Omega}}_{y\lambda}^{(k\ell)}\|_F)$ for the edge $\{k,\ell\}$, are color-coded ($\hat{\mathbf{\Omega}}_{y\lambda}$ denotes the estimate of $\mathbf{\Omega}_y$ obtained using a regularization parameter λ as discussed in Sec. IV-A). A visual examination shows very minor differences in the three graphs. The number of distinct edges are 347, 307 and 310 for $d = 3$, 4 and 5, respectively, with differences being in weaker edges (smaller $\|\hat{\mathbf{\Omega}}^{(k\ell)}\|_F$ values). □

B. TIME SERIES MODEL

To quantify the dependence structure of $\{\boldsymbol{x}(t)\}$, we will follow [48]; other possibilities include [46]. Assume

(A1) Assume that $\{\boldsymbol{x}(t)\}$ of Sec. II-B obeys

$$\boldsymbol{x}(t) = \sum_{i=0}^{\infty} \boldsymbol{A}_i \boldsymbol{e}(t-i)\,, \tag{24}$$

where $\{\boldsymbol{e}(t)\}$ is i.i.d., Gaussian, zero-mean with identity covariance, $\boldsymbol{e}(t) \in \mathbb{R}^p$, $\boldsymbol{A}_i \in \mathbb{R}^{p\times p}$, and

$$\max_{1\leq q\leq p} \sqrt{\sum_{k=1}^{p} ([\boldsymbol{A}_i]_{qk})^2} \leq \frac{c_a(p)}{(\max(1,i))^{\gamma}} \tag{25}$$

for all $i \geq 0$, some $c_a(p) \in (0,\infty)$, and $\gamma > \frac{1}{2}$.

By assumption (A1), it follows that with $\boldsymbol{y}(t)$ as in (14), $\boldsymbol{y}(t) = \sum_{i=0}^{\infty} \boldsymbol{B}_i \boldsymbol{e}(t-i)$, $\boldsymbol{e}(t) \in \mathbb{R}^p$ is as in assumption (A1), $m = d+1$, $\boldsymbol{B}_i \in \mathbb{R}^{(mp)\times p}$, for some $\boldsymbol{B}_i$'s such that

$$\max_{1\leq q\leq mp} \sqrt{\sum_{k=1}^{p} ([\boldsymbol{B}_i]_{qk})^2} \leq \frac{c_a(p)}{(\max(1,i))^{\gamma}} \tag{26}$$

for all $i \geq 0$, with $c_a(p)$ and γ as in assumption (A1).

By assumption (A1), $|[\boldsymbol{A}_i]_{qk}| = \mathcal{O}(i^{-\gamma})$, $i \geq 1$. In assumption (A1), the constant $c_a(p)$ may be a function of p. For instance, suppose that $|[\boldsymbol{A}_i]_{qk}| \leq \bar{a} i^{-\gamma} \forall\, i,q,k \geq 1$, for some $0 < \bar{a} < \infty$. Then $c_a(p) = \mathcal{O}(\sqrt{p})$. For the specific models considered in Secs. III-B1 and III-B2 and used in our numerical examples, the condition $|[\boldsymbol{A}_i]_{qk}| \leq \bar{a} i^{-\gamma}$ holds true. If $\boldsymbol{A}_i$ has a block-structure (as e.g., resulting from the model (100) considered in Sec. VI-A1) where the block-size is fixed independent of p, then $c_a(p)$ is a constant.

We have $\boldsymbol{R}_x(\tau) = \mathbb{E}\{\boldsymbol{x}(t+\tau)\boldsymbol{x}^T(t)\} = \sum_{i=0}^{\infty} \boldsymbol{A}_{i+\tau}\boldsymbol{A}_i^{\top}$. Then by [48, Lemma VI.1,Supplement], for every $i,j \in [p]$, $|[\boldsymbol{R}(\tau)]_{ij}| = \mathcal{O}(\tau^{-\gamma})$, $\mathcal{O}(\tau^{-\gamma}\ln(\tau))$ and $\mathcal{O}(\tau^{1-2\gamma})$ for $\gamma > 1$, $\gamma = 1$ and $\frac{1}{2} < \gamma < 1$, respectively, for $\tau \geq 1$. For $\gamma > 1$, $[\boldsymbol{R}(\tau)]_{ij}$ is (absolutely) summable $\forall\, i,j \in [p]$, hence, $\{\boldsymbol{x}(t)\}$ is SRD. For $\gamma \in (\frac{1}{2}, 1]$, $\sum_{\tau=-\infty}^{\infty} |[\boldsymbol{R}(\tau)]_{ij}|$ may not be finite for some $i,j \in [p]$, implying $\{\boldsymbol{x}(t)\}$ is LRD [21], [22], [48].

We now consider specific models that result in (24) satisfying assumption (A1) for different values of γ.

1) VECTOR ARMA MODELS

In the SRD case ($\gamma > 1$), assumption (A1) is satisfied if $\boldsymbol{x}(t)$ is generated by an asymptotically stable vector autoregressive moving average (ARMA) model satisfying

$$\boldsymbol{x}(t) = \sum_{i=1}^{q} \tilde{\boldsymbol{A}}_i \boldsymbol{x}(t-i) + \sum_{i=0}^{r} \tilde{\boldsymbol{B}}_i \boldsymbol{e}(t-i) \tag{27}$$

where $\{\boldsymbol{e}(t)\}$ is as in (24) and $\tilde{\boldsymbol{B}}_0$ is full rank. The model in (27) is called the ARMA(q,r) model. Let ($z \in \mathbb{C}$)

$$\mathcal{A}(z) := \boldsymbol{I}_p - \sum_{i=1}^{q} \tilde{\boldsymbol{A}}_i z^{-i}\,, \quad \mathcal{B}(z) := \sum_{i=0}^{r} \tilde{\boldsymbol{B}}_i z^{-i}\,. \tag{28}$$

We assume that $\det(\mathcal{B}(z)) \neq 0$ for any $|z| = 1$ and $\det(\mathcal{A}(z)) \neq 0$ for any $|z| \geq 1$. Consider the roots of $\det(\mathcal{A}(z)) = 0$. For stationarity of $\boldsymbol{x}(t)$ we need these roots $\bar{z}_\ell$'s ($\ell \in [pq]$) to satisfy $|\bar{z}_\ell| < 1$. Suppose that the maximum magnitude root is $\bar{z}_{\max}$ with multiplicity $\bar{m}$. Then by [52, p. 186], the impulse response sequence $\{\boldsymbol{A}_i\}_{i=0}^{\infty}$ of (27) (see also (24)) satisfies

$$\|\boldsymbol{A}_i\|_F \leq a\, i^{\bar{m}-1}\, |\bar{z}_{\max}|^i\,, \quad i \geq 1, \quad 1 \leq \bar{m} \leq pq\,, \tag{29}$$

for some positive $a < \infty$. Then there exists $0 < b < \infty$ and $1 < \gamma < \infty$ such that $ai^{\bar{m}-1}|\bar{z}_{\max}|^i \leq b\, i^{-\gamma}$ for $i \geq 1$, thereby satisfying assumption (A1). [Proof: Pick $\gamma > 1$ and then pick i_0 such that for any $i \geq i_0$, $i^{\bar{m}-1}|\bar{z}_{\max}|^i \leq i^{-\gamma}$. Such a solution for i_0 exists. The desired inequality is equivalent to $\gamma \leq -(\bar{m}-1) + \frac{i}{\ln(i)}(-\ln(|\bar{z}_{\max}|))$ for $i \geq i_0$. Now $-\ln(|\bar{z}_{\max}|) > 0$, $\frac{i}{\ln(i)}$ is increasing in i for $\ln(i) > 1$ and $\lim_{i\to\infty} \frac{i}{\ln(i)} = \infty$. Hence, for a fixed $\gamma > 1$, there exists an i_0 such that $i^{\bar{m}-1}|\bar{z}_{\max}|^i \leq i^{-\gamma}$ for any $i \geq i_0$. Now set $b = \max\left\{\max_{1\leq i\leq i_0}\left\{a|\bar{z}_{\max}|^i\, i^{\bar{m}-1+\gamma}\right\}, 1\right\}$, leading to $\|\boldsymbol{A}_i\|_F \leq b\, i^{-\gamma}$ for $i \geq 1$, proving the claim.]

The PSD of the ARMA(q, r) model (27) is given by

$$\boldsymbol{S}_x(f) = \mathcal{A}^{-1}(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})\mathcal{B}^H(e^{\iota 2\pi f})(\mathcal{A}^{-1}(e^{\iota 2\pi f}))^H\,. \tag{30}$$

Since $\det(\mathcal{B}(z)) \neq 0$ for any $|z| = 1$, $\boldsymbol{S}_x^{-1}(f)$ exists. By [53, Sec. 2.2], $\boldsymbol{S}_x(f)$ is bounded and continuous in f with $\boldsymbol{S}_x(f) \succ \boldsymbol{0}$ and $\boldsymbol{S}_x^{-1}(f) \succ \boldsymbol{0}$ for $f \in [-0,5, 0.5]$. By [53, (2.6)], we have

$$\min_f \phi_{\min}(\boldsymbol{S}_x(f)) \geq \frac{\min_f \phi_{\min}\left(\mathcal{B}^H(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})\right)}{\max_f \phi_{\max}\left(\mathcal{A}^H(e^{\iota 2\pi f})\mathcal{A}(e^{\iota 2\pi f})\right)} \tag{31}$$

$$\max_f \phi_{\max}(\boldsymbol{S}_x(f)) \leq \frac{\max_f \phi_{\max}\left(\mathcal{B}^H(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})\right)}{\min_f \phi_{\min}\left(\mathcal{A}^H(e^{\iota 2\pi f})\mathcal{A}(e^{\iota 2\pi f})\right)} \tag{32}$$

where the max and min operations are over $f \in [-0.5, 0.5]$ since $\boldsymbol{S}_x(f)$, $\mathcal{A}(e^{\iota 2\pi f})$ and $\mathcal{B}(e^{\iota 2\pi f})$ are periodic in f with period 1. Since $\det(\mathcal{B}(z)) \neq 0$ for any $|z| = 1$,

$$\det(\mathcal{B}^H(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})) = |\det(\mathcal{B}(e^{\iota 2\pi f})|^2 > 0 \tag{33}$$

and therefore, all eigenvalues of Hermitian $\mathcal{B}^H(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})$ are positive for $f \in [-0,5, 0.5]$. By continuity of the eigenvalues of $\mathcal{B}^H(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})$ in f and compactness of the interval $[-0, 5, 0.5]$, the maximum and the minimum of $\phi_{\max}\left(\mathcal{B}^H(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})\right)$ are achieved for some $\bar{f}$ and $\underline{f}$ in $[-0, 5, 0.5]$, respectively. Similar considerations apply to $\mathcal{A}^H(e^{\iota 2\pi f})\mathcal{A}(e^{\iota 2\pi f})$. Thus, $\boldsymbol{S}_x(f) \succ \boldsymbol{0}$ and $\boldsymbol{S}_x^{-1}(f) \succ \boldsymbol{0}$.

2) VECTOR ARFIMA MODELS

There are several variations of vector autoregressive fractionally integrated moving average (ARFIMA) models which are all equivalent in the scalar case [17], [18], [19], [22]. They are also known as FARIMA (fractional autoregressive integrated moving average) models [20], [22]. They generate both short range and long range dependencies through ARMA and fractional integration modeling, respectively [16], [19], [20], [21], [22]. We consider the following version from [19]. Let $\{\boldsymbol{u}(t)\}$ be an ARMA(q, r) sequence as in (27) (with a notation change), given by

$$\boldsymbol{u}(t) = \sum_{i=1}^{q} \tilde{\boldsymbol{A}}_i \boldsymbol{u}(t-i) + \sum_{i=0}^{r} \tilde{\boldsymbol{B}}_i \boldsymbol{e}(t-i) \tag{34}$$

where $\tilde{\boldsymbol{A}}_i, \tilde{\boldsymbol{B}}_i$ and $\{\boldsymbol{e}(t)\}$ are as in Sec. III-B1. We generate the ith component $x_i(t)$ of $\boldsymbol{x}(t)$ by filtering the ith component $u_i(t)$ of $\boldsymbol{u}(t)$, $i \in [p]$, as

$$x_i(t) = \sum_{\ell=0}^{\infty} c_{i\ell} u_i(t-\ell)\,, \tag{35}$$

$$c_{i\ell} := \frac{\Gamma(\ell + d_i)}{\Gamma(\ell+1)\Gamma(d_i)}\,, \quad d_i \in (0, 0.5)\,, \tag{36}$$

where $\Gamma(z)$ is the Gamma function $\Gamma(z) := \int_0^\infty t^{z-1}e^{-t}\,dt$, $z \in \mathbb{C}$, $\mathrm{Re}(z) > 0$. With L denoting the backward-shift operator $L\boldsymbol{u}(t) := \boldsymbol{u}(t-1)$, we have [16], [19], [20], [22]

$$x_i(t) = (1-L)^{-d_i} u_i(t)\,, \quad (1-L)^{d_i} x_i(t) = u_i(t) \tag{37}$$

and if $\{u_i(t)\}$ is stationary then so is $\{x_i(t)\}$ for $0 < d_i < 0.5$ [16], [19], [20], [22]. Note that $\lim_{\ell\to\infty} c_{i\ell} = \mathcal{O}(\ell^{d_i-1})$ [18], [20]. This ARFIMA model is called the ARFIMA(q, d_i, r) model.

The PSD of $\{x_i(t)\}$ is given by [16], [20], [22]

$$S_{x_i}(f) = (1 - e^{-\iota 2\pi f})^{-d_i} S_{u_i}(f)(1 - e^{\iota 2\pi f})^{-d_i}\,. \tag{38}$$

Define a diagonal matrix $\boldsymbol{D} = \mathrm{diag}\{d_1, d_2, \cdots, d_p\} \in \mathbb{R}^p$ and for a scalar a, let a^D denote $\mathrm{diag}\{a^{d_1}, a^{d_2}, \cdots, a^{d_p}\}$. The PSD of the ARFIMA process (34)-(37) is given by [19]

$$\boldsymbol{S}_x(f) = (1 - e^{-\iota 2\pi f})^{-\boldsymbol{D}} \boldsymbol{S}_u(f)(1 - e^{\iota 2\pi f})^{-\boldsymbol{D}} \tag{39}$$

where

$$\boldsymbol{S}_u(f) = \mathcal{A}^{-1}(e^{\iota 2\pi f})\mathcal{B}(e^{\iota 2\pi f})\mathcal{B}^H(e^{\iota 2\pi f})(\mathcal{A}^{-1}(e^{\iota 2\pi f}))^H\,. \tag{40}$$

It follows from Sec. III-B1, (39) and (40) that $\boldsymbol{S}_x(f) \succ \boldsymbol{0}$ and $\boldsymbol{S}_x^{-1}(f) \succ \boldsymbol{0}$ for $f \in [-0.5, 0.5]\backslash\{0\}$, and some or all elements of $\boldsymbol{S}_x^{-1}(0)$ are zero and some or all elements of $\lim_{f\to 0} \boldsymbol{S}_x(f)$ tend to ∞. As $f \to 0^+$, $(1 - e^{-\iota 2\pi f})^{-\boldsymbol{D}} \to (\iota 2\pi f)^{-\boldsymbol{D}}$, therefore,

$$\lim_{f\to 0^+} \boldsymbol{S}_x(f) \propto f^{-\boldsymbol{D}} \mathcal{A}^{-1}(1)\mathcal{B}(1)\mathcal{B}^H(1)(\mathcal{A}^{-1}(1))^H f^{-\boldsymbol{D}}\,. \tag{41}$$

The expression (41), together with (39) and (40), satisfies the spectral domain definition of a multivariate stationary LRD process given in [21, Definition 2.2]. It then follows from [21, Prop. 3.1] that for large $|\tau|$, the autocorrelation function $\boldsymbol{R}_x(\tau)$ of $\{\boldsymbol{x}(t)\}$ satisfies

$$[\boldsymbol{R}_x(\tau)]_{\ell k} \propto [\bar{\boldsymbol{R}}]_{\ell k} \tau^{d_\ell + d_k - 1}\,, \quad \ell, k \in [p], \tag{42}$$

for some $\bar{\boldsymbol{R}} \in \mathbb{R}^{p\times p}$ such that $[\bar{\boldsymbol{R}}]_{ii} \neq 0$, $i \in [p]$. Thus, $[\boldsymbol{R}_x(\tau)]_{\ell k} = \mathcal{O}(\tau^{d_\ell + d_k - 1})$ as $|\tau| \to \infty$. Setting

$\gamma = 1 - \max_{\ell\in[p]} d_\ell$, it then follows from [21, Prop. 3.1] that $[\boldsymbol{R}_x(\tau)]_{\ell k} = \mathcal{O}(\tau^{d_\ell+d_k-1}) = \mathcal{O}(\tau^{1-2\gamma})$, $|\tau| > 0$. For $d_\ell \in (0, 0.5)$, $\ell \in [p]$, we have $\gamma \in (0.5, 1)$ and therefore, $\boldsymbol{R}_x(\tau)$ is not absolutely summable, implying an LRD process [21], [22], [48], [53].

Remark 3: For the ARFIMA model we have $\boldsymbol{S}_x^{-1}(f) = (1 - e^{\iota 2\pi f})^{\boldsymbol{D}} \boldsymbol{S}_u^{-1}(f)(1 - e^{-\iota 2\pi f})^{\boldsymbol{D}}$ with the ARMA inverse PSD $\boldsymbol{S}_u^{-1}(f) = \mathcal{A}^H(e^{\iota 2\pi f})(\mathcal{B}^{-1}(e^{\iota 2\pi f}))^H \mathcal{B}^{-1}(e^{\iota 2\pi f})\mathcal{A}(e^{\iota 2\pi f})$. Therefore,

$$[\boldsymbol{S}_x^{-1}(f)]_{ij} = [\boldsymbol{S}_u^{-1}(f)]_{ij}(1 - e^{\iota 2\pi f})^{d_i}(1 - e^{-\iota 2\pi f})^{d_j} \quad (43)$$

for $i, j \in [p]$, implying $[\boldsymbol{S}_x^{-1}(f)]_{ij} = 0 \Leftrightarrow [\boldsymbol{S}_u^{-1}(f)]_{ij} = 0$ for any $f \in [-0.5, 0.5] \setminus \{0\}$. Therefore, the conditional independence structure of the ARFIMA process $\{\boldsymbol{x}(t)\}$ coincides with that of the underlying ARMA process $\{\boldsymbol{u}(t)\}$. □

C. SUMMARY AND DISCUSSION OF MODEL ASSUMPTIONS

To summarize, we require $\{\boldsymbol{x}(t)\}$ to be a stationary, zero-mean, $p-$dimensional multivariate Gaussian time series of dimension p such that assumption (A1) holds and its PSD $\boldsymbol{S}_x(f) \succ \boldsymbol{0}$. The vector ARMA and ARFIMA models discussed in Secs. III-B1 and III-B2, respectively, generate such sequences. Assumption (A1) quantifies the temporal dependence structure of $\{\boldsymbol{x}(t)\}$. If $d_i \geq 0.5$ in (36), then $x_i(t)$ in (35) is non-stationary [16], [19], [20], [22] and $\gamma \leq 0.5$ in (25). Later, to establish some theoretical results in Sec. V, we requires additional assumptions (A2) and (A3) whose implications are discussed therein after their statements.

The assumptions that $\{\boldsymbol{x}(t)\}$ is stationary and linear (i.e., generated via (24)) are critical to the results of this paper.

In this paper we are interested in CIGs for time series, as defined in [5]. As noted in [5], if $\{\boldsymbol{x}(t)\}$ is not necessarily Gaussian, then the zeros of $\boldsymbol{S}_x^{-1}(f)$ encode a partial correlation graph which is not necessarily a CIG since uncorrelation does not necessarily imply independence for non-Gaussian processes. For non-Gaussian sequences, our time-delay embedding approach yields an approximation to the underlying time series partial correlation graph. In Lemma 1, one replaces the conditional expectations with the corresponding optimum linear estimators, e.g., $E\{z_i(t)|\boldsymbol{z}_{-ij}(t)\}$ in (17) is replaced with the optimum linear estimator of $z_i(t)$ based on a linear weighted sum of elements in $\boldsymbol{z}_{-ij}(t)$ (see [5, Sec. 2]), and similarly, in (21), $E\{x_i(t')|\tilde{\boldsymbol{x}}_{-ij;t,d}\}$ is replaced with the optimum linear estimator of $x_i(t')$ based on a linear weighted sum of elements in $\tilde{\boldsymbol{x}}_{-ij;t,d}$. For Gaussian processes the conditional expectations equal the optimum linear estimators. In Sec. VI-A5 we present some simulation results for non-Gaussian sequences (using either exponential or Laplacian $\boldsymbol{e}(t)$ in (24), instead of Gaussian $\boldsymbol{e}(t)$), with all sequences, Gaussian or non-Gaussian, having identical PSDs and therefore, identical partial correlation graphs. These results show comparable performances for Gaussian and non-Gaussian sequences in graph recovery.

IV. OBJECTIVE FUNCTION AND ITS OPTIMIZATION

As noted in Remark 1, under delay embedding, we have a multi-attribute graph learning problem which has been addressed in [36] and [42] with lasso regularization and in [43] with non-convex log-sum and SCAD regularizations, except that [36], [42], and [43] address estimation from i.i.d. data whereas in this paper we have dependent data. Nevertheless we will use the log-likelihood of [36], [42], and [43] as our pseudo log-likelihood for dependent data. We show in Sec. V (see Theorems 1-3 therein) that when one chooses $\boldsymbol{\Omega}_y$ to minimize the resulting objective function, the estimator of $\boldsymbol{\Omega}_y$ has several desirable properties resulting in estimator consistency (Theorem 1), uniqueness (Theorem 2) and graph recovery (Theorem 3). Also, the resulting objective function and its optimization is just as in [43] to which one should refer to for details. In this section we introduce the objective and the regularization functions to introduce notation (and MCP, not discussed in [43]), and to provide some details regarding optimization that are used later for theoretical analysis in Sec. V and for numerical results in Secs. VI and VII

A. PENALIZED NEGATIVE PSEUDO LOG-LIKELIHOOD

Consider a finite set of data comprised of n zero-mean observations $\boldsymbol{x}(t)$, $t = 1, 2, \cdots, n$. Pick $d > 0$ and as in (14), construct $\boldsymbol{y}(t)$ for $t = d+1, d+2, \cdots, n$ with sample size $\bar{n} = n - d$. Define the sample covariance

$$\hat{\boldsymbol{\Sigma}}_y = \frac{1}{\bar{n}} \sum_{t=d+1}^{n} \boldsymbol{y}(t)\boldsymbol{y}^\top(t). \quad (44)$$

If the vector sequence $\{\boldsymbol{y}(t)\}_{t=d+1}^n$ were i.i.d., the negative log-likelihood (up to some constants) would be given by [42]

$$\mathcal{L}(\boldsymbol{\Omega}_y) := -\ln(|\boldsymbol{\Omega}_y|) + tr\left(\hat{\boldsymbol{\Sigma}}_y \boldsymbol{\Omega}_y\right). \quad (45)$$

We will still use this expression as a pseudo log-likelihood and consider the penalized negative pseudo log-likelihood $\bar{\mathcal{L}}(\boldsymbol{\Omega}_y)$ given by

$$\bar{\mathcal{L}}(\boldsymbol{\Omega}_y) = \mathcal{L}(\boldsymbol{\Omega}_y) + \alpha P_e(\boldsymbol{\Omega}_y) + (1-\alpha) P_g(\boldsymbol{\Omega}_y), \quad (46)$$

$$P_e(\boldsymbol{\Omega}_y) = \sum_{i\neq j}^{mp} \rho_\lambda\left(\left|[\boldsymbol{\Omega}_y]_{ij}\right|\right), \quad (47)$$

$$P_g(\boldsymbol{\Omega}_y) = m \sum_{k\neq \ell}^{p} \rho_\lambda\left(\|\boldsymbol{\Omega}_y^{(k\ell)}\|_F\right) \quad (48)$$

where $\boldsymbol{\Omega}_y^{(k\ell)} \in \mathbb{R}^{m\times m}$ is defined as in (15), $\lambda > 0$, $\alpha \in [0, 1]$, m in (48) reflects the number of group variables [54], and for $u \in \mathbb{R}$, $\rho_\lambda(u)$ is a penalty function that is function of $|u|$. In (47), the penalty term is applied to each off-diagonal element of $\boldsymbol{\Omega}$ and in (48), the penalty term is applied to the off-block-diagonal group of m^2 terms via $\boldsymbol{\Omega}^{(k\ell)}$. The parameter $\alpha \in [0, 1]$ "balances" element-wise and group-wise penalties [42], [44], [55], yielding a convex combination of the two.

1) PENALTY FUNCTIONS

We consider the following:

- *Lasso*. For some $\lambda > 0$, $\rho_\lambda(u) = \lambda|u|$, $u \in \mathbb{R}$. It is a convex function that is widely used.
- *Log-sum* [27]. For some $\lambda > 0$ and $1 \gg \epsilon > 0$, $\rho_\lambda(u) = \lambda\epsilon \ln\left(1 + \frac{|u|}{\epsilon}\right)$. It is a non-convex function.
- *Smoothly Clipped Absolute Deviation (SCAD)*. [34], [49] For some $\lambda > 0$ and $a > 2$, $\rho_\lambda(u) = \lambda|u|$ for $|u| \le \lambda$, $= (2a\lambda|u| - |u|^2 - \lambda^2)/(2(a-1))$ for $\lambda < |u| < a\lambda$, and $= \lambda^2(a+1)/2$ for $|u| \ge a$. It is a non-convex function. We have $\rho'_\lambda(u) = \frac{d\rho_\lambda(u)}{du} = 0$ for $|u| \ge a\lambda$.
- *Minimax Concave Penalty (MCP)* [35]. For some $\lambda > 0$ and $b > 0$, $\rho_\lambda(u) = \lambda|u| - \frac{u^2}{2b}$ for $|u| \le b\lambda$, and $= \frac{b\lambda^2}{2}$ for $|u| \ge b\lambda$. In [56] $b > 1$ is taken. We have $\rho'_\lambda(u) = 0$ for $|u| \ge b\lambda$.

The MCP penalty has not been considered in [36], [42], and [43]. As shown in [45, Sec. 2.2], $\rho_\lambda(u) = \rho_\lambda(-u)$ and differentiable for $u \ne 0$ with $\lim_{u\to 0^+} \rho'_\lambda(u) = \lambda$. Furthermore, $\rho_\lambda(u) + \frac{\mu}{2}u^2$ is convex, for some $\mu \ge 0$. It is shown in [45, Appendix A.1], that $\mu = 0$ for lasso, $\mu = 1/(a-1)$ for SCAD and $\mu = 1/b$ for MCP. For the log-sum penalty $\mu = \frac{\lambda}{\epsilon}$ [43]. The above penalties also satisfy:

(i) For some $C_\lambda > 0$ and $\delta_\lambda > 0$,

$$\rho_\lambda(u) \ge C_\lambda|u| \text{ for } |u| \le \delta_\lambda\,. \tag{49}$$

(ii) $\frac{d\rho_\lambda(u)}{d|u|} \le \lambda$ for $u \ne 0$.

Property (ii) is straightforward to verify. For lasso, $C_\lambda = \lambda$ and $\delta_\lambda = \infty$. For SCAD, $C_\lambda = \lambda$ and $\delta_\lambda = \lambda$. For the log-sum penalty, $C_\lambda = \frac{\lambda}{2}$ and $\delta_\lambda = \epsilon$ [43]. We take $C_\lambda = \frac{\lambda}{2}$ for all penalties (in proving Theorem 1), in which case (49) holds for MCP with $\delta_\lambda = b\lambda$.

B. OPTIMIZATION

For non-convex $\rho_\lambda(u)$, we use a local linear approximation (LLA) as in [43], following [34] and [57], to yield

$$\rho_\lambda(u) \approx \rho_\lambda(|u_0|) + \rho'_\lambda(|u_0|)(|u| - |u_0|) \;\Rightarrow\; \rho'_\lambda(|u_0|)|u|\,, \tag{50}$$

where u_0 is an initial guess, and $\rho'_\lambda(|u_0|)$ is the penalty function derivative at $|u| = |u_0|$. Therefore, with u_0 fixed, we consider only the last term above for optimization w.r.t. u. By [57, Theorem 1], the LLA provides a majorization of the non-convex penalty, thereby yielding a majorization-minimization approach. Suppose the initial guess is $\bar{\mathbf{\Omega}}_y$, then with LLA, the original objective function is transformed to its convex LLA approximation

$$\tilde{\mathcal{L}}(\mathbf{\Omega}_y) = \mathcal{L}(\mathbf{\Omega}_y) + \alpha\tilde{P}_e(\mathbf{\Omega}_y) + (1-\alpha)\tilde{P}_g(\mathbf{\Omega}_y), \tag{51}$$

$$\tilde{P}_e(\mathbf{\Omega}_y) = \sum_{i\ne j}^{mp} \lambda_{e,ij}\Big|[\mathbf{\Omega}_y]_{ij}\Big|, \tag{52}$$

Algorithm 1 ADMM Algorithm

Input: Data $\{\boldsymbol{x}(t)\}_{t=1}^n$, delay $d \ge 0$, $m = d+1$, regularization parameters λ and α, ADMM penalty parameter $\rho = \bar{\rho}$, tolerances τ_{abs} and τ_{rel}, variable penalty factor $\bar{\phi}$, maximum number of iterations $r_{\max}$. Initial guess $\bar{\mathbf{\Omega}}$.

Output: estimated inverse covariance $\hat{\mathbf{\Omega}}_{y\lambda}$ and edge-set $\hat{\mathcal{E}}$

1: Construct $\boldsymbol{y}(t)$ for $t = d+1, d+2, \cdots, n$ with sample size $\bar{n} = n - d$, as in (14). Calculate sample covariance $\hat{\mathbf{\Sigma}}_y = \frac{1}{\bar{n}}\sum_{t=d+1}^n \boldsymbol{y}(t)\boldsymbol{y}^\top(t)$. For lasso, $\lambda_{e,ij} = \lambda$ ($\forall i,j \in [mp]$), $\lambda_{g,k\ell} = \lambda$ ($k,\ell \in [p]$). For non-convex penalties, set $\lambda_{e,ij}$ ($i,j \in [mp]$) and $\lambda_{g,k\ell}$ ($k,\ell \in [p]$) as in (54), (55)-(56), or (57)-(58), for LSP, SCAD or MCP, respectively, using the initial guess $\bar{\mathbf{\Omega}}$.

2: Initialize: $\boldsymbol{U}^{(0)} = \boldsymbol{V}^{(0)} = \mathbf{0}$, where $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{R}^{(mp)\times(mp)}$, $\mathbf{\Omega}_{y\lambda}^{(0)} = \bar{\mathbf{\Omega}}$, $\rho^{(0)} = \bar{\rho}$.

3: converged = **false**, $r = 0$

4: **while** converged = **false** **and** $r \le r_{max}$, **do**

5: Eigen-decompose $\hat{\mathbf{\Sigma}}_y - \rho^{(r)}\left(\boldsymbol{V}^{(r)} - \boldsymbol{U}^{(r)}\right)$ as $\hat{\mathbf{\Sigma}}_y - \rho^{(r)}\left(\boldsymbol{V}^{(r)} - \boldsymbol{U}^{(r)}\right) = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^\top$ with diagonal matrix $\boldsymbol{D}$ consisting of eigenvalues. Define diagonal matrix $\tilde{\boldsymbol{D}}$ with ℓth diagonal element $\tilde{\boldsymbol{D}}_{\ell\ell} = (-\boldsymbol{D}_{\ell\ell} + \sqrt{\boldsymbol{D}_{\ell\ell}^2 + 4\rho^{(r)}})/(2\rho^{(r)})$. Set $\mathbf{\Omega}_{y\lambda}^{(r+1)} = \boldsymbol{P}\tilde{\boldsymbol{D}}\boldsymbol{P}^\top$.

6: Define soft thresholding scalar operator $T(a,\beta) := (1 - \beta/|a|)_+ a$ where $(a)_+ := \max(0,a)$. Set $\boldsymbol{A}^{(k\ell)} = (\mathbf{\Omega}_{y\lambda}^{(r+1)})^{(k\ell)} + (\boldsymbol{U}^{(r)})^{(k\ell)}$. The diagonal $m\times m$ subblocks of $\boldsymbol{V}$ are updated as

$$[(\boldsymbol{V}^{(r+1)})^{(kk)}]_{uv} = \begin{cases} [\boldsymbol{A}^{(kk)}]_{uu} & \text{if } u = v \\ T([\boldsymbol{A}_k^{(kk)}]_{uv}, \frac{\alpha\lambda_{e,ij}}{\rho^{(r)}}) & \text{if } u \ne v \end{cases}$$

$k \in [p]$, $u,v \in [m]$, $i = (k-1)m+u$, $j = (k-1)m+v$. The off-diagonal $m\times m$ subblocks of $\boldsymbol{V}$ are updated as

$$(\boldsymbol{V}^{(r+1)})^{(k\ell)} = \boldsymbol{B}\Big(1 - \frac{(1-\alpha)m\lambda_{g,k\ell}}{\rho^{(r)}\|\boldsymbol{B}\|_F}\Big)_+$$

where $k \ne \ell \in [p]$, $m\times m\boldsymbol{B}$ has its (u,v)th element as $[\boldsymbol{B}]_{uv} = T([\boldsymbol{A}^{(k\ell)}]_{uv}, \alpha\lambda_{e,ij}/\rho^{(r)})$, $i = (k-1)m+u$, $j = (\ell-1)m+v$.

7: Dual update $\boldsymbol{U}^{(r+1)} = \boldsymbol{U}^{(r)} + \left(\mathbf{\Omega}_{y\lambda}^{(r+1)} - \boldsymbol{V}^{(r+1)}\right)$.

8: Check convergence. Set tolerances

$$\tau_{pri} = mp\,\tau_{abs} + \tau_{rel}\,\max(\|\mathbf{\Omega}_{y\lambda}^{(r+1)}\|_F, \|\boldsymbol{V}^{(r+1)}\|_F)$$
$$\tau_{dual} = mp\,\tau_{abs} + \tau_{rel}\,\|\boldsymbol{U}^{(r+1)}\|_F/\rho^{(r)}\,.$$

Define $d_p = \|\mathbf{\Omega}_{y\lambda}^{(r+1)} - \boldsymbol{V}^{(r+1)}\|_F$ and $d_d = \rho^{(r)}\|\boldsymbol{V}^{(r+1)} - \boldsymbol{V}^{(r)}\|_F$. If $(d_p \le \tau_{pri})$ **and** $(d_d \le \tau_{dual})$, set converged = **true**.

9: Update penalty parameter ρ: $\rho^{(r+1)} = 2\rho^{(r)}$ if $d_p > \bar{\phi}d_d$, $= \rho^{(r)}/2$ if $d_d > \bar{\phi}d_p$, $= \rho^{(r)}$ otherwise. We also need to set $\boldsymbol{U}^{(r+1)} = \boldsymbol{U}^{(r+1)}/2$ for $d_p > \bar{\phi}d_d$ and $\boldsymbol{U}^{(r+1)} = 2\boldsymbol{U}^{(r+1)}$ for $d_d > \bar{\phi}d_p$.

10: $r \leftarrow r+1$

11: **end while**

12: For $k \ne \ell$, if $\|\boldsymbol{V}^{(k\ell)}\|_F > 0$, assign edge $\{k,\ell\} \in \hat{\mathcal{E}}$, else $\{k,\ell\} \notin \hat{\mathcal{E}}$. Inverse covariance estimate $\hat{\mathbf{\Omega}}_{y\lambda} = \boldsymbol{V}$.

$$\tilde{P}_g(\mathbf{\Omega}_y) = m \sum_{k\neq\ell}^{p} \lambda_{g,k\ell} \|\mathbf{\Omega}_y^{(k\ell)}\|_F \tag{53}$$

where $\lambda_{e,ij} = \rho'_\lambda(|[\bar{\mathbf{\Omega}}_y]_{ij}|)$ and $\lambda_{g,k\ell} = \rho'_\lambda(\|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F)$. We have for the log-sum penalty (LSP),

$$\lambda_{e,ij} := \frac{\lambda\epsilon}{|[\bar{\mathbf{\Omega}}_y]_{ij}| + \epsilon}, \quad \lambda_{g,k\ell} := \frac{\lambda\epsilon}{\|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F + \epsilon}, \tag{54}$$

for SCAD,

$$\lambda_{e,ij} = \begin{cases} \lambda, & \text{if } |[\bar{\mathbf{\Omega}}_y]_{ij}| \le \lambda \\ \frac{a\lambda - |[\bar{\mathbf{\Omega}}_y]_{ij}|}{a-1}, & \text{if } \lambda < |[\bar{\mathbf{\Omega}}_y]_{ij}| \le a\lambda \\ 0, & \text{if } a\lambda < |[\bar{\mathbf{\Omega}}_y]_{ij}| \end{cases}, \tag{55}$$

$$\lambda_{g,k\ell} = \begin{cases} \lambda, & \text{if } \|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F \le \lambda \\ \frac{a\lambda - \|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F}{a-1}, & \text{if } \lambda < \|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F \le a\lambda \\ 0, & \text{if } a\lambda < \|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F \end{cases}, \tag{56}$$

and for MCP,

$$\lambda_{e,ij} = \begin{cases} \lambda - \frac{|[\bar{\mathbf{\Omega}}_y]_{ij}|}{b}, & \text{if } |[\bar{\mathbf{\Omega}}_y]_{ij}| \le b\lambda \\ 0, & \text{if } b\lambda < |[\bar{\mathbf{\Omega}}_y]_{ij}| \end{cases}, \tag{57}$$

$$\lambda_{g,k\ell} = \begin{cases} \lambda - \frac{\|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F|}{b}, & \text{if } \|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F \le b\lambda \\ 0, & \text{if } b\lambda < \|\bar{\mathbf{\Omega}}_y^{(k\ell)}\|_F \end{cases}. \tag{58}$$

For lasso, we have $\lambda_{e,ij} = \lambda \forall i,j$ and $\lambda_{g,k\ell} = \lambda \forall k,\ell$.

We use the ADMM approach as described in detail in Algorithm 1 in [43, Sec. IV] to minimize $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ in (46) w.r.t. $\mathbf{\Omega}_y$ for both convex lasso and non-convex LSP/SCAD/MCP regularizations. The initial guess $\bar{\mathbf{\Omega}}_y$ for LLA is taken to be the lasso penalized solution. Using variable splitting [59], consider

$$\min_{\mathbf{\Omega}_y \succ \mathbf{0}, \boldsymbol{V}} \left\{ \mathcal{L}(\mathbf{\Omega}_y) + \alpha \tilde{P}_e(\boldsymbol{V}) + (1-\alpha)\tilde{P}_g(\boldsymbol{V}) \right\} \tag{59}$$

$$\text{subject to } \mathbf{\Omega}_y = \boldsymbol{V}, \tag{60}$$

where $\boldsymbol{V} \in \mathbb{R}^{(mp)\times(mp)}$ results from variable splitting, and in the penalties we use $\boldsymbol{V}$ instead of $\mathbf{\Omega}_y$, adding the equality constraint $\boldsymbol{V} = \mathbf{\Omega}$. The scaled augmented Lagrangian [59] for this problem after variable splitting, is given by

$$\bar{\mathcal{L}}_\rho(\mathbf{\Omega}_y, \boldsymbol{V}, \boldsymbol{U}) = \mathcal{L}(\mathbf{\Omega}_y) + \alpha\tilde{P}_e(\boldsymbol{V}) + (1-\alpha)\tilde{P}_g(\boldsymbol{V}) + \frac{\rho}{2}\|\mathbf{\Omega}_y - \boldsymbol{V} + \boldsymbol{U}\|_F^2, \tag{61}$$

where $\boldsymbol{U}$ is the dual variable and $\rho > 0$ is the "penalty parameter" [59]. Given the results $\mathbf{\Omega}^{(r)}, \boldsymbol{V}^{(r)}, \boldsymbol{U}^{(r)}$ of the rth iteration, in the $(r+1)$st iteration, an ADMM algorithm executes the following three updates:

(a) $\mathbf{\Omega}_{y\lambda}^{(r+1)} \leftarrow \arg\min_{\mathbf{\Omega}_y} L_a(\mathbf{\Omega}_y)$, $L_a(\mathbf{\Omega}_y) := \mathcal{L}(\mathbf{\Omega}_y) + \frac{\rho}{2}\|\mathbf{\Omega}_y - \boldsymbol{V}^{(r)} + \boldsymbol{U}^{(r)}\|_F^2$

(b) $\boldsymbol{V}^{(r+1)} \leftarrow \arg\min_{\boldsymbol{V}} L_b(\boldsymbol{V})$, $L_b(\boldsymbol{V}) := \alpha\tilde{P}_e(\boldsymbol{V}) + (1-\alpha)\tilde{P}_g(\boldsymbol{V}) + \frac{\rho}{2}\|\mathbf{\Omega}_{y\lambda}^{(r+1)} - \boldsymbol{V} + \boldsymbol{U}^{(r)}\|_F^2$

(c) $\boldsymbol{U}^{(r+1)} \leftarrow \boldsymbol{U}^{(r)} + \left(\mathbf{\Omega}_{y\lambda}^{(r+1)} - \boldsymbol{V}^{(r+1)}\right)$

A pseudo code for the ADMM algorithm used in this paper is given in Algorithm 1 where we use the stopping (convergence) criterion following [59, Sec. 3.3.1] and varying penalty parameter ρ following [59, Sec. 3.4.1]. See [42] for further details; note that by construction, $\mathbf{\Omega}_{y\lambda}^{(r+1)}$ in step 5 of Algorithm 1 is positive definite.

For non-convex penalties, we have an iterative solution: first solve with sparse group-lasso penalty, then use the LLA formulation and solve the resulting adaptive lasso type convex problem. In practice, just two iterations seem to be enough.

1) OPTIMIZATION ALGORITHM SUMMARY

Our ADMM-based optimization algorithm is as follows.

1. Given data $\{\boldsymbol{x}(t)\}_{t=1}^n$ and chosen delay $d \ge 0, m = d+1$, construct $\boldsymbol{y}(t)$ for $t = d+1, d+2, \cdots, n$ with sample size $\bar{n} = n - d$, as in (14). Calculate sample covariance $\hat{\mathbf{\Sigma}}_y = \frac{1}{\bar{n}}\sum_{t=d+1}^n \boldsymbol{y}(t)\boldsymbol{y}^\top(t)$. Initialize iteration $\tilde{m} = 1$, $\mathbf{\Omega}_{y\lambda}^{(0)} = (\text{diag}(\hat{\mathbf{\Sigma}}_y))^{-1}$, $\bar{\mathbf{\Omega}} = \mathbf{\Omega}_{y\lambda}^{(0)}$ and use $\bar{\mathbf{\Omega}}$ to compute $\lambda_{e,ij}$'s and $\lambda_{g,k\ell}$'s.
2. Execute Algorithm 1 with initial guess $\bar{\mathbf{\Omega}}$ to obtain the estimate $\hat{\mathbf{\Omega}}_{y\lambda}$.
3. Quit if using sparse-group lasso, else set $\mathbf{\Omega}_{y\lambda}^{(\tilde{m})} = \hat{\mathbf{\Omega}}_{y\lambda}$ and $\bar{\mathbf{\Omega}} = \mathbf{\Omega}_{y\lambda}^{(\tilde{m})}$ to re-compute $\lambda_{e,ij}$'s and $\lambda_{g,k\ell}$'s via the LLA. Set $\tilde{m} \leftarrow \tilde{m} + 1$.
4. Repeat steps 2 and 3 until convergence. The converged $\hat{\mathbf{\Omega}}_{y\lambda}$ is the final estimate of the inverse covariance. (For the numerical results shown in Sec. VI, we terminated after two iterations of steps 2 and 3, similar to [34], [57].)
5. Given the estimate $\hat{\mathbf{\Omega}}_{y\lambda}$ resulting from the ADMM algorithm, we estimate the graph edge-set as

$$\hat{\mathcal{E}} = \left\{ \{k,\ell\} : \|\hat{\mathbf{\Omega}}_{y\lambda}^{(k\ell)}\|_F > 0,\ k,\ell \in [p] \right\}. \tag{62}$$

A pseudocode for the above ADMM algorithm is given in Algorithm 2. It utilizes Algorithm 1 in each LLA step.

For the numerical results presented in the paper, we picked $r_{\max} = 200$, $\bar{\rho} = 2$, $\bar{\phi} = 10$, $\tau_{abs} = \tau_{rel} = 10^{-4}$, and $\tilde{m}_{\max} = 1$ for lasso and $= 2$ for LSP, SCAD and MCP penalties in Algorithms 1 and 2. For the SCAD penalty $a = 3.7$ (as in [34]), for MCP $b = 3.7$, and for the log-sum penalty $\epsilon = 0.0001$.

C. COMPUTATIONAL COMPLEXITY

Each iteration of the ADMM algorithm 1 has three updates (a), (b) and (c), implemented in steps 5, 6 and 7, respectively, of Algorithm 1. Step 5 requires an eigen-decomposition of an $(mp) \times (mp)$ matrix, resulting in $\mathcal{O}((mp)^3) = \mathcal{O}(((d+1)p)^3)$ computational complexity. In step 6 of Algorithm 1 one first performs elementwise soft thresholding of an $(mp)\times(mp)$ matrix ($\mathcal{O}((mp)^2)$ complexity), followed by group soft thresholding of $p^2 m \times m$ submatrices ($\mathcal{O}((mp)^2)$ complexity). Step 7 involves additions/subtractions among three $(mp)\times(mp)$ matrices, resulting in $\mathcal{O}((mp)^2)$ complexity. Step 8 of Algorithm 1 requires computation of the Frobenius norms of three $(mp)\times(mp)$ matrices, resulting in $\mathcal{O}((mp)^2)$ complexity. The computational complexity in step 9 is $\mathcal{O}((mp)^2)$ as it

Algorithm 2 LLA-Based ADMM Algorithm for Optimizing (46)

Input: Data $\{\boldsymbol{x}(t)\}_{t=1}^n$, delay $d \geq 0$, $m = d+1$, regularization parameters λ and α, ADMM penalty parameter $\rho = \bar{\rho}$, tolerances τ_{abs} and τ_{rel}, variable penalty factor $\bar{\phi}$, maximum number of iterations $r_{\max}$ and $\tilde{m}_{\max}$. For lasso penalty, $\tilde{m}_{\max} = 1$.

Output: estimated inverse covariance $\hat{\boldsymbol{\Omega}}_{y\lambda}$ and edge-set $\hat{\mathcal{E}}$

1: Construct $\boldsymbol{y}(t)$ for $t = d+1, d+2, \cdots, n$ with sample size $\bar{n} = n-d$, as in (14). Calculate sample covariance $\hat{\boldsymbol{\Sigma}}_y = \frac{1}{\bar{n}} \sum_{t=d+1}^n \boldsymbol{y}(t)\boldsymbol{y}^\top(t)$.
2: Initialize $\boldsymbol{\Omega}_{y\lambda}^{(0)} = (\text{diag}(\hat{\boldsymbol{\Sigma}}_y))^{-1}$, and set $\lambda_{e,ij} = \lambda$, $\forall i, j \in [mp]$, and $\lambda_{g,k\ell} = \lambda$, $\forall k, \ell \in [p]$.
3: $\tilde{m} = 1$
4: **while** $\tilde{m} \leq \tilde{m}_{\max}$, **do**
5: Execute steps 2-12 of Algorithm 1, resulting in output $\boldsymbol{\Omega}_{y\lambda}$. Set $\boldsymbol{\Omega}_{y\lambda}^{(\tilde{m})} = \hat{\boldsymbol{\Omega}}_{y\lambda}$.
6: **if** LSP/SCAD/MCP **then**
7: Set $\bar{\boldsymbol{\Omega}} = \boldsymbol{\Omega}_{y\lambda}^{(\tilde{m})}$ and re-compute $\lambda_{e,ij}$'s and $\lambda_{g,k\ell}$'s via the LLA.
8: **end if**
9: $\tilde{m} \leftarrow \tilde{m} + 1$
10: **end while**
11: Denote the converged estimate by $\hat{\boldsymbol{\Omega}}_{y\lambda}$. For $k \neq \ell$, if $\|\hat{\boldsymbol{\Omega}}_{y\lambda}^{(k\ell)}\|_F > 0$, assign edge $\{k, \ell\} \in \hat{\mathcal{E}}$, else $\{k, \ell\} \notin \hat{\mathcal{E}}$.

involves scaling of an $(mp) \times (mp)$ matrix. Combining the results of steps 5-9, the computational complexity in each iteration of the ADMM algorithm is $\mathcal{O}(((d+1)p)^3)$. The sample covariance is computed just once, with a computational complexity of $\mathcal{O}(n(mp)^2)$. The number of iterations to convergence is problem dependent. If the number of iterations to convergence is N_c, the computational complexity would be $\mathcal{O}(n((d+1)p)^2 + N_c((d+1)p)^3)$.

We now turn to memory requirements in each iteration of the ADMM algorithm 1. The data $\{\boldsymbol{x}(t)\}_{t=1}^n$ needs storage for np real numbers while $\boldsymbol{y}(t)$ requires $(n-d)(d+1)p$ real numbers. We need eleven $(mp) \times (mp)$ matrices ($m = d+1$), $\hat{\boldsymbol{\Sigma}}_y$, $\boldsymbol{\Omega}_{y\lambda}^{(r)}$, $\boldsymbol{\Omega}_{y\lambda}^{(r+1)}$, $\boldsymbol{U}^{(r)}$, $\boldsymbol{U}^{(r+1)}$, $\boldsymbol{V}^{(r)}$, $\boldsymbol{V}^{(r+1)}$, $\boldsymbol{P}$, $\hat{\boldsymbol{\Sigma}}_y - \rho^{(r)}\left(\boldsymbol{V}^{(r)} - \boldsymbol{U}^{(r)}\right)$, $\boldsymbol{\Omega}_{y\lambda}^{(r+1)} - \boldsymbol{V}^{(r+1)}$ and $\boldsymbol{V}^{(r+1)} - \boldsymbol{V}^{(r)}$, each requiring storage of $((d+1)p)^2$ reals. Ignoring storage of $m \times m$ matrices $\boldsymbol{A}^{(k\ell)}$'s which can be reused for different k, ℓ values, the memory needs are $\mathcal{O}(n(d+2)p + 11((d+1)p)^2)$.

D. MODEL SELECTION

For model selection (i.e., regularization parameter selection), we follow the BIC (Bayesian information criterion) as discussed in [43, Sec. IV-A] (also [42]) even though it applies only to i.i.d. data. Let $\hat{\boldsymbol{\Omega}}_{y\lambda}$ and $\hat{\bar{\mathcal{E}}}$ $(= \{\{i,j\} : |[\hat{\boldsymbol{\Omega}}_{y\lambda}]_{ij}| > 0,\ i \neq j \in [mp]\})$ denote the estimated inverse covariance matrix of $\{\boldsymbol{y}(t)\}$ and estimated "enlarged" (element-level) edge-set, and let $|\hat{\bar{\mathcal{E}}}|$ denote the cardinality (# of nonzero elements) of $\hat{\bar{\mathcal{E}}}$. Noting that $\hat{\boldsymbol{\Omega}}_{y\lambda}$ is symmetric with nonzero diagonal elements, the number of free nonzero elements of $\hat{\boldsymbol{\Omega}}_{y\lambda}$ equal $\frac{1}{2}|\hat{\bar{\mathcal{E}}}| + pm$. The (pseudo-)BIC is then given by

$$\text{BIC}(\lambda, \alpha) = tr(\hat{\boldsymbol{\Sigma}}_y \hat{\boldsymbol{\Omega}}_{y\lambda}) - \ln(|\hat{\boldsymbol{\Omega}}_{y\lambda}|) + \frac{\ln(\bar{n})}{\bar{n}} \left(\frac{1}{2}|\hat{\bar{\mathcal{E}}}|\right) \quad (63)$$

based on the optimized negative pseudo log-likelihood $-\ln f_{\boldsymbol{Y}}(\boldsymbol{Y}) \propto \frac{\bar{n}}{2}\left(tr(\hat{\boldsymbol{\Sigma}}_y \hat{\boldsymbol{\Omega}}_{y\lambda}) - \ln|\hat{\boldsymbol{\Omega}}_{y\lambda}|\right)$, where $\boldsymbol{Y} = \{\boldsymbol{y}(t)\}_{t=d+1}^n$ and $f_{\boldsymbol{Y}}(\boldsymbol{Y})$ is the joint pseudo probability density function (pdf) of $\boldsymbol{Y}$ under the assumption that $\{\boldsymbol{y}(t)\}_{t=d+1}^n$ is i.i.d. We do not know the exact pdf of $\boldsymbol{Y}$ as $\{\boldsymbol{y}(t)\}_{t=d+1}^n$ is a correlated sequence. There are approaches in the literature [61] where one uses information criteria mimicking true BIC $(= -2\ln f_{\boldsymbol{Y}}(\boldsymbol{Y}) + \ln(\bar{n})(\#\text{ of free nonzero parameters}))$ but replacing the true likelihood with a suitable loss function that is used for parameter estimation. We have chosen to use a pseudo pdf under the i.i.d. assumption. It is a heuristic that seems to work well in our numerical results presented later.

Our objective is to select the pair (λ, α) to minimize BIC given by (63). To reduce computational burden, as a heuristic, instead of searching over a two-dimensional space (λ, α), we first search over a grid of λ values with fixed $\alpha = 0.05\,(= \alpha_0)$, (somewhat arbitrary, $\alpha = 0$ will make it group lasso). Then with λ fixed at the selected value, we search over a grid of α values over $[0, 1]$.

We search over λ values in the range $[\lambda_\ell, \lambda_u]$ selected via the following heuristic. For $\alpha = \alpha_0$, we first find the smallest λ, labeled λ_{sm}, for which we get a no-edge model (i.e., $|\hat{\mathcal{E}}| = 0$). To this end, searching over a grid of λ values, we find the smallest λ for which the corresponding $|\hat{\mathcal{E}}| \approx \tau_{th} \approx 0$, i.e, for which we get close to a no-edge model; we take this value of λ as λ_{sm}. We picked $\tau_{th} = 0.005\,p(p-1)/2$ (0.5% of possible $p(p-1)/2$ edges are connected). Then we set $\lambda_u = \lambda_{sm}/2$ and $\lambda_\ell = \lambda_u/10$. The given choice of λ_u precludes "extremely" sparse models while that of λ_ℓ precludes "very" dense models. We search over a grid of λ values in the range $[\lambda_\ell, \lambda_u]$ to minimize BIC.

For the numerical results presented in the paper for synthetic data, we picked eight logarithmically spaced values in the interval $[\lambda_\ell, \lambda_u]$ for λ selection and the grid $\{0.05, 0.1, 0.3, 0.5, 0.7, 0.9\}$ for α selection. For real data, we picked sixteen logarithmically spaced values in the interval $[\lambda_\ell, \lambda_u]$, with α grid as for synthetic data.

1) EXTENDED BIC

In the high-dimensional case where both the sample size n and the graph nodes p grow with n, BIC for Gaussian graphical models may not be consistent and additional penalty may be warranted [62] (see also [61] for other cases, such as regression problems). In [62] an extended BIC (eBIC) is proposed for Gaussian graphical models with p nodes, i.i.d. data $\{\boldsymbol{x}(t)\}_{t=1}^n =: \boldsymbol{X}$, joint pdf $f_{\boldsymbol{X}}(\boldsymbol{X})$ and edge-set $\mathcal{E}$, as

$$\text{eBIC}_{\bar{\gamma}}(\mathcal{E}) = -2\ln f_{\boldsymbol{X}}(\boldsymbol{X}) + \frac{|\mathcal{E}|}{2}\Big(\ln(n) + 4\bar{\gamma}\ln(p)\Big) \quad (64)$$

where $\bar{\gamma} \in [0, 1]$, and $|\mathcal{E}|$ counts all edges in the graph with $\{i, j\}$ and $\{j, i\}$, $i \neq j$, counted as distinct edges (hence division

by 2). In (64) there is additional penalty for the number of nodes p, and $\bar{\gamma}$ is a user chosen parameter. If $\bar{\gamma} = 0$, one has the classical BIC. Applying this concept of including the additional penalty, we propose an extended (pseudo-)BIC based on (63) as

$$\text{eBIC}(\lambda,\alpha) = tr(\hat{\mathbf{\Sigma}}_y\hat{\mathbf{\Omega}}_{y\lambda}) - \ln(|\hat{\mathbf{\Omega}}_{y\lambda}|) + \frac{\ln(\bar{n}) + 4\bar{\gamma}\ln(mp)}{\bar{n}}\left(\frac{1}{2}|\hat{\bar{\mathcal{E}}}|\right) \tag{65}$$

where $\bar{\gamma} \in [0, 1]$. Selection of (λ, α) to minimize $\text{eBIC}(\lambda, \alpha)$ is performed exactly as for $\text{BIC}(\lambda, \alpha)$.

V. THEORETICAL ANALYSIS

Here we analyze the properties of $\hat{\mathbf{\Omega}}_{y\lambda} = \arg\min_{\mathbf{\Omega}_y \succ \mathbf{0}} \bar{\mathcal{L}}(\mathbf{\Omega}_y)$. Since the SCAD, MCP and log-sum penalties are non-convex, the objective function is non-convex and in general, any optimization of the objective function will yield only a stationary point or a local minimum. The proofs of Theorems 1-3 in this section closely follow the corresponding results in [43, Sec. V] dealing with multi-attribute graph estimation from i.i.d. data under lasso, SCAD and log-sum penalties; [43] does not discuss MCP. The proofs of all theoretical results are given in the two appendices. In this section we explicitly allow p and λ to be a functions of sample size n, denoted as p_n and λ_n, respectively.

Let $\mathbf{\Sigma}_y^\diamond$ and $\mathbf{\Omega}_y^\diamond$ denote the true value of the covariance $\mathbf{\Sigma}_y = E\{\boldsymbol{y}(t)\boldsymbol{y}^\top(t)\}$ and its inverse $\mathbf{\Omega}_y = \mathbf{\Sigma}_y^{-1}$, respectively. In Lemma 2, proved in Appendix A, we bound the sample covariance estimation error $\hat{\mathbf{\Sigma}}_y - \mathbf{\Sigma}_y^\diamond$ with probability $> 1 - \frac{1}{p_n^{\tau-2}}$, $\tau > 2$. The error bound depends upon the sample size and the decay parameter γ in assumption (A1). Let $\hat{\mathbf{\Omega}}_{y\lambda} = \arg\min_{\mathbf{\Omega}_y \succ \mathbf{0}} \bar{\mathcal{L}}(\mathbf{\Omega}_y)$. Using Lemma 2, and assumptions (A2) and (A3) stated later, in Theorem 1 we establish an upperbound on the Frobenius norm of the error $\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond$. The proof of Theorem 1 follows the method of proof of [43, Theorem 1], which, in turn, for the most part, follows the method of proof of [38, Theorem 1] pertaining to lasso penalty.

Let $\mathbf{\Delta} := \mathbf{\Omega}_y - \mathbf{\Omega}_y^\diamond$. The proof of Theorem 1 examines the cost difference $\bar{\mathcal{L}}(\mathbf{\Omega}_y) - \bar{\mathcal{L}}(\mathbf{\Omega}_y^\diamond) = \sum_{i=1}^4 A_i$ where $A_1 = -\ln(|\mathbf{\Omega}_y^\diamond + \Delta|) + \ln(|\mathbf{\Omega}_y^\diamond|) + \text{tr}(\mathbf{\Sigma}_y^\diamond\mathbf{\Delta})$, $A_2 = \text{tr}((\hat{\mathbf{\Sigma}}_y - \mathbf{\Sigma}_y^\diamond)\mathbf{\Delta})$, $A_3 = \alpha(P_e(\mathbf{\Omega}_y^\diamond + \Delta) - P_e(\mathbf{\Omega}_y))$ and $A_4 = (1-\alpha)(P_g(\mathbf{\Omega}_y^\diamond + \Delta) - P_g(\mathbf{\Omega}_y))$. Consider the compact set $\Psi = \{\mathbf{\Delta} : \mathbf{\Delta} = \mathbf{\Delta}^\top,\ \|\mathbf{\Delta}\|_F \le B\}$ for some $B > 0$. The proof of Theorem 1 establishes that for some $\mathbf{\Delta} \in \Psi$ such that $\|\mathbf{\Delta}\|_F = B$, $\bar{\mathcal{L}}(\mathbf{\Omega}_y^\diamond + \Delta) - \bar{\mathcal{L}}(\mathbf{\Omega}_y^\diamond) > 0$. But the cost difference is zero when $\mathbf{\Delta} = \mathbf{0}$. Hence, there exists a local minimum of $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ over the set Ψ with Frobenius error norm bounded by B. In the proof, Lemma 2 is used to upperbound A_2 only. Assumptions (A2) and (A3) are used to upper/lower bound entities in A_i, $i \in [4]$, where the sparsity of $\mathbf{\Omega}_y^\diamond$ stated in assumption (A2) plays a critical role. For the lasso penalty, $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ is a convex function of $\mathbf{\Omega}_y$, therefore, the local minimizer $\hat{\mathbf{\Omega}}_{y\lambda}$ of Theorem 1 is a global minimizer.

Lemma 3 characterizes the local convexity of $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ for non-convex penalties. Theorem 1 and Lemma 3 are then used in Theorem 2 to establish conditions under which the local minimizer $\hat{\mathbf{\Omega}}_{y\lambda}$ of Theorem 1 is unique. Theorem 3 addresses graph recovery using the estimator $\hat{\mathbf{\Omega}}_{y\lambda}$ of Theorem 1.

We now turn to the statement of Lemma 2.

Lemma 2: Under assumption (A1), the sample covariance $\hat{\mathbf{\Sigma}}_y$ satisfies the tail bound

$$P\left(\max_{k,l}\left|[\hat{\mathbf{\Sigma}}_y - \mathbf{\Sigma}_y^\diamond]_{kl}\right| > \delta_{*n}\right) \le \frac{1}{p_n^{\tau-2}} \tag{66}$$

for $\tau > 2$, under the following conditions on $\delta_{*n} > 0$ and sample size $\bar{n} = n - d$:

(a) For $\gamma > \frac{3}{4}$,

$$\delta_{*n} = c_a^2(p_n)\sqrt{\ln(2m^2p_n^\tau)/(\bar{n}C_u)}\,, \tag{67}$$

and sample size satisfies $\bar{n} \ge N_{\delta_*,\gamma}$,

$$N_{\delta_*,\gamma} \ge \begin{cases} N_1, & \gamma \in (1,\infty) \\ \arg\min\{\bar{n} : \bar{n}/\ln^4(\bar{n}) \ge N_1\}, & \gamma = 1 \\ \arg\min\{\bar{n} : \bar{n}^{4\gamma-3} \ge N_1\}, & \gamma \in (\frac{3}{4}, 1) \end{cases}$$

where $C_u \in (0,\infty)$ is a constant that depends only on γ and

$$N_1 = \ln(2m^2p_n^\tau)/C_u\,. \tag{68}$$

(b) For $\gamma = \frac{3}{4}$,

$$\delta_{*n} = c_a^2(p_n)\sqrt{\ln(2m^2p_n^\tau)\,\ln(\bar{n})/(\bar{n}C_u)}\,, \tag{69}$$

and sample size satisfies $\bar{n} \ge \arg\min\{\bar{n} : \ln(\bar{n}) \ge N_1\}$.

(c) For $\gamma \in (\frac{1}{2}, \frac{3}{4})$,

$$\delta_{*n} = \begin{cases} c_a^2(p_n)\frac{\sqrt{\ln(2m^2p_n^\tau)/C_u}}{\bar{n}^{2\gamma-1}} & \text{if } N_1 \le 1 \\ c_a^2(p_n)\frac{\ln(2m^2p_n^\tau)}{\bar{n}^{2\gamma-1}C_u} & \text{if } N_1 > 1 \end{cases} \tag{70}$$

•

See also Remark 10 in Appendix A regarding the universal constant C_u.

Further assume

(A2) Let $\mathbf{\Sigma}_y^\diamond = E\{\boldsymbol{y}(t)\boldsymbol{y}^\top(t)\} \succ \mathbf{0}$ denote the true covariance of $\boldsymbol{y}(t)$. Define $\mathcal{E}_y^\diamond = \{\{i,j\} : (\mathbf{\Omega}_y^\diamond)^{(ij)} \ne \mathbf{0},\ i \ne j\}$ where $\mathbf{\Omega}_y^\diamond = (\mathbf{\Sigma}_y^\diamond)^{-1}$. Assume that $\text{card}(\mathcal{E}_y^\diamond) = |\mathcal{E}_y^\diamond| \le s_n^\diamond$.

(A3) The minimum and maximum eigenvalues of $\mathbf{\Sigma}_y^\diamond$ satisfy

$$0 < \beta_{\min} \le \phi_{\min}(\mathbf{\Sigma}_y^\diamond) \le \phi_{\max}(\mathbf{\Sigma}_y^\diamond) \le \beta_{\max} < \infty\,.$$

Here $\beta_{\min}$ and $\beta_{\max}$ are not functions of n.

Assumption (A2) is more of a definition in that it specifies that the edge-set determined by $\mathbf{\Omega}_y^\diamond$ has number of edges bounded by s_n. This bound influences the convergence rate in Theorem 1 as discussed later in Remark 6. Assumption (A2) is a sparsity condition. Assumption (A3) bounds the eigenvalues of the true covariance $\mathbf{\Sigma}_y^\diamond$ of $\{\boldsymbol{y}(t)\}$ from below as well as above. Since $\mathbf{\Omega}_y^\diamond = (\mathbf{\Sigma}_y^\diamond)^{-1}$, the eigenvalues of $\mathbf{\Omega}_y^\diamond$ are also bounded from below and above. Assumptions (A2) and (A3) are technical conditions needed to prove Theorem 1 using

the proof technique of [38, Theorem 1]. These conditions are mild.

Remark 4: Assumption (A3) is stated in terms of the covariance $\mathbf{\Sigma}_y^\diamond$ of augmented $\boldsymbol{y}(t)$. Now we relate it to the properties of $\{\boldsymbol{x}(t)\}$. Define

$$\tilde{\boldsymbol{y}}(t) := [\boldsymbol{x}^\top(t)\, \boldsymbol{x}^\top(t-1)\, \cdots\, \boldsymbol{x}^\top(t-d)]^\top . \tag{71}$$

Then $\tilde{\boldsymbol{y}}(t) = \boldsymbol{P}\boldsymbol{y}(t)$ for some permutation matrix $\boldsymbol{P}$ ($\boldsymbol{P}\boldsymbol{P}^\top = \boldsymbol{P}^\top\boldsymbol{P} = \boldsymbol{I}_{mp}$, $m = d+1$) and $\mathbf{\Sigma}_{\tilde{y}}^\diamond = E\{\tilde{\boldsymbol{y}}(t)\tilde{\boldsymbol{y}}^\top(t)\} = \boldsymbol{P}\mathbf{\Sigma}_y^\diamond\boldsymbol{P}^\top$. By [53, Prop. 2.3], for ARMA models discussed in Sec. III-B1, for any $d \geq 0$ we have

$$\begin{aligned}&\min_f \phi_{\min}(\boldsymbol{S}_x(f))\\ &\leq \phi_{\min}(\mathbf{\Sigma}_{\tilde{y}}^\diamond) = \phi_{\min}(\mathbf{\Sigma}_y^\diamond)\\ &\leq \phi_{\max}(\mathbf{\Sigma}_{\tilde{y}}^\diamond) = \phi_{\max}(\mathbf{\Sigma}_y^\diamond) \leq \max_f \phi_{\max}(\boldsymbol{S}_x(f))\end{aligned} \tag{72}$$

where we have used the fact that $\boldsymbol{P}$ is orthogonal, hence minimum and maximum eigenvalues of $\mathbf{\Sigma}_{\tilde{y}}^\diamond$ and $\mathbf{\Sigma}_y^\diamond$ are identical. One can take $\beta_{\min} = \min_f \phi_{\min}(\boldsymbol{S}_x(f))$ and $\beta_{\max} = \max_f \phi_{\max}(\boldsymbol{S}_x(f))$ for all values of d for ARMA models with $0 < \beta_{\min} \leq \beta_{\max} < \infty$. This is also true for $\{\boldsymbol{x}(t)\}$ satisfying assumption (A1) with $\gamma > 1$ [53, Sec. 2] so long as $\boldsymbol{S}_x(f) \succ \boldsymbol{0}$. Since $\beta_{\min} > 0$, we also have $\mathbf{\Sigma}_y^\diamond \succ \boldsymbol{0}$ for every $d \geq 0$. □

Remark 5: The results stated in Remark 4 for ARMA and related linear processes do not apply to ARFIMA models discussed in Sec. III-B2. By (39), $\lim_{f\to 0} \boldsymbol{S}_x(f) = \infty$. By [58, Cor. 4.5.11] of the Ostrowski's theorem [58, Cor. 4.5.9], for the ARFIMA process $\{\boldsymbol{x}(t)\}$ discussed in Sec. III-B2, we have

$$\begin{aligned}&|1-e^{-\iota 2\pi f}|^{-2\,\underline{\mathrm{d}}}\, \phi_{\min}(\boldsymbol{S}_u(f)) \leq \phi_{\min}(\boldsymbol{S}_x(f))\\ &\leq \phi_{\max}(\boldsymbol{S}_x(f)) \leq |1-e^{-\iota 2\pi f}|^{-2\,\bar{d}}\, \phi_{\max}(\boldsymbol{S}_u(f))\end{aligned} \tag{73}$$

where $\underline{\mathrm{d}} = \min\{d_i,\ i \in [p]\} \in (0, 0.5)$ and $\bar{d} = \max\{d_i,\ i \in [p]\} \in (0, 0.5)$. Since $\min_f 1/|1-e^{-\iota 2\pi f}|^{2\,\underline{\mathrm{d}}} = 1/2^{2\,\underline{\mathrm{d}}}$, we have

$$\min_f \phi_{\min}(\boldsymbol{S}_x(f)) \geq \big(\min_f \phi_{\min}(\boldsymbol{S}_u(f))\big)/4^{\underline{\mathrm{d}}} > 0 \tag{74}$$

and since $\max_f 1/|1-e^{-\iota 2\pi f}|^{2\,\underline{\mathrm{d}}} = \infty$, we have

$$\max_f \phi_{\max}(\boldsymbol{S}_x(f)) \leq \infty . \tag{75}$$

Thus, we may take $\beta_{\min}$ to be the lower bound in (74) applicable for any $d \geq 0$ whereas the upper bound in (75) is trivial and of no use. Qualitatively, the ARMA part of ARFIMA captures the short-range dependence ("smaller" lags in the auto-correlation function) while the fractional integration part captures the long-range dependence (much larger lags in the auto-correlation function, as $\tau \to \infty$) [16], [21], [22]). As noted in Remark 3, the conditional independence structure of the ARFIMA process $\{\boldsymbol{x}(t)\}$ coincides with that of the corresponding ARMA process $\{\boldsymbol{u}(t)\}$. By Remark 1, we would like d to be "large," but not too large so that only short range dependence is captured. For any finite d (maximum delay in delay embedding), $\phi_{\max}(\mathbf{\Sigma}_y^\diamond)$ will be finite. Consider a simple example with $\boldsymbol{u}(t) = \tilde{\boldsymbol{B}}_0\boldsymbol{e}(t)$, $q = 0$ and $r = 0$ in (34) and let $\mathbf{\Sigma}_u = \tilde{\boldsymbol{B}}_0\tilde{\boldsymbol{B}}_0^\top \succ \boldsymbol{0}$. Let $\tilde{\boldsymbol{B}}_0$ be such that $[\mathbf{\Sigma}_u]_{ii} = 1$ for $i \in [p]$. Suppose $d_i = \tilde{d}$, $i \in [p]$ in (37). Then $\boldsymbol{R}_x(\tau) = r(\tau) \otimes \mathbf{\Sigma}_u$ where $r(\tau)$ is the inverse discrete-time Fourier transform of $|1-e^{-\iota 2\pi f}|^{-2\,\tilde{d}}$, i.e.,

$$r(\tau) = \int_{-1/2}^{1/2} |1-e^{-\iota 2\pi f}|^{-2\,\tilde{d}}\, e^{\iota 2\pi f\tau}\, df\ ,$$

$\tau \in \mathcal{Z}$. Then we have

$$\mathbf{\Sigma}_{\tilde{y}}^\diamond = \mathbf{\Sigma}_r \otimes \mathbf{\Sigma}_u\,, \tag{76}$$

$$\mathbf{\Sigma}_r = \begin{bmatrix} r(0) & r(1) & \cdots & r(d) \\ r(-1) & r(0) & \cdots & r(d-1) \\ \vdots & \vdots & \ddots & \vdots \\ r(-d) & r(-d+1) & \cdots & r(0) \end{bmatrix}, \tag{77}$$

$$\phi_{\max}(\mathbf{\Sigma}_y^\diamond) = \phi_{\max}(\mathbf{\Sigma}_{\tilde{y}}^\diamond) = \phi_{\max}(\mathbf{\Sigma}_r)\, \phi_{\max}(\mathbf{\Sigma}_u)\,. \tag{78}$$

By [22, pp. 37-38] (note that $[\mathbf{\Sigma}_u]_{ii} = 1$ for $i \in [p]$)

$$r(\tau) = \alpha_\tau r(0)\,, \quad r(0) = \frac{\Gamma(1-2\tilde{d})}{(\Gamma(1-\tilde{d}))^2}\,, \tag{79}$$

$$\alpha_\tau = \prod_{\ell=1}^{\tau} \frac{\ell-1+\tilde{d}}{\ell-\tilde{d}}\,. \tag{80}$$

For large τ, $r(\tau) = \mathcal{O}(\tau^{2\tilde{d}-1})$ [22, p. 37]. For $\tilde{d} = 0.01, 0.35, 0.49$, we obtain $\phi_{\max}(\mathbf{\Sigma}_r) = 1.03, 5.04, 93.90$, respectively for $d = 5$, $\phi_{\max}(\mathbf{\Sigma}_r) = 1.04, 7.68, 170.03$, respectively for $d = 10$, and $\phi_{\max}(\mathbf{\Sigma}_r) = 1.06, 12.06, 320.39$, respectively for $d = 20$. Thus, with $\phi_{\max}(\mathbf{\Sigma}_u)$ uniformly bounded in delay d, $\phi_{\max}(\mathbf{\Sigma}_y^\diamond)$ is finite but increases with increasing d (as well as increasing $\tilde{d}$). For $\tilde{d} = 0.01, 0.35, 0.49$, we obtain $\phi_{\min}(\mathbf{\Sigma}_r) = 0.987, 0.629, 0.523$, respectively for $d = 5$, $\phi_{\min}(\mathbf{\Sigma}_r) = 0.986, 0.620, 0.512$, respectively for $d = 10$, and $\phi_{\min}(\mathbf{\Sigma}_r) = 0.986, 0.617, 0.508$, respectively for $d = 20$. Note that $1/4^{\tilde{d}} = 0.986, 0.616, 0.507$ for $\tilde{d} = 0.01, 0.35, 0.49$, respectively. Thus, $\phi_{\min}(\mathbf{\Sigma}_r)$ values for various delays agree with the lower bound in (74) quite well. □

Let $\hat{\mathbf{\Omega}}_{y\lambda} = \arg\min_{\mathbf{\Omega}_y \succ \boldsymbol{0}} \bar{\mathcal{L}}(\mathbf{\Omega}_y)$. Theorem 1 establishes local consistency of $\hat{\mathbf{\Omega}}_{y\lambda}$. Its proof is given in Appendix B.

Theorem 1 (Local Consistency): For $\tau > 2$, $m = d+1$, $\bar{n} = n-d$, and δ_{*n}, N_1 and $N_{\delta_*,\gamma}$ as in Lemma 2, define

$$R = 8(1+m)/\beta_{\min}^2\,, \tag{81}$$

$$r_n = \delta_{*n}\sqrt{mp_n + m^2 s_n^\diamond} = o(1)\,, \tag{82}$$

$$N_2 = \arg\min\left\{\bar{n} : r_n \leq \frac{0.1}{R\beta_{\min}}\right\}, \tag{83}$$

$$N_3 = \arg\min\left\{\bar{n} : r_n \leq \frac{\epsilon}{R}\right\}, \tag{84}$$

$$N_4 = \arg\min\left\{\bar{n} : \lambda_n \leq \frac{\min_{(i,j):\,[\mathbf{\Omega}_y^\diamond]_{ij}\neq 0} |[\mathbf{\Omega}_y^\diamond]_{ij}|}{a+1}\right\}, \tag{85}$$

$$N_5 = \arg\min\left\{\bar{n} : \lambda_n \leq \frac{\min_{(i,j):\,[\mathbf{\Omega}_y^\diamond]_{ij}\neq 0} |[\mathbf{\Omega}_y^\diamond]_{ij}|}{b+1}\right\}, \tag{86}$$

$$\lambda_{n\ell} = 2\delta_{*n}\,, \tag{87}$$

$$\lambda_{nu1} = \frac{m+1}{m\sqrt{s_n^\diamond}} r_n = (m+1)\sqrt{1+p_n/(m^2 s_n^\diamond)}\,\delta_{*n}\,, \tag{88}$$

$$\lambda_{nu2} = \max\left(Rr_n, \lambda_{nu1}\right)\,. \tag{89}$$

Under assumptions (A1)-(A3), there exists a local minimizer $\hat{\mathbf{\Omega}}_{y\lambda}$ of $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ satisfying

$$\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F \le Rr_n \tag{90}$$

with probability greater than $1 - 1/p_n^{\tau-2}$ if

(i) for the lasso penalty $\rho_\lambda(t) = \lambda|t|$, sample size $\bar{n} > \max\{N_{\delta_*,\gamma}, N_2\}$ and the regularization parameter satisfies $\lambda_{n\ell} \le \lambda_n \le \lambda_{nu1}$,

(ii) for the SCAD penalty $\rho_\lambda(t)$, sample size $\bar{n} > \max\{N_{\delta_*,\gamma}, N_2, N_4\}$ and the regularization parameter satisfies $\lambda_n = \lambda_{nu2}$,

(iii) for the log-sum penalty $\rho_\lambda(t)$, sample size $\bar{n} > \max\{N_{\delta_*,\gamma}, N_2, N_3\}$ and the regularization parameter satisfies $\lambda_{n\ell} \le \lambda_n \le \lambda_{nu1}$,

(iv) for the MCP penalty $\rho_\lambda(t)$, sample size $\bar{n} > \max\{N_{\delta_*,\gamma}, N_2, N_5\}$ and the regularization parameter satisfies $\lambda_n = \lambda_{nu2}$.

For the lasso penalty, $\hat{\mathbf{\Omega}}_{y\lambda}$ is a global minimizer whereas for the other three penalties, it is a local minimizer. •

Remark 6: Convergence Rate. Theorem 1 helps in choosing $m = d+1$ so that for given p_n, $\lim_{n\to\infty} \|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F = 0$. In terms of the rate of convergence, $\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F = \mathcal{O}_P\left(mr_n/\beta_{\min}^2\right) = \mathcal{O}_P(e_n/\beta_{\min}^2)$ where $e_n := m\delta_{*,n}\sqrt{mp_n + m^2 s_n^\diamond}$ ($mp_n + m^2 s_n^\diamond$ is the maximum number of nonzero elements in $\mathbf{\Omega}_y^\diamond$) and $\beta_{\min}$ is defined in assumption (A3). Therefore, for $\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F \to 0$ as $n \to \infty$, we must have $e_n \to 0$, assuming $\beta_{\min}$ is unchanged with n. For $\gamma > 3/4$, by Lemma 2(a), $e_n^2 = c_a^4(p_n) m^4 (p_n m^{-1} + s_n^\diamond)\ln(m^2 p_n^\tau)/n$. Suppose that for some positive constants b_1, b_2, θ_1 and θ_2, we have $m = b_1 n^{\theta_1}$ and $p_n + s_n^\diamond = b_2 n^{\theta_2}$. Ignoring $c_a(p_n)$, we have $\lim_{n\to\infty} e_n = 0$ if $1 - 4\theta_1 - \theta_2 > 0$. If p_n and $s_n^\diamond$ do not change with n (i.e., $\theta_2 = 0$), then $\theta_1 < 1/4$, i.e., $d = m - 1 = \mathcal{O}(n^{1/4})$. If $\theta_2 \le 1/3$ so that $p_n + s_n^\diamond = \mathcal{O}(n^{1/3})$, then $\theta_1 < 1/6$, i.e., d should increase much more slowly with sample size n if the graph size p_n also increases with n. □

Remark 7: Long Range Dependence. Here we examine Theorem 1 in terms of differences between SRD and LRD time series. By Theorem 1, the Frobenius error norm $\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F = \mathcal{O}_P(\delta_{*n}/\beta_{\min}^2)$ where $\delta_{*n} \propto 1/n^{0.5}$ (cf. Lemma 2(a)) for SRD sequences and for LRD sequences with $\gamma > 0.75$, and $\delta_{*n} \propto 1/n^{2\gamma-1}$ (cf. Lemma 2(c)) for LRD sequences when $\gamma \in (0.5, 0.75)$, with all other parameters remaining the same in R and r_n in (81) and (82), respectively. As discussed in Remark 5, $\beta_{\min}$ decreases with increasing $\tilde{d}$ (recall that $\gamma = 1 - \tilde{d}$; see Sec. III-B2). Consider three series with identical CIGs: a vector AR(3) series as in Sec. VI-A1, and ARFIMA(3,0.15,0) and ARFIMA(3,0.35,0) series as in Sec. VI-B which are generated via the aforementioned AR(3) series. The AR(3) series is SRD, the ARFIMA(3,0.15,0) series is LRD with $\gamma = 0.85$ and the ARFIMA(3,0.35,0) series is LRD with $\gamma = 0.65$. Let $\beta_{\min} = \beta_0, \beta_{0.15}$ and $\beta_{0.35}$ for the AR(3), ARFIMA(3,0.15,0) and ARFIMA(3,0.35,0) series, respectively. Then $\beta_0 > \beta_{0.15} > \beta_{0.35}$. The Frobenius error norms $e_0, e_{0.15}$ and $e_{0.35}$ for the AR(3), ARFIMA(3,0.15,0) and ARFIMA(3,0.35,0) series, respectively, are $\mathcal{O}_P(1/(\beta_0 n^{0.5}))$, $\mathcal{O}_P(1/(\beta_{0.15} n^{0.5}))$ and $\mathcal{O}_P(1/(\beta_{0.35} n^{0.3}))$, respectively, implying $e_0 < e_{0.15} < e_{0.35}$. We revisit this discussion later in Sec. VI-B. □

We follow the proof technique of [45, Lemma 6] in establishing Lemma 3, as in [43, Sec. V].

Lemma 3 (Local Convexity): The optimization problem

$$\hat{\mathbf{\Omega}}_{y\lambda} = \arg\min_{\mathbf{\Omega}_y \in \mathcal{B}} \bar{\mathcal{L}}(\mathbf{\Omega}_y)\,, \tag{91}$$

$$\mathcal{B} = \{\mathbf{\Omega}_y : \mathbf{\Omega}_y \succ \mathbf{0},\ \|\mathbf{\Omega}_y\| \le 0.99\,\theta\}\,, \tag{92}$$

$$\theta = \begin{cases} \infty & : \text{ Lasso} \\ \sqrt{(a-1)/m} & : \text{ SCAD} \\ \sqrt{\epsilon/(m\lambda_n)} & : \text{ log-sum} \\ \sqrt{b/m} & : \text{ MCP}\,, \end{cases} \tag{93}$$

consists of a strictly convex objective function over a convex constraint set, for all four penalties, where λ_n is as defined in Theorem 1. •

Lemma 3 and Theorem 1 lead to Theorem 2.

Theorem 2: Assume the conditions of Theorem 1. Then $\hat{\mathbf{\Omega}}_{y\lambda}$ as defined in Theorem 1 is unique, satisfying $\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F \le Rr_n$ with probability greater than $1 - 1/p_n^{\tau-2}$ if $Rr_n + 1/\beta_{\min} \le 0.99\,\theta$ where θ is as in Lemma 3. •

Remark 8: We see from Theorem 1 that as $n \to \infty$, $\lambda_n \to 0$ (since $r_n = o(1)$), therefore, we eventually have "global" convexity for log-sum penalty by (93) for any $\mathbf{\Omega}_y^\diamond$. But such is not the case for SCAD or MCP where one may need a (or b, respectively) to become large in which case they both would behave more like lasso. □

We now turn to graph recovery. Define

$$\hat{\mathcal{E}} = \left\{\{q,\ell\} : \|\hat{\mathbf{\Omega}}_{y\lambda}^{(q\ell)}\|_F > \gamma_n > 0, q \ne \ell\right\}\,, \tag{94}$$

$$\mathcal{E}^\diamond = \left\{\{q,\ell\} : \|(\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F > 0, q \ne \ell\right\}\,, \tag{95}$$

$$\bar{\sigma}_n = Rr_n\,, \tag{96}$$

$$\nu_t = \min_{\{q,\ell\}\in\mathcal{E}^\diamond} \|(\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F\,, \tag{97}$$

$$N_6 = \arg\min\left\{n : \bar{\sigma}_n \le 0.4\nu_t\right\}, \tag{98}$$

where R and r_n are as in (81) and (82), respectively.

Theorem 3: For $\gamma_n = 0.5\nu_t$ and $n \ge N_6$, $\hat{\mathcal{E}} = \mathcal{E}^\diamond$ with probability greater than $1 - 1/p_n^{\tau-2}$ under the conditions of Theorem 1. •

VI. SYNTHETIC DATA EXAMPLES

We now present numerical results using synthetic data to illustrate the proposed approaches and compare them with several existing approaches. In synthetic data examples the ground truth is known which allows for assessment of the efficacy of various approaches to graph learning.

A. SHORT-RANGE DEPENDENCE MODELS

1) GRAPHS WITH 120 NODES

We consider two models for data generation with $p = 120$.

MA model. The time series data $\{\boldsymbol{x}(t)\}$ is generated using a vector moving average (MA) model of order 3 (MA(3)):

$$\boldsymbol{x}(t) = 0.5\boldsymbol{I}_p\boldsymbol{w}(t) + \sum_{i=1}^{3}(\boldsymbol{B}_i/i)\boldsymbol{w}(t-i)\,, \quad \boldsymbol{x}(t) \in \mathbb{R}^p\,, \quad (99)$$

where $\boldsymbol{w}(t)$ is i.i.d. zero-mean Gaussian with precision matrix $\boldsymbol{\Omega}$ and $\boldsymbol{B}_i$ is block-diagonal with 15×15 sub-blocks $\boldsymbol{B}_i^{(q)}$, $q \in [8]$. The diagonal entries of $\boldsymbol{\Omega}$ are set to 0.5, and the off-diagonal entries follow an Erdös-Rènyi (ER) graph with connection probability $p_{er} = 0.001$: if nodes j and k are not connected in the ER graph, we have $[\boldsymbol{\Omega}]_{jk} = 0$, and if they are connected, then $[\boldsymbol{\Omega}]_{jk}$ is uniformly distributed over $[-0.4, -0.1] \cup [0.1, 0.4]$. Only 20% of entries of $\boldsymbol{B}_i^{(q)}$'s are nonzero (randomly picked) and the nonzero elements are independently and uniformly distributed over $[-0.4, -0.2] \cup [0.2, 0.4]$.

AR model. Here the time series data $\{\boldsymbol{x}(t)\}$ is generated using a vector autoregressive (AR) model of order 3 (VAR(3)):

$$\boldsymbol{x}(t) = \sum_{i=1}^{3}\boldsymbol{A}_i\boldsymbol{x}(t-i) + \boldsymbol{w}(t)\,, \quad \boldsymbol{x}(t) \in \mathbb{R}^p\,, \quad (100)$$

where $\boldsymbol{w}(t)$ is i.i.d. zero-mean Gaussian with precision matrix $\boldsymbol{\Omega} = 0.5\boldsymbol{I}_p$ and $\boldsymbol{A}_i$ is block-diagonal with 15×15 sub-blocks $\boldsymbol{A}_i^{(q)}$, $q \in [8]$. Only 10% of entries of $\boldsymbol{A}_i^{(q)}$'s are nonzero (randomly picked) and the nonzero elements are independently and uniformly distributed over $[-0.8, -0.3] \cup [0.3, 0.8]$. We then check if the VAR(3) model is stable with all eigenvalues of the companion matrix ≤ 0.95 in magnitude; if not, then we scale $\boldsymbol{A}_i$'s to fulfill this condition (see [11, Sec. VI.A]).

For both models, the first 100 samples are discarded to eliminate transients, and we generate n observations if $\boldsymbol{x}(t)$ with $n \in \{256, 1024, 2048\}$. In each run, we calculate the true PSD $\boldsymbol{S}_x^{\diamond}(f)$ for $f \in [0, 0.5]$ at intervals of 0.01, and then take $\{i, j\} \in \mathcal{E}^{\diamond}$ (the true edge-set) if $g_{ij} > \bar{\tau}\, g_{\max}$, else $\{i, j\} \notin \mathcal{E}^{\diamond}$, where $g_{\max} = \max_{i,j\in[p]}(1/F)\sum_f |[(\boldsymbol{S}_x^{\diamond}(f))^{-1}]_{ij}|$, $g_{ij} = (1/F)\sum_f |[(\boldsymbol{S}_x^{\diamond}(f))^{-1}]_{ij}|$, F is number of frequencies points in $[0, 0.5]$ at intervals of 0.01, and the threshold $\bar{\tau} = 0.01$ for both MA and AR models. To avoid very "peaky" inverse PSDs, if $g_{\max} > 5,000$ we redraw the samples till this condition is satisfied: it is needed for the MA models where we do not control the roots of $\det(\mathcal{B}(z)) = 0$ ($\mathcal{B}(z)$ is defined in Sec. III-B1). We need $\det(\mathcal{B}(z)) \neq 0$ for any $|z| = 1$ for $\boldsymbol{S}_x^{-1}(f)$ to exist for every f. In practical terms, we should have $\det(\mathcal{B}(z)) \neq 0$ for $\{z : 1 - \delta \leq |z| \leq 1 + \delta,\ \delta > 0\}$. This set-up leads to connected edges in the true graphs in the range [4.75%,12.44%] and [8.53%,9.94%] for MA and AR models, respectively, based on 100 runs.

Simulation results based on 100 runs are shown in Table 1 where the performance measures are F_1-score and Hamming distance (between estimated and true edge-sets $\hat{\mathcal{E}}$ and $\mathcal{E}^{\diamond}$, respectively) for efficacy in edge detection, and timing per run as a surrogate for computational complexity. The F_1-score is defined as $F_1 = 2 \times$ precision $\times$ recall/(precision + recall) where precision $= |\hat{\mathcal{E}} \cap \mathcal{E}^{\diamond}|/|\hat{\mathcal{E}}|$, recall $= |\hat{\mathcal{E}} \cap \mathcal{E}^{\diamond}|/|\mathcal{E}^{\diamond}|$, and the Hamming distance is between $\hat{\mathcal{E}}$ and $\mathcal{E}^{\diamond}$, scaled by 0.5 to count only distinct edges. The ideal F_1 score is 1 and the ideal Hamming distance is 0. All algorithms were run on a Window 11 Enterprise operating system with processor Intel(R) Core(TM) i7-10700 CPU @2.90 GHz with 32 GB RAM, using MATLAB R2023a. We compare eight approaches:

(i) "IID:lasso": a time-domain approach assuming the data is i.i.d. and therefore exploits only the sample covariance $\hat{\boldsymbol{\Sigma}}_x = (1/n)\sum_{t=1}^{n}\boldsymbol{x}(t)\boldsymbol{x}^{\top}(t)$ and the resulting graph is based on the precision matrix $\boldsymbol{\Omega}_x$ of $\{\boldsymbol{x}(t)\}$. We minimize $-\ln(|\boldsymbol{\Omega}_x|) + tr\left(\hat{\boldsymbol{\Sigma}}_x\boldsymbol{\Omega}_x\right)$ w.r.t. $\boldsymbol{\Omega}_x$ with the lasso penalty, using the ADMM approach of [59, Sec. 6.5]. If the resulting estimate is denoted by $\hat{\boldsymbol{\Omega}}_x$, the estimated edge-set is $\hat{\mathcal{E}} = \{\{k, \ell\} : |[\hat{\boldsymbol{\Omega}}_x]_{k\ell}| > 0,\ k, \ell \in [p]\}$ (cf. (62)).

(ii) "IID:log-sum": a time-domain approach assuming the data is i.i.d. and it is as for IID:lasso except the lasso penalty is replaced with the log-sum penalty ($\epsilon = 0.0001$) and we use LLA with ADMM for optimization.

(iii) "DE:lasso": proposed delay embedding (DE) approach with lasso penalty using ADMM.

(iv) "DE:log-sum": proposed DE approach with log-sum penalty ($\epsilon = 0.0001$) using LLA with ADMM.

(v) "DE:SCAD": proposed DE approach with SCAD penalty ($a = 3.7$) using LLA with ADMM.

(vi) "DE:MCP": proposed DE approach with MCP penalty ($b = 3.7$) using LLA with ADMM.

(vii) "FD:lasso": the frequency-domain (FD) approach of [10] with lasso penalty using ADMM.

(viii) "FD:log-sum": the FD approach of [11] as applied to single-attribute models, with log-sum penalty ($\epsilon = 0.0001$) using LLA with ADMM.

For the DE approaches we need to pick delay d which was $d = 3$ for both MA and AR models and for all sample sizes. For the FD approaches we need to pick number of frequency test points M in the interval $(0, 0.5)$ which were $M = 2, 3, 4$ for both MA and AR models, for sample sizes $n = 256, 1024, 2048$, respectively. All non-convex penalized approaches were based on LLA, initialized with the lasso results, therefore, the timing for non-convex penalties includes the timing for lasso as well. In approaches (iii)-(viii) we use sparse group penalties (cf. (46)) requiring choice of the pair (λ, α). In Table 1, λ's and α's were picked from a grid of values to maximize the F_1 score in each Monte Carlo run (ground truth is known in synthetic examples) – this establishes how well a method will perform if λ's and α's are judiciously picked. Instead of searching over a two-dimensional space (λ, α), we first search over a grid of λ values with fixed $\alpha = 0.05\,(= \alpha_0)$. Then with λ fixed at the selected value, we search over a grid of α values over

TABLE 1. F_1 scores, Hamming distances and timings (s) for the synthetic data examples ($p = 120$, MA model (99), AR model (100)), averaged over 100 runs (standard deviation σ in parentheses). "DE:lasso and "DE:log-sum" are proposed delay embedding approaches with lasso and log-sum penalties, respectively, "FD:lasso" and "FD:log-sum" are the frequency-domain approaches with lasso [10] and log-sum [11] penalties, respectively, and "IID:lasso" and "IID:log-sum" are time-domain approaches based on i.i.d. modeling of data (see also remark 9).

n	256	1024	2048	256	1024	2048
λ's and α's picked to maximize F_1 score						
	MA model: F_1 score (σ)			AR model: F_1 score (σ)		
IID:lasso	0.326 (0.036)	0.482 (0.044)	0.555 (0.053)	0.264 (0.035)	0.400 (0.055)	0.483 (0.055)
IID:log-sum	0.331 (0.035)	0.511 (0.045)	0.593 (0.054)	0.256 (0.034)	0.406 (0.060)	0.501 (0.062)
DE:lasso	0.492 (0.048)	0.730 (0.067)	0.791 (0.077)	0.517 (0.042)	0.695 (0.032)	0.763 (0.032)
DE:log-sum	0.587 (0.050)	0.861 (0.083)	0.903 (0.093)	0.621 (0.038)	0.808 (0.028)	0.873 (0.021)
DE:SCAD	0.489 (0.046)	0.744 (0.069)	0.811 (0.079)	0.495 (0.046)	0.687 (0.025)	0.762 (0.032)
DE:MCP	0.498 (0.049)	0.751 (0.068)	0.815 (0.081)	0.495 (0.037)	0.694 (0.022)	0.770 (0.030)
FD:lasso [10]	0.463 (0.044)	0.697 (0.059)	0.752 (0.070)	0.447 (0.032)	0.639 (0.028)	0.710 (0.023)
FD:log-sum [11]	0.515 (0.044)	0.824 (0.077)	0.883 (0.086)	0.482 (0.030)	0.716 (0.039)	0.782 (0.021)
	MA model: Hamming distance (σ)			AR model: Hamming distance (σ)		
IID:lasso	905 (160)	618 (84)	542 (41)	989 (192)	681 (113)	592 (86)
IID:log-sum	910 (198)	619 (66)	505 (30)	1027 (223)	680 (89)	562 (61)
DE:lasso	785 (169)	387 (49)	310 (76)	646 (108)	396 (56)	318 (51)
DE:log-sum	549 (78)	205 (80)	140 (113)	466 (44)	245 (35)	172 (26)
DE:SCAD	810 (142)	370 (54)	284 (75)	675 (107)	409 (51)	324 (52)
DE:MCP	740 (107)	355 (50)	280 (84)	695 (107)	401 (50)	313 (48)
FD:lasso	715 (108)	409 (39)	354 (67)	652 (106)	417 (41)	345 (33)
FD:log-sum	716 (145)	263 (77)	166 (98)	601 (62)	312 (28)	256 (29)
	MA model: Timing (s) (σ)			AR model: Timing (s) (σ)		
IID:lasso	0.069 (0.008)	0.068 (0.008)	0.078 (0.010)	0.088 (0.015)	0.079 (0.011)	0.078 (0.013)
IID:log-sum	0.297 (0.029)	0.231 (0.021)	0.225 (0.019)	0.244 (0.030)	0.166 (0.012)	0.148 (0.014)
DE:lasso	3.65 (0.51)	4.46 (0.37)	4.86 (0.79)	4.76 (0.79)	4.62 (0.76)	4.69 (0.82)
DE:log-sum	13.23 (1.28)	15.12 (1.30)	16.69 (2.51)	12.11 (1.59)	12.83 (1.38)	13.11 (1.60)
DE:SCAD	14.48 (2.16)	12.58 (0.83)	14.05 (1.48)	12.98 (2.11)	11.98 (2.24)	13.27 (2.24)
DE:MCP	13.28 (1.21)	12.34 (1.04)	14.27 (1.39)	13.48 (2.05)	12.21 (2.19)	13.38 (2.35)
FD:lasso	2.99 (1.12)	3.39 (0.48)	3.84 (0.64)	4.13 (1.09)	5.43 (1.14)	6.35 (1.59)
FD:log-sum	8.89 (1.24)	8.38 (0.56)	7.82 (0.56)	7.79 (0.92)	9.66 (2.24)	10.26 (2.77)
DE:log-sum, λ's and α's picked to minimize an information criterion						
	MA model: F_1 score (σ)			AR model: F_1 score (σ)		
BIC	0.535 (0.039)	0.853 (0.083)	0.892 (0.092)	0.643 (0.057)	0.824 (0.035)	0.876 (0.019)
eBIC, $\bar{\gamma}$=0.5	0.532 (0.041)	0.555 (0.047)	0.819 (0.089)	0.529 (0.031)	0.737 (0.059)	0.838 (0.043)
eBIC, $\bar{\gamma}$=1	0.519 (0.043)	0.548 (0.026)	0.638 (0.071)	0.522 (0.041)	0.637 (0.074)	0.787 (0.046)
	MA model: Hamming distance (σ)			AR model: Hamming distance (σ)		
BIC	513 (40)	224 (93)	163 (116)	372 (36)	219 (30)	169 (21)
eBIC, $\bar{\gamma}$=0.5	514 (41)	383 (43)	240 (86)	430 (24)	284 (48)	197 (39)
eBIC, $\bar{\gamma}$=1	520 (46)	489 (39)	413 (50)	434 (28)	358 (52)	243 (40)
	MA model: Timing (s) (σ)			AR model: Timing (s) (σ)		
BIC	11.73 (1.23)	15.79 (1.48)	15.58 (1.45)	13.28 (2.14)	12.07 (1.59)	13.43 (1.33)
eBIC, $\bar{\gamma}$=0.5	12.32 (1.41)	15.75 (1.68)	15.98 (1.52)	12.58 (2.04)	12.52 (2.15)	13.73 (1.43)
eBIC, $\bar{\gamma}$=1.0	13.26 (1.84)	14.68 (1.74)	14.54 (2.15)	12.97 (2.24)	13.17 (1.89)	13.29 (1.60)

[0, 1]. We used the grid $\{0.05, 0.1, 0.3, 0.5, 0.7, 0.9\}$ for α selection. We picked ten logarithmically spaced values in the interval $[\underline{\lambda}, \bar{\lambda}]$ for λ selection with $(\underline{\lambda}, \bar{\lambda}) = (0.01, 0.5)$ for lasso, SCAD and MCP penalties, and $(\epsilon\underline{\lambda}, \epsilon\bar{\lambda}) = (0.001, 0.1)$ for the log-sum penalty.

In Table 1 we also show results for log-sum penalty (DE:log-sum) when λ's and α's are picked based on data alone, using the BIC or eBIC method discussed in Sec. IV-D (BIC is based on [43, Sec. IV-A] and [42, Sec. III-B], and eBIC is based on [62]). For eBIC we picked $\bar{\gamma}$ in (64) as either 0.5 or 1 (recommended values in [62]).

It is seen from Table 1 that log-sum-penalized approaches (FD or DE) significantly outperform lasso based DE and FD approaches, respectively, which in turn, significantly outperform the i.i.d. modeling based methods that ignore time-dependence, yielding higher F_1 scores and lower Hamming distances. The performance of the DE approaches is slightly to significantly better than the FD approaches when comparing DE:lasso with FD:lasso, and DE:log-sum with FD:log-sum, with F_1 scores and Hamming distances as performance metrics. For instance, for $n = 256$, for the proposed DE:log-sum we have an F_1 score of 0.587 and Hamming distance of 549 for the MA model, whereas FD:log-sum yields an F_1 score of 0.515 and Hamming distance of 716. The corresponding F_1 score and Hamming distance for the AR model for $n = 256$ are 0.621 and 466, respectively, for DE:log-sum, and 0.482 and 601, respectively, for FD:log-sum. It is also seen that the proposed DE:SCAD and DE:MCP approaches do not offer as much performance improvements over DE:lasso as offered by DE:log-sum. Finally, it is also seen from Table 1 that for (λ, α) selection based on data alone,

the performance of the eBIC method is worse than that of the BIC method.

Remark 9: Given the sample covariance $\hat{\mathbf{\Sigma}}_x = (1/n)\sum_{t=1}^{n} \boldsymbol{x}(t)\boldsymbol{x}^\top(t)$, the IID:lasso approach considered in Table 1 minimizes

$$-\ln(|\mathbf{\Omega}_x|) + tr\left(\hat{\mathbf{\Sigma}}_x \mathbf{\Omega}_x\right) + \sum_{i\neq j} |[\mathbf{\Omega}_x]_{ij}| \tag{101}$$

w.r.t. $\mathbf{\Omega}_x$ using the ADMM approach of [59, Sec. 6.5], which is also the approach in the Algorithm 1 with $d = 0$ and $\alpha = 1$ (requires modifications to step 6 of Algorithm 1). The resulting graph is based on the estimated $\mathbf{\Omega}_x$. This cost function has been used in [37], [38], and [39], but optimized using different approaches. In [40] a constrained ℓ_1-minimization (CLIME) approach is proposed to estimate $\mathbf{\Omega}_x$ under the assumption of i.i.d. data, and the resulting graph is based on the estimated $\mathbf{\Omega}_x$. A software package called "huge" is available in [41] for precision matrix based graph estimation where several approaches including graphical lasso of [39], have been implemented for i.i.d. data. The CLIME approach [40] exhibited "numerical instability" in the simulation examples considered in [41]. In [46] the approaches of [38] and [40] to estimate $\mathbf{\Omega}_x$ are analyzed and compared via simulations when the data $\{\boldsymbol{x}(t)\}$ is time-dependent. It is concluded in [46] that the performances of the approaches of [38] and [40] are similar in terms of the matrix estimation error norms in estimating $\mathbf{\Omega}_x$. Note that $\mathbf{\Omega}_x$ does not yield the underlying time-series graph for dependent data. The lasso penalty in (101) is replaced with non-convex penalties such as SCAD or LSP, in [34], [43], and [45], and one minimizes

$$-\ln(|\mathbf{\Omega}_x|) + tr\left(\hat{\mathbf{\Sigma}}_x \mathbf{\Omega}_x\right) + \sum_{i\neq j} \rho_\lambda\left(|[\mathbf{\Omega}_x]_{ij}|\right) \tag{102}$$

where $\rho_\lambda(u)$, $u \in \mathcal{R}$, is as in Sec. IV-A. Note that [43] considers the multi-attribute case with i.i.d. data, and it applies here after setting the number of attributes to one. The IID:log-sum approach considered in Table 1 is the approach in the Algorithm 2 with $d = 0$ and $\alpha = 1$ (requires modifications to step 6 of Algorithm 1). We note that non-convex penalties are not considered in [37], [38], [39], and [40]. □

2) VARYING EMBEDDING TIME-DELAYS, 120 NODES

Table 2 shows the performance results for the MA model (99) and the AR model (100) using the proposed DE:log-sum approach with varying embedding time-delay $d \in \{1, 2, 3, 4, 5\}$. The number of unknown parameters of $\mathbf{\Omega}_y$ being estimated are $\mathcal{O}(((d+1)p)^2)$. As noted in Remark 1, we are using $\mathbf{\Omega}_y^{(ij)} = \mathbf{0}$, $\boldsymbol{y}(t) \in \mathbb{R}^{(d+1)p}$, as surrogate for $[\boldsymbol{S}_x^{-1}(f)]_{ij} = 0\ \forall f \in [0, 1]$ with the approximation improving with increasing d. The simulation results are shown in Table 2. It is seen from Table 2 that as d increases, the performance metrics the F_1 score and Hamming distance improve significantly at first, but then they plateau out with increasing d (compare $d = 3$ results with $d = 4, 5$ results). Note that the number of unknown parameters increase as $\mathcal{O}((d+1)^2)$, which also accounts for significant increase in computational time with increasing d.

TABLE 2. MA model (99) and AR model (100), $p = 120$: F_1 scores, Hamming distances and timings using DE:log-sum approach, averaged over 100 runs, for varying time-delays d.

n	256	1024	2048
MA model (99)			
F_1 score (σ): λ's and α's picked to maximize F_1			
d=1	0.551 (0.051)	0.788 (0.071)	0.846 (0.081)
d=2	0.586 (0.052)	0.848 (0.082)	0.893 (0.090)
d=3	0.587 (0.050)	0.861 (0.083)	0.903 (0.093)
d=4	0.589 (0.050)	0.863 (0.081)	0.903 (0.093)
d=5	0.581 (0.049)	0.860 (0.080)	0.900 (0.092)
Hamming distance (σ): λ's and α's picked to maximize F_1			
d=1	636 (60)	301 (57)	219 (85)
d=2	596 (79)	224 (79)	155 (108)
d=3	549 (78)	205 (80)	140 (113)
d=4	553 (70)	200 (78)	139 (112)
d=5	543 (53)	202 (77)	143 (110)
Timing (s) (σ): λ's and α's picked to maximize F_1			
d=1	4.75 (0.34)	4.19 (0.43)	4.43 (0.46)
d=2	7.94 (0.54)	8.13 (0.62)	8.57 (0.62)
d=3	13.23 (1.28)	15.12 (1.30)	16.69 (2.51)
d=4	21.29 (1.46)	23.50 (1.81)	24.32 (3.27)
d=5	27.89 (3.32)	31.34 (1.75)	33.16 (2.85)
AR model (100)			
F_1 score (σ): λ's and α's picked to maximize F_1			
d=1	0.497 (0.040)	0.666 (0.051)	0.741 (0.041)
d=2	0.581 (0.031)	0.759 (0.034)	0.824 (0.023)
d=3	0.621 (0.038)	0.808 (0.028)	0.873 (0.021)
d=4	0.631 (0.049)	0.817 (0.037)	0.877 (0.022)
d=5	0.634 (0.052)	0.818 (0.038)	0.877 (0.023)
Hamming distance (σ): λ's and α's picked to maximize F_1			
d=1	574 (37)	392 (38)	320 (34)
d=2	502 (31)	307 (28)	236 (23)
d=3	466 (44)	245 (35)	172 (26)
d=4	437 (66)	231 (45)	164 (26)
d=5	420 (73)	223 (43)	160 (26)
Timing (s) (σ): λ's and α's picked to maximize F_1			
d=1	3.65 (0.40)	3.45 (0.19)	3.31 (0.45)
d=2	7.13 (0.93)	6.97 (0.46)	6.89 (0.79)
d=3	12.11 (1.59)	12.83 (1.38)	13.11 (1.60)
d=4	19.48 (3.03)	19.44 (2.78)	19.87 (2.98)
d=5	27.18 (4.37)	28.02 (5.10)	28.12 (4.41)

As noted earlier in Remark 2 in Sec. III-A, this observation suggests the heuristic that the user should try several values of d and pick the smallest value, say d_0, beyond which (e.g., for $d = d_0 + 1$) the estimated graph is only marginally different from the estimated graph for $d = d_0$. In practice, since the true graph is unknown, one cannot compute the F_1 score or the Hamming distance. For real data, what constitutes marginal difference would be problem and user dependent. We revisit this later in the context of financial time series based real data results presented in Sec. VII.

3) VARYING ORDER AND SPARSITY OF AR MODELS, 120 NODES

Varying Order. In Table 3 we display some numerical ablation results by varying the AR model order. We use a VAR(L)

TABLE 3. Varying AR model order: VAR(L) as in (103), $L \in \{1, 2, 3, 4\}$, $p = 120$. F_1 scores, Hamming distances and timings using DE:log-sum with $d = 3$, averaged over 100 runs.

n	256	1024	2048
F_1 score (σ): λ's and α's picked to maximize F_1			
L=1	0.764 (0.028)	0.866 (0.017)	0.892 (0.014)
L=2	0.676 (0.048)	0.823 (0.022)	0.852 (0.019)
L=3	0.621 (0.038)	0.808 (0.028)	0.873 (0.021)
L=4	0.589 (0.040)	0.796 (0.034)	0.865 (0.030)
Hamming distance (σ): λ's and α's picked to maximize F_1			
L=1	104 (16)	64 (10)	56 (10)
L=2	286 (51)	170 (29)	148 (15)
L=3	466 (44)	245 (35)	172 (26)
L=4	559 (29)	291 (33)	198 (36)
Timing (s) (σ): λ's and α's picked to maximize F_1			
L=1	16.09 (2.24)	14.92 (2.63)	12.86 (3.05)
L=2	14.50 (2.58)	14.22 (2.68)	13.34 (3.06)
L=3	12.11 (1.59)	12.83 (1.38)	13.11 (1.60)
L=4	12.28 (1.53)	12.57 (0.98)	12.78 (1.15)

model with $L \in \{1, 2, 3, 4\}$, given by

$$\boldsymbol{x}(t) = \sum_{i=1}^{L} \boldsymbol{A}_i \boldsymbol{x}(t-i) + \boldsymbol{w}(t), \quad \boldsymbol{x}(t) \in \mathbb{R}^p, \tag{103}$$

where $\boldsymbol{w}(t)$ is i.i.d. zero-mean Gaussian with precision matrix $\boldsymbol{\Omega} = 0.5\boldsymbol{I}_p$ and $\boldsymbol{A}_i$ is block-diagonal with 15×15 sub-blocks $\boldsymbol{A}_i^{(q)}$, $q \in [8]$, as for (100), and $p = 120$. Only 10% of entries of $\boldsymbol{A}_i^{(q)}$'s are nonzero (randomly picked) and the nonzero elements are independently and uniformly distributed over $[-0.8, -0.3] \cup [0.3, 0.8]$. The graph sparsity varies with varying L even though sparsity of $\boldsymbol{A}_i^{(q)}$'s remains unchanged. Based on 100 Monte Carlo runs, the number of edges in the resulting true graphs are in the ranges [2.90%, 4.47%], [5.94%, 8.01%], [8.53%, 9.94%] and [9.43%, 11.04%] for $L = 1, 2, 3$ and 4, respectively. We used the log-sum penalty and $d = 3$. The simulation results are shown in Table 3. It is seen that the results are consistent across model orders, both in terms of the F_1 scores and the Hamming distances.

Varying Sparsity, AR model (100). Here we use VAR(3) model as in (100) (or in (103) with $L = 3$), and vary the percentage of nonzero entries in $\boldsymbol{A}_i^{(q)}$'s. By varying these percentages as 2%, 5% and 10%, we obtain the number of edges in the resulting true graphs in the ranges [1.26%, 2.46%], [4.19%, 6.29%] and [8.53%, 9.94%], respectively (based on 100 Monte Carlo runs). The simulation results are shown in Table 4. It is seen that the results are consistent across various sparsity levels in terms of the F_1 scores. Higher sparsity levels implies more edges in the graph, and the larger Hamming distances for higher sparsity reflects more errors in graph recovery for the same relative errors measured by the F_1 score. By Theorem 1, the estimation error norm $\|\hat{\boldsymbol{\Omega}}_{y\lambda} - \boldsymbol{\Omega}_y^{\diamond}\|_F = \mathcal{O}(\sqrt{mp_n + m^2 s_n^{\diamond}})$ where $s_n^{\diamond}$ is the number of (off-diagonal) edges in the graph and $m = d + 1$.

4) AR MODELS, VARYING NUMBER OF NODES

We now consider AR(3) models for time-dependent data generation with varying number of graph nodes $p \in \{60, 120, 240\}$. The objective is to empirically study the performance stability/scalability of the proposed solutions with varying model dimensions. The AR(3) model

TABLE 4. Varying sparsity of AR(3) model (100), p=120: F_1 scores, Hamming distances and timings using DE:log-sum with $d = 3$, averaged over 100 runs.

n	256	1024	2048
% edges	F_1 score (σ): λ's, α's maximize F_1		
[1.26%, 2.46%]	0.779 (0.030)	0.852 (0.021)	0.861 (0.021)
[4.19%, 6.29%]	0.650 (0.052)	0.793 (0.018)	0.824 (0.015)
[8.53%, 9.94%]	0.621 (0.038)	0.808 (0.028)	0.873 (0.021)
% edges	Hamming distance (σ): λ's, α's maximize F_1		
[1.26%, 2.46%]	53 (11)	40 (9)	41 (10)
[4.19%, 6.29%]	220 (50)	148 (18)	142 (15)
[8.53%, 9.94%]	466 (44)	245 (35)	172 (26)
% edges	Timing (s) (σ): λ's, α's maximize F_1		
[1.26%, 2.46%]	11.02 (1.92)	9.04 (1.46)	7.98 (1.43)
[4.19%, 6.29%]	11.68 (1.94)	12.40 (2.53)	13.34 (1.53)
[8.53%, 9.94%]	12.11 (1.59)	12.83 (1.38)	13.11 (1.60)

TABLE 5. F_1 scores, Hamming distances and timings (s) for AR(3) models, varying number of nodes p=60, 120 or 240, using DE:log-sum with $d = 3$, averaged over 100 runs (standard deviation σ in parentheses).

n	256	1024	2048
F_1 score (σ): λ's and α's picked to maximize F_1			
p=60	0.713 (0.047)	0.845 (0.029)	0.880 (0.024)
p=120	0.621 (0.038)	0.808 (0.028)	0.873 (0.021)
p=240	0.524 (0.034)	0.767 (0.030)	0.856 (0.018)
Hamming distance (σ): λ's and α's picked to maximize F_1			
p=60	101 (18)	58 (19)	47 (8)
p=120	466 (44)	245 (35)	172 (26)
p=240	2622 (67)	1388 (106)	889 (81)
Timing (s) (σ): λ's and α's picked to maximize F_1			
p=60	2.90 (0.31)	2.85 (0.40)	2.96 (0.41)
p=120	12.11 (1.59)	12.83 (1.38)	13.11 (1.60)
p=240	58.93 (6.55)	58.09 (8.36)	60.75 (7.56)

TABLE 6. F_1 scores, Hamming distances and timings (s) for non-Gaussian AR(3) model (100) ($p = 120$), averaged over 100 runs (standard deviation σ in parentheses). "DE: lasso and "DE: log-sum" are proposed delay embedding approaches with lasso and log-sum penalties, respectively, using $d = 3$.

n	256	1024	2048
Laplace $\boldsymbol{w}(t)$; λ's and α's picked to maximize F_1 score			
F_1 score (σ)			
DE:lasso	0.515 (0.052)	0.698 (0.035)	0.758 (0.038)
DE:log-sum	0.621 (0.046)	0.811 (0.027)	0.868 (0.023)
Hamming distance (σ)			
DE:lasso	658 (131)	392 (60)	324 (56)
DE:log-sum	468 (47)	243 (34)	178 (24)
Timing (s) (σ)			
DE:lasso	5.36 (1.03)	5.38 (1.06)	5.13 (0.72)
DE:log-sum	13.94 (2.14)	15.07 (2.46)	15.14 (2.59)
Exponential $\boldsymbol{w}(t)$; λ's and α's picked to maximize F_1 score			
F_1 score (σ)			
DE:lasso	0.514 (0.053)	0.697 (0.033)	0.760 (0.037)
DE:log-sum	0.619 (0.048)	0.812 (0.027)	0.867 (0.023)
Hamming distance (σ)			
DE:lasso	657 (125)	392 (56)	322 (54)
DE:log-sum	466 (52)	243 (33)	180 (25)
Timing (s) (σ)			
DE:lasso	5.57 (0.94)	5.66 (1.13)	5.18 (0.75)
DE:log-sum	14.26 (1.77)	15.67 (1.82)	14.73 (1.74)

follows (100) where $\boldsymbol{A}_i \in \mathbb{R}^{p\times p}$, $i \in [3]$, is block-diagonal with

(i) six 10×10 sub-blocks $\boldsymbol{A}_i^{(q)}$, $q \in [6]$, when $p = 60$,
(ii) eight 15×15 sub-blocks $\boldsymbol{A}_i^{(q)}$, $q \in [8]$, when $p = 120$ (as in Sec. VI-A1),
(iii) eight 30×30 sub-blocks $\boldsymbol{A}_i^{(q)}$, $q \in [8]$, when $p = 240$.

All other details regarding generation of $\boldsymbol{\Omega}$, $\boldsymbol{A}_i^{(q)}$'s and $\{\boldsymbol{x}(t)\}$ are exactly as before in Sec. VI-A1. For the given choices, the percentage of distinct connected edges in the true graphs do not change much with changing p: in 100 Monte Carlo runs, the number of edges in the resulting true graphs are in the ranges [9.04%, 12.77%], [8.53%, 9.94%] and [10.22%, 11.35%] for $p = 60$, 120 and 240, respectively. Since the total number of edges in a p-node graph would be $p(p-1)/2$, 10% edges result in 123, 714 and 2868 edges in graphs with p =60, 120 and 240 nodes, respectively.

We used the log-sum penalty and $d = 3$. The simulation results are shown in Table 5. Recall that the computational complexity is $\mathcal{O}((d+1)^3p^3)$, the number of unknowns in $\boldsymbol{\Omega}_y$ is $\mathcal{O}((d+1)^2p^2)$ and the estimation error norm $\|\hat{\boldsymbol{\Omega}}_{y\lambda} - \boldsymbol{\Omega}_y^\diamond\|_F = \mathcal{O}(\sqrt{(d+1)p_n + (d+1)^2 s_n^\diamond})$ where $s_n^\diamond$ is the number of edges in the graph. As p increases, so does $s_n^\diamond$ in our models, increasing the estimation error norm. This explains the degradation in the performance metrics with increasing p, but the results improve with increasing sample size, achieving comparable F_1 scores of 0.880, 0.873 and 0.856 for p =60, 120 and 240, respectively, when n =2048.

5) AR MODELS WITH NON-GAUSSIAN DATA

Here we use the AR model (100) as in Sec. VI-A1, except that now $\{\boldsymbol{w}(t)\}$ is i.i.d., zero-mean and non-Gaussian; other details remain unchanged. We have the components of $\boldsymbol{w}(t)$ as mutually independent and identically distributed, zero-mean, unit variance either exponential or Laplace random variables, whereas recall that in Sec. VI-A1, they are Gaussian. As discussed in Sec. III-C, the resulting graphs are no longer CIGs, but are partial correlation graphs reflecting uncorrelation instead of independence. However, since the PSDs are identical for all three cases (Gaussian, exponential or Laplace $\{\boldsymbol{w}(t)\}$), the corresponding graphs based on $\boldsymbol{S}_x^{-1}(f)$ are identical. The simulation results are shown in Table 6 using the proposed approaches, DE:lasso and DE:log-sum, with $d = 3$. The probability density function (pdf) of Laplacian $\boldsymbol{w}(t)$ has heavier (longer) tails than Gaussian pdf of $\boldsymbol{w}(t)$, and the pdf of exponential $\boldsymbol{w}(t)$ is skewed. The simulation results based on 100 runs are shown in Table 6. Compared to the Gaussian AR(3) model results shown in Table 1, the results shown in Table 6 are quite similar. For instance, for $n = 2048$, for the proposed DE:log-sum approach, we have an F_1 score of 0.873, 0.868 and 0.867, and Hamming distance of 172, 178 and 180, for the Gaussian, Laplace and exponential series, respectively. For $n = 256$, for the proposed DE:log-sum approach, we have an F_1 score of 0.621, 0.621 and 0.619, and Hamming distance of 466, 468 and 466, for the Gaussian, Laplace and exponential series, respectively.

TABLE 7. AR(3), ARFIMA(3,0.15,0) and ARFIMA(3,0.35,0) Models, $p = 120$: F_1 scores, Hamming distances and timings (s), averaged over 100 runs. For fractional integration $d_i = \tilde{d} = 0.15$ or $0.35 \forall i \in [120]$. "DE:lasso and "DE:log-sum" are proposed delay embedding approaches with lasso and log-sum penalties, respectively ($d = 3$), "FD:lasso" and "FD:log-sum" are the frequency-domain approaches with lasso [10] and log-sum [11] penalties, respectively.

n	256	1024	2048
model	λ's and α's picked to maximize F_1		
DE:lasso F_1 score (σ)			
AR(3)	0.517 (0.042)	0.695 (0.032)	0.763 (0.032)
ARFIMA(3,0.15,0)	0.470 (0.043)	0.664 (0.044)	0.725 (0.040)
ARFIMA(3,0.35,0)	0.370 (0.037)	0.495 (0.042)	0.558 (0.052)
DE:log-sum F_1 score (σ)			
AR(3)	0.621 (0.038)	0.808 (0.028)	0.873 (0.021)
ARFIMA(3,0.15,0)	0.597 (0.046)	0.801 (0.038)	0.865 (0.032)
ARFIMA(3,0.35,0)	0.551 (0.051)	0.732 (0.049)	0.802 (0.047)
DE:lasso Hamming distance (σ)			
AR(3)	646 (108)	396 (56)	318 (51)
ARFIMA(3,0.15,0)	777 (123)	454 (77)	376 (63)
ARFIMA(3,0.35,0)	1267 (199)	900 (138)	714 (138)
DE:log-sum Hamming distance (σ)			
AR(3)	466 (44)	245 (35)	172 (26)
ARFIMA(3,0.15,0)	499 (78)	243 (45)	178 (36)
ARFIMA(3,0.35,0)	549 (87)	343 (63)	246 (50)
DE:lasso Timing (s) (σ)			
AR(3)	4.76 (0.79)	4.62 (0.76)	4.69 (0.82)
ARFIMA(3,0.15,0)	5.06 (0.80)	5.36 (0.98)	5.14 (1.13)
ARFIMA(3,0.35,0)	7.19 (1.69)	8.45 (1.51)	8.38 (1.58)
DE:log-sum Timing (s) (σ)			
AR(3)	12.11 (1.59)	12.83 (1.38)	13.11 (1.60)
ARFIMA(3,0.15,0)	13.51 (3.01)	14.37 (3.71)	13.60 (2.50)
ARFIMA(3,0.35,0)	29.06 (5.28)	25.51 (3.98)	24.82 (4.23)
FD:lasso F_1 score (σ)			
AR(3)	0.447 (0.032)	0.639 (0.028)	0.710 (0.023)
ARFIMA(3,0.15,0)	0.412 (0.036)	0.592 (0.052)	0.670 (0.040)
ARFIMA(3,0.35,0)	0.326 (0.032)	0.430 (0.041)	0.412 (0.028)
FD:log-sum F_1 score (σ)			
AR(3)	0.482 (0.030)	0.716 (0.039)	0.782 (0.021)
ARFIMA(3,0.15,0)	0.460 (0.037)	0.704 (0.022)	0.767 (0.038)
ARFIMA(3,0.35,0)	0.426 (0.041)	0.636 (0.040)	0.667 (0.032)
FD:lasso Hamming distance (σ)			
AR(3)	652 (106)	417 (41)	345 (33)
ARFIMA(3,0.15,0)	749 (135)	472 (52)	394 (59)
ARFIMA(3,0.35,0)	1119 (193)	826 (140)	1258 (108)
FD:log-sum Hamming distance (σ)			
AR(3)	601 (62)	312 (28)	256 (29)
ARFIMA(3,0.15,0)	637 (98)	346 (24)	264 (48)
ARFIMA(3,0.35,0)	605 (81)	408 (32)	460 (39)
FD:lasso Timing (s) (σ)			
AR(3)	4.13 (1.09)	5.43 (1.14)	6.35 (1.59)
ARFIMA(3,0.15,0)	4.75 (1.44)	7.05 (1.85)	7.54 (1.55)
ARFIMA(3,0.35,0)	8.31 (1.90)	12.08 (3.17)	10.85 (1.65)
FD:log-sum Timing (s) (σ)			
AR(3)	7.79 (0.92)	9.66 (2.24)	10.26 (2.77)
ARFIMA(3,0.15,0)	8.75 (1.52)	12.47 (2.16)	12.09 (2.05)
ARFIMA(3,0.35,0)	13.46 (2.73)	19.07 (2.16)	19.44 (3.10)

B. LONG-RANGE DEPENDENCE MODEL

We now consider an ARFIMA$(q, \tilde{d}, 0)$ model with $\boldsymbol{u}(t)$ in (34) being a VAR(3) models exactly as in Sec. VI-A1 with $p = 120$, and the fractionally integrated part is as described in Sec. III-B2 with $d_i = \tilde{d} = 0.15$ or 0.35, $\forall i \in [p]$. This leads to $\gamma = 1 - \tilde{d} = 0.85$ or 0.65 in Assumption (A1), and for large τ, the correlation function $\boldsymbol{R}_x(\tau)$ of $\{\boldsymbol{x}(t)\}$ follows $|[\boldsymbol{R}(\tau)]_{ij}| = \mathcal{O}(|\tau|^{-0.7})$ or $\mathcal{O}(|\tau|^{-0.3})\forall i, j \in [p]$ for

$\gamma = 0.85$ and 0.65, respectively (see Sec. III-B2). To generate $x_i(t)$ as in (35), we used a moving-average approximation $x_i(t) = \sum_{\ell=0}^{T-1} c_{i\ell} u_i(t-\ell)$ with $T = 2000$ [18]. We generate the AR part as in Sec. VI-A1 (with stability check etc.). As discussed in Remark 3 (Sec. III-B2), the underlying AR(3) sequence and the two ARFIMA sequences all have identical CIGs.

Simulation results based on 100 runs and using the proposed DE:lasso and DE:log-sum approaches, as well as the FD:lasso and FD:log-sum approaches, (all four approaches as in Sec. VI-A1 and Table 1), are shown in Table 7. We used delay $d = 3$ for all sample sizes for DE:lasso and DE:log-sum, and for FD:lasso and FD:log-sum, the number of frequency test points M in the interval $(0, 0.5)$ were $M = 2, 3, 4$ for sample sizes $n = 256, 1024, 2048$, respectively, as in Sec. VI-A1 and Table 1). The CIGs of the ARFIMA(3,0.15,0) and ARFIMA(3,0.35,0) processes are identical, and are exactly as the CIG of the underlying AR(3) model which was also used in Sec. VI-A1. In Table 7 we show the results for these three processes where the results for the AR(3) series are exactly as in Table 1. We see that the F_1 scores for ARFIMA(3,0.35,0) in Table 7 are lower than those for the ARFIMA(3,0.15,0) model, which, in turn, are lower than those for the AR(3) model, for the corresponding approaches, reflecting the bounds on $\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^{\diamond}\|_F$ given in Theorem 1. As noted in Remark 7, we see from Theorem 1 that $\|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^{\diamond}\|_F = \mathcal{O}_P(\delta_{*n}/\beta_{\min}^2)$ where $\beta_{\min}$ is defined in assumption (A3), and $\delta_{*n} \propto 1/n^{0.5}$ (cf. Lemma 2(a)) for SRD sequences and for LRD sequences with $\gamma > 0.75$, and $\delta_{*n} \propto 1/n^{0.3}$ (cf. Lemma 2(c)) for LRD sequences when $\gamma = 0.65$, with all other parameters remaining the same in R and r_n in (81) and (82), respectively. As discussed in Remark 5, $\beta_{\min}$ decreases with increasing $\tilde{d}$, with $\beta_{\min}$ being the largest for the AR(3) series and the smallest for ARFIMA(3,0.35,0) series, among the three series under discussion. The performance in Table 7 improves with sample size in terms of both the F_1 score and Hamming distance.

Note that the analysis of the frequency-domain approaches of [9], [10], [11], and [12] is restricted to SRD processes although they (or any other approach) can be applied to both SRD and LRD series. It is seen from Table 7 that the proposed delay embedding approaches DE:lasso and DE:log-sum significantly outperform the corresponding frequency-domain approaches FD:lasso and FD:log-sum. For instance, for $n = 2048$, for the proposed DE:log-sum approach, we have an F_1 score of 0.802 and Hamming distance of 246 for the ARFIMA(3,0.35,0) series, whereas FD:log-sum yields an F_1 score of 0.667 and Hamming distance of 460.

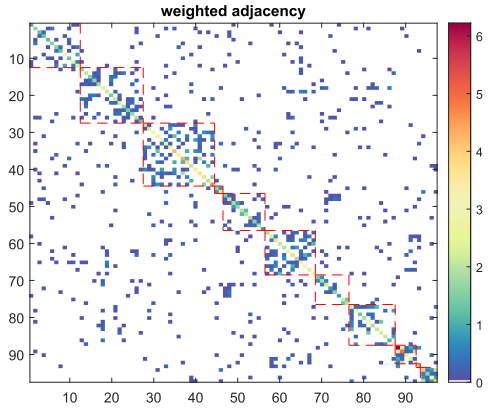
VII. REAL DATA: FINANCIAL TIME SERIES

Here our aim is to visualize and explore the conditional dependency structure underlying a financial time series. We consider daily share prices (at close of the day) of 97 stocks in the S&P 100 index from Jan. 1, 2013 through Jan. 14, 2015. This data was gathered from the Yahoo Finance website. Since the ground truth is not available, the main objective is to illustrate how our proposed approaches compare with some existing approaches, and how the real data results relate to the synthetic data results.

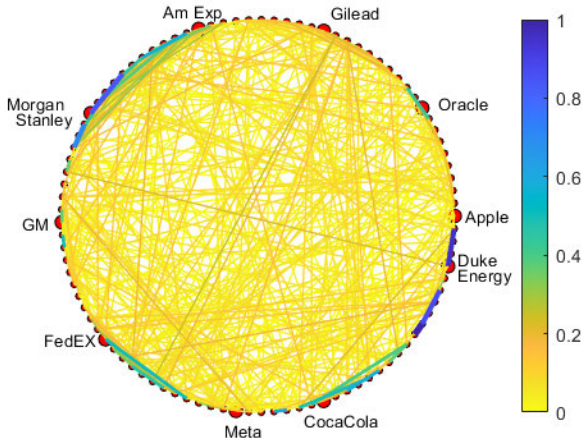
If $z_m(t)$ is share price of mth stock on day t, we consider (as is conventional in such studies) $x_m(t) = \ln(z_m(t)/z_m(t-1))$ as the time series to analyze. Such transformations are common in the analysis of financial time series. For instance, such pre-processing is used in [30, Sec. 5.2] for topology selection for graphical models for international stock market data. We have $x_m(t) = \ln(z_m(t)) - \ln(z_m(t-1))$ which implies that we first perform $\log(\cdot)$ transformation (generally believed to make data "more Gaussian"), followed by lag one differencing to make the data close to univariate uncorrelated and stationary.

These 97 stocks are classified into 11 sectors (according to the Global Industry Classification Standard (GICS)) and we order the nodes to group them as information technology (nodes 1-12), health care (nodes 13-27), financials (nodes 28-44), real estate (nodes 45-46), consumer discretionary (nodes 47-56), industrials (nodes 57-68), communication services (nodes 69-76), consumer staples (nodes 77-87), energy (nodes 88-92), materials (node 93), and utilities (nodes 94-97). For each m, $x_m(t)$ was centered and normalized to unit variance. The pre-processed data from Jan. 2, 2013 to Jan. 14, 2015 was taken as the time series $\{\boldsymbol{x}(t)\}$ with 512 samples, $p = 97$. The resulting time series weighted adjacency heatmaps and the corresponding graphs, using log-sum penalty ($\epsilon = 0.0001$), are shown in Fig. 1 for IID, freq.-domain (number of frequency test points $M = 3$) and the proposed delay embedding (delay $d = 3$) approaches, resulting in 508, 414 and 347, edges, respectively. The tuning parameter pair (λ, α) was selected via BIC (pseudo-BIC for DE as for the synthetic data examples in Table 1); for FD:log-sum we used the BIC method detailed in [10] and [11], and for IID:log-sum we used the classical BIC method [62] to select λ. The strength of the connections in the graphs and the weights of the adjacency matrices in Fig. 1 are based on the metrics used to determine presence/absence of graph edges: $\|\hat{\mathbf{\Omega}}_{y\lambda}^{(k\ell)}\|_F$ (see (62)) for the proposed DE approach, and $|[\hat{\mathbf{\Omega}}_x]_{k\ell}|$ for the IID approach for heatmaps, and normalized $\|\hat{\mathbf{\Omega}}_{y\lambda}^{(k\ell)}\|_F / (\max_{k,\ell:\, k\neq\ell} \|\hat{\mathbf{\Omega}}_{y\lambda}^{(k\ell)}\|_F)$ for graph edge weights which are color-coded. In the heatmaps shown in Figs. 1(a), 1(c) and 1(e), the nodes are labeled by integers 1-97 and they have been ordered in terms of the 11 GICS sectors, as stated earlier. In the graphs shown in Figs. 1(b), 1(d) and 1(f), some of the nodes (larger red circles) are labeled by names, with node 1 labeled as Apple. Starting from Apple as node 1, one counts counterclock-wise. Thus, the node labeled Oracle is node 10, and nodes labeled Gilead, Am Exp, Morgan Stanley, GM. FedEX, Meta, CocaCola and Duke Energy are nodes 20, 30, 41, 50, 60, 72, 79 and 94, respectively.

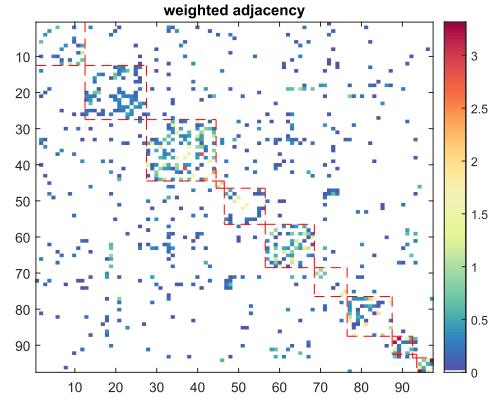
The IID model based graph in Fig. 1(a) is relatively dense with 508 edges. The FD:log-sum and the proposed



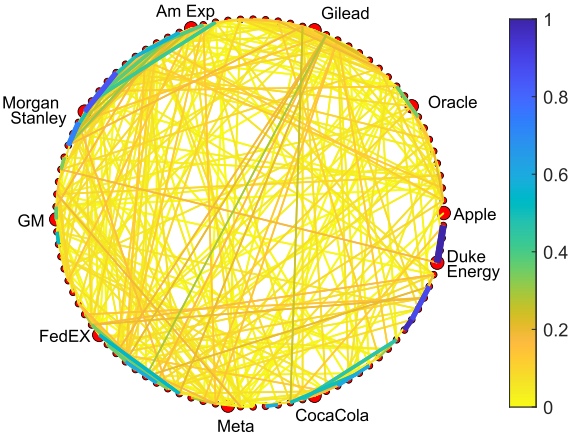
(a) Heatmap, IID:log-sum, 508 edges



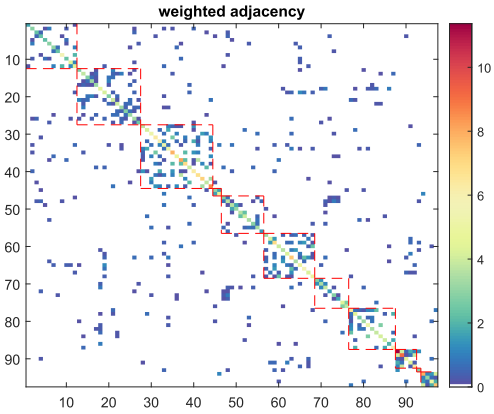
(b) Graph, IID:log-sum, 508 edges



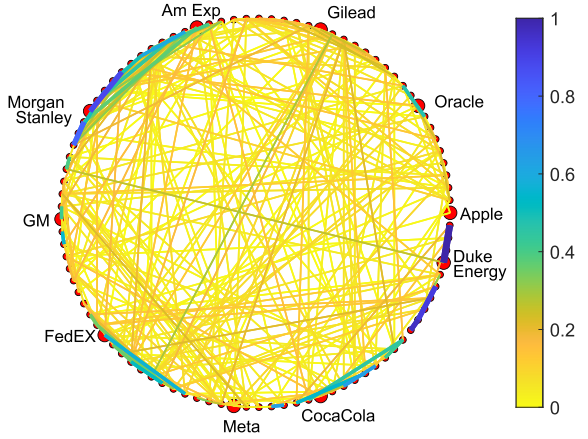
(c) Heatmap, FD:log-sum, 414 edges



(d) Graph, FD:log-sum, 414 edges

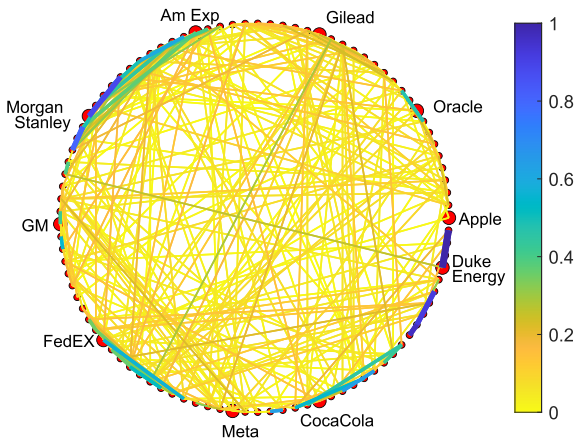


(e) Heatmap, DE:log-sum, 347 edges

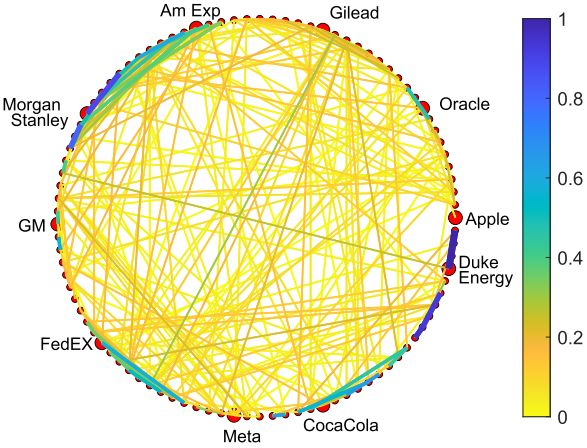


(f) Graph, DE:log-sum, 347 edges

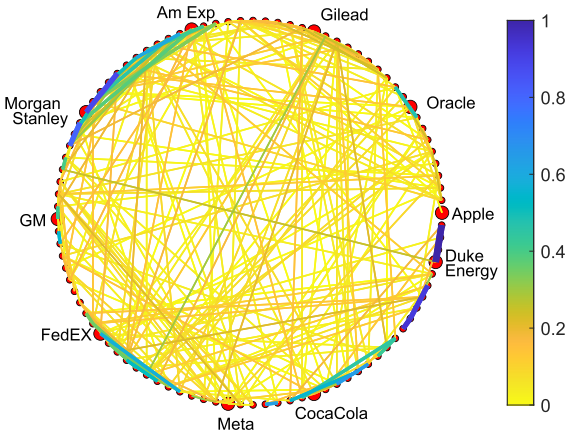
FIGURE 1. **Weighted adjacency heatmaps and conditional independence graphs for financial time series (S&P 97 stocks share prices) over period Jan. 2, 2013 to Jan. 14, 2015 (512 samples): (a)-(b) time-domain IID modeling with log-sum penalty, (c)-(d) freq-domain approach with log-sum penalty, (e)-(f) proposed delay embedding approach with log-sum penalty (delay d=3). The heatmaps show $\|\hat{\Omega}^{(k\ell)}_{y\lambda}\|_F$, $k, \ell \in$ [97] and the graphs show color-coded normalized $\|\hat{\Omega}^{(k\ell)}_{y\lambda}\|_F/(\max_{k,\ell:\, k\neq\ell}\|\hat{\Omega}^{(k\ell)}_{y\lambda}\|_F)$. In Figs. (a), (c) and (e), the red squares (in dashed lines) show the 11 GICS sectors – they are not part of the edge weights. The nodes are labeled as numbers 1 through 97 in heatmaps, and in the graphs, one counts counter-clockwise starting with the node labeled Apple, which is node 1. Nodes 10, 20, 30, 41, 50, 60, 72, 79 and 94 are labeled Oracle, Gilead, Am Exp, Morgan Stanley, GM, FedEx, Meta, CocaCola and Duke Energy, respectively.**



(a) DE:log-sum with delay=3, 347 edges



(b) DE:log-sum with delay=4, 307 edges



(c) DE:log-sum with delay=5, 310 edges

FIGURE 2. Conditional independence graphs for financial time series (S&P 97 stocks share prices) over period Jan. 2, 2013 to Jan. 14, 2015 (512 samples). All graphs are based on the proposed delay embedding approach with log-sum penalty: (a) delay d=3, (b) delay d=4, (c) delay d=5. Other details are as in fig. 1.

DE:log-sum approaches yield sparser graphs with 414 and 347 edges, respectively. These observations are consistent with the synthetic data results shown in Table 1 where in terms of the F_1 score and the Hamming distance, DE:log-sum outperforms FD:log-sum, which, in turn, outperforms the IID:log-sum approach. All three graphs appear to agree on stronger edges but differ in low-strength edges, which very likely to be false in the graph resulting from the IID:log-sum approach (cf. synthetic data results). We also see that in general, the stocks in a GICS sector are more strongly connected to each other compared to the stocks in a different sector. For instance, the stocks in the financial sector are nodes 28-44 (including American Express (Am Exp) at node 30 and Morgan Stanley at node 41) and they are strongly connected to each other in all three graphs. Similarly, the stocks in the industrials sector are nodes 57-68 (including Federal Express (FedEX) at node 60) and they too are strongly connected to each other in all three graphs.

The degree of a node in an undirected graph is defined as the number of edges incident on the node. The maximum degrees of the estimated graphs using IID:log-sum, FD:log-sum and DE:log-sum approaches are 17, 24 and 17, respectively, and the corresponding mean degrees ($\pm$ standard deviation) are 10.47 ($\pm$2.70), 8.54 ($\pm$4.14) and 7.15 ($\pm$2.86), respectively. Some of the highest degree nodes are:

- *IID:log-sum graph*
  Johnson and Johnson (node 21, degree=17),
  Blackrock (node 33, degree=17),
  Berkshire Hathaway (node 34, degree=17),
  Metlife (node 40, degree=16),
  Emerson Electric (node 59, degree=16),
  3M Co. (node 65, degree=16).
- *FD:log-sum graph*
  Blackrock (node 33, degree=24),
  Berkshire Hathaway (node 34, degree=21),
  Honeywell (node 63, degree=19),
  Johnson and Johnson (node 21, degree=18),
  Danaher Corp. (node 19, degree=17),
  Emerson Electric (node 59, degree=16).
- *DE:log-sum graph*
  Blackrock (node 33, degree=17),
  3M Co. (node 65, degree=15),
  Metlife (node 40, degree=13),
  Honeywell (node 63, degree=13),
  Johnson and Johnson (node 21, degree=12),
  Thermo Fisher Scientific (node 26, degree=12),
  Berkshire Hathaway (node 34, degree=12),
  Booking Holdings (node 48, degree=12),
  Walt Disney (node 71, degree=12).

The strongest six edges as measured by the edge weights, are (in the order of decreasing weights):

- *IID:log-sum graph*
  {88, 89}={Conoco Phillips, Chevron},
  {94, 97}={Duke Energy, Southern Co.},
  {89, 92}={Chevron, Exxon Mobil},
  {37, 41}={Goldman Sachs, Morgan Stanley},

{42, 44}={US Bancorp, Wells Fargo},
{75, 76}={AT&T, Verizon}.

- *FD:log-sum graph*
{94, 97}={Duke Energy, Southern Co.},
{88, 89}={Conoco Phillips, Chevron},
{89, 92}={Chevron, Exxon Mobil},
{37, 41}={Goldman Sachs, Morgan Stanley},
{42, 44}={US Bancorp, Wells Fargo},
{81, 84}={Altria Group, Phillip Morris}.
- *DE:log-sum graph*
{94, 97}={Duke Energy, Southern Co.},
{88, 89}={Conoco Phillips, Chevron},
{88, 92}={Conoco Phillips, Exxon Mobil},
{37, 41}={Goldman Sachs, Morgan Stanley}),
{42, 44}={US Bancorp, Wells Fargo},
{81, 84}={Altria Group, Phillip Morris}.

Notice that the strongest edges detailed above are between node pairs that are in the same GICS sector.

An in-depth analysis would require domain expertise and is not within the scope of this paper. Our objective here has been to illustrate how our proposed approaches compare with some existing approaches.

In Fig. 2 we turn to the issue raised/addressed in Remark 2 (Sec. III-A) and Table 2 (Sec. VI-A2: how to pick time delay d? It was argued in Remark 2 to follow the heuristic of trying several values of d and picking the smallest value $d = d_0$ beyond which (e.g., for $d = d_0 + 1$) the estimated graph is only marginally different from the estimated graph for $d = d_0$. The synthetic data results of Sec. VI-A2 and Table 2 support this heuristic. In practice, since the true graph is unknown, one cannot compute the F_1 score or the Hamming distance to quantify marginal difference. It was stated in Remark 2 that what constitutes marginal difference would be problem and user dependent. Using the DE:log-sum approach (coupled with the BIC method for (λ, α) selection) for delay values of d =3, 4 and 5, we show the estimated graphs in Fig. 2. Fig. 2(a) is the same as Fig. 1(f). The estimated graphs have 347, 307 and 310 edges for $d = 3$, 4 and 5, respectively. The graphs for $d = 3$ and $d = 4$ have 305 edges in common and they disagree on 44 edges, while the graphs for $d = 4$ and $d = 5$ have 304 edges in common and they disagree on 9 edges. A visual examination also shows very minor differences in the three graphs. This suggests that one may take Fig. 2(a) (i.e., Fig. 1(f)) as the estimated graph.

VIII. CONCLUSION

Estimation of the CIG of a high-dimensional stationary multivariate Gaussian time series was investigated using delay embedding and time-domain methods which allowed the series to be possibly LRD, unlike past work on this topic where frequency-domain methods whose analysis is restricted to SRD series, abound. The problem was formulated as one of multi-attribute graph estimation for random vectors where a vector is associated with each node of the graph. For an embedding dimension d, at each node the associated $(d + 1)$-dimensional random vector consists of a time series component and its d delayed copies, with integer delays up to $d \geq 1$. A penalized negative pseudo log-likelihood function with convex lasso or non-convex regularizations was proposed to estimate the CIG. An existing ADMM approach was used to optimize the objective function. We established sufficient conditions in a high-dimensional setting for consistency, local convexity when using non-convex penalties, and graph recovery, for both SRD and LRD series.

We illustrated our approach with numerical examples using both synthetic and real (financial time series) data where comparisons with some competing approaches were provided. Compared to i.i.d. modeling based approaches, our proposed approaches yield much higher F_1 scores and lower Hamming distances between estimated and true edge-sets of the time series graph. Compared to the existing frequency-domain approaches, our proposed approaches are slightly to significantly better. For a p-dimensional time series, the computational complexity of the ADMM algorithm is $\mathcal{O}((d+1)^3 p^3)$. Based on the numerical examples, a heuristic was suggested and illustrated for selection of d which, in theory, should tend to infinity to infer the time series graph.

In our theoretical analysis we exploited some results from [43] whose analysis is limited to i.i.d. data. For analysis, in [43] two different sets of sufficient conditions have been presented: with and without some irrepresentability conditions. In this paper we did not consider any irrepresentability condition because while results of [43, Sec. V-B] are applicable here, they are stated in terms of the covariance of the multi-attribute process $\{\boldsymbol{y}(t)\}$ whereas in this paper we would like to formulate them in terms of the original process $\{\boldsymbol{x}(t)\}$ also, which we are unable to do, unlike as in Remarks 4 and 5 of this paper.

The assumptions that $\{\boldsymbol{x}(t)\}$ is stationary and linear (i.e., generated via (24)) are critical to the results of this paper. If $\{\boldsymbol{x}(t)\}$ is not necessarily Gaussian, then the zeros of $\boldsymbol{S}_x^{-1}(f)$ encode a partial correlation graph [5] which is not necessarily a CIG. For non-Gaussian sequences, our time-delay embedding approach yields an approximation to the underlying time series partial correlation graph.

APPENDIX A
PROOF OF LEMMA 2

Here we prove Lemma 2 for which we need [48, Lemma VI.2,supplementary], restated here as Lemma 4 (γ is called β in [48]) as applied to $\hat{\boldsymbol{\Sigma}}_y = \frac{1}{\bar{n}} \sum_{t=d+1}^{n} \boldsymbol{y}(t)\boldsymbol{y}^\top(t)$, $\bar{n} = n - d$, $\boldsymbol{\Sigma}_y^\diamond = E\{\boldsymbol{y}(t)\boldsymbol{y}^\top(t)\}$. (The cases $\gamma = 1$ and $\gamma = \frac{3}{4}$ are not in [48, Lemma VI.2,supplementary], but can be inferred from [48, Lemma VI.1,supplementary].) It is the basis of our Lemma 2, which in turn, is central to our Theorem 1.

Lemma 4 [48]: Under assumption (A1), the elements of the sample covariance $\hat{\boldsymbol{\Sigma}}_y$ satisfy the tail bound

$$P\left(\left|[\hat{\boldsymbol{\Sigma}}_y - \boldsymbol{\Sigma}_y^\diamond]_{kl}\right| \geq \delta\right)$$

$$\leq 2\,\exp\left(-C_u \min\left(\frac{\delta^2}{c_a^4(p)L_{n,\gamma}}, \frac{\delta}{c_a^2(p)J_{n,\gamma}}\right)\right) \quad (104)$$

for every $\delta > 0$ and every k, l, where $C_u \in (0,\infty)$ is a constant that depends only on γ, and

$$(L_{n,\gamma}^{-1}, J_{n,\gamma}^{-1}) = \begin{cases} (\bar{n}, \bar{n}), & \gamma > 1 \\ (\bar{n}, \bar{n}/\ln^2(\bar{n})), & \gamma = 1 \\ (\bar{n}, \bar{n}^{2\gamma-1}), & 1 > \gamma > \frac{3}{4} \\ (\bar{n}/\ln(\bar{n}), \bar{n}^{2\gamma-1}), & \gamma = \frac{3}{4} \\ (\bar{n}^{4\gamma-2}, \bar{n}^{2\gamma-1}), & \frac{3}{4} > \gamma > \frac{1}{2}\,. \end{cases} \bullet$$

Remark 10. The universal constant C_u results from the application of the Hanson-Wright inequality [47]. In [48], the constant $c_a(p)$ (see assumption (A1)) is absorbed in C_u. We keep it separate to highlight its possible dependence on p. □

Proof of Lemma 2: For $\gamma \in (1,\infty)$ and $\bar{n} \geq N_1$, using Lemma 4 it is easy to verify that $\delta_{*n}^2/(c_a^4(p)L_{n,\gamma}) \leq \delta_{*n}/(c_a^2(p)J_{n,\gamma})$ for the specified δ_{*n}. Applying the union bound over all $(mp_n)^2$ entries of $\hat{\mathbf{\Sigma}}_y - \mathbf{\Sigma}_y^\diamond$ in Lemma 4, we have

$$\begin{aligned} &P\left(\max_{k,l}\left|[\hat{\mathbf{\Sigma}}_y - \mathbf{\Sigma}_y^\diamond]_{kl}\right| > \delta_{*n}\right) \leq P_{tb} \\ &= 2(mp_n)^2\,\exp\left(-C_u\,\frac{\ln(2m^2p_n^\tau)}{\bar{n}C_u}\,\bar{n}\right) \\ &= 2(mp_n)^2\,\exp\left(\ln(2m^2p_n^\tau)^{-1}\right) = \frac{1}{p_n^{\tau-2}}\,. \end{aligned} \quad (105)$$

For $\gamma \in (\frac{3}{4}, 1]$ and stated conditions on $\bar{n}$, we again have $\delta_{*n}^2/(c_a^4(p)L_{n,\gamma}) \leq \delta_{*n}/(c_a^2(p)J_{n,\gamma})$ for the specified δ_{*n}, and therefore, the desired result follows in a manner similar to (105). For part (b), when $\gamma = \frac{3}{4}$, for the specified δ_{*n} and stated condition on $\bar{n}$, we have $\delta_{*n}^2/(c_a^4(p)L_{n,\gamma}) \leq \delta_{*n}/(c_a^2(p)J_{n,\gamma})$ for the specified δ_{*n}, and therefore, the desired result follows as before. For $\gamma \in (\frac{1}{2}, \frac{3}{4})$, for the specified δ_{*n}, it is easy to verify that we have $\delta_{*n}^2/(c_a^4(p)L_{n,\gamma}) \leq \delta_{*n}/(c_a^2(p)J_{n,\gamma})$ if $N_1 \leq 1$, else we have $\delta_{*n}^2/(c_a^4(p)L_{n,\gamma}) > \delta_{*n}/(c_a^2(p)J_{n,\gamma})$ if $N_1 > 1$. For $N_1 \leq 1$, the desired results follows in a manner similar to (105). If $N_1 > 1$, $\gamma \in (\frac{1}{2}, \frac{3}{4})$ and $\delta_{*n} = c_a^2(p)\ln(2m^2p_n^\tau)/(\bar{n}^{2\gamma-1}C_u)$, then applying the union bound to left-side of (66), we have

$$\begin{aligned} &P\left(\max_{k,l}\left|[\hat{\mathbf{\Sigma}}_y - \mathbf{\Sigma}_y^\diamond]_{kl}\right| > \delta_{*n}\right) \leq P_{tb} \\ &= 2(mp_n)^2\,\exp\left(-C_u\,\frac{\ln(2m^2p_n^\tau)}{(\bar{n}^{2\gamma-1}C_u)}\,\bar{n}^{2\gamma-1}\right) \\ &= 2(mp_n)^2\,\exp\left(\ln(2m^2p_n^\tau)^{-1}\right) = \frac{1}{p_n^{\tau-2}}\,. \end{aligned} \quad (106)$$

This completes the proof. ■

APPENDIX B
PROOFS OF THEOREMS 1-3 AND LEMMA 3

First some notation. Given $\boldsymbol{A} \in \mathbb{R}^{p\times p}$, $\boldsymbol{A}^+ = \text{diag}(\boldsymbol{A})$ is a diagonal matrix with the same diagonal as $\boldsymbol{A}$, and $\boldsymbol{A}^- = \boldsymbol{A} - \boldsymbol{A}^+$ is $\boldsymbol{A}$ with all its diagonal elements set to zero.

Proof of Theorem 1: To lighten notation, in this proof, we denote $\mathbf{\Omega}_y$, $\mathbf{\Omega}_y^\diamond$, $\mathbf{\Sigma}_y^\diamond$, $\mathcal{E}_y^\diamond$ and $\hat{\mathbf{\Omega}}_\lambda$ as $\mathbf{\Omega}$, $\mathbf{\Omega}^\diamond$, $\mathbf{\Sigma}^\diamond$, $\mathcal{E}^\diamond$ and $\hat{\mathbf{\Omega}}$, respectively. Let $\mathbf{\Omega} = \mathbf{\Omega}^\diamond + \mathbf{\Delta}$ with both $\mathbf{\Omega}$, $\mathbf{\Omega}^\diamond \succ \mathbf{0}$, and

$$Q(\mathbf{\Omega}) := \bar{\mathcal{L}}(\mathbf{\Omega}) - \bar{\mathcal{L}}(\mathbf{\Omega}^\diamond)\,. \quad (107)$$

The estimate $\hat{\mathbf{\Omega}}$ minimizes $Q(\mathbf{\Omega})$, or equivalently, $\hat{\mathbf{\Delta}} = \hat{\mathbf{\Omega}} - \mathbf{\Omega}^\diamond$ minimizes $G(\mathbf{\Delta}) := Q(\mathbf{\Omega}^\diamond + \mathbf{\Delta})$. We will follow the method of proof of [43, Theorem 1], which, in turn, for the most part, follows the method of proof of [38, Theorem 1] pertaining to lasso penalty. Consider the set

$$\Theta_n(R) := \left\{\mathbf{\Delta} : \mathbf{\Delta} = \mathbf{\Delta}^\top,\ \|\mathbf{\Delta}\|_F = Rr_n\right\} \quad (108)$$

where R and r_n are as in (81) and (82), respectively. Since $G(\hat{\mathbf{\Delta}}) \leq G(\mathbf{0}) = 0$, if we can show that $\inf_{\mathbf{\Delta}}\{G(\mathbf{\Delta}) : \mathbf{\Delta} \in \Theta_n(R)\} > 0$, then a minimizer $\hat{\mathbf{\Delta}}$ must be inside $\Theta_n(R)$, and hence $\|\hat{\mathbf{\Delta}}\|_F \leq Rr_n$. It is shown in [38, (9)] that

$$\ln(|\mathbf{\Omega}^\diamond + \Delta|) - \ln(|\mathbf{\Omega}^\diamond|) = \text{tr}(\mathbf{\Sigma}^\diamond\mathbf{\Delta}) - A_1 \quad (109)$$

where, with $\boldsymbol{H}(\mathbf{\Omega}^\diamond, \mathbf{\Delta}, v) = (\mathbf{\Omega}^\diamond + v\mathbf{\Delta})^{-1} \otimes (\mathbf{\Omega}^\diamond + v\mathbf{\Delta})^{-1}$ and v denoting a scalar,

$$A_1 = \text{vec}(\mathbf{\Delta})^\top \left(\int_0^1 (1-v)\boldsymbol{H}(\mathbf{\Omega}^\diamond, \mathbf{\Delta}, v)\,dv\right)\text{vec}(\mathbf{\Delta})\,. \quad (110)$$

Noting that $\mathbf{\Omega}^{-1} = \mathbf{\Sigma}$, we can rewrite $G(\mathbf{\Delta})$ as

$$G(\mathbf{\Delta}) = A_1 + A_2 + A_3 + A_4\,, \quad (111)$$

where

$$A_2 = \text{tr}\left((\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}^\diamond)\mathbf{\Delta}\right)\,, \quad (112)$$

$$A_3 = \alpha \sum_{i,j=1;i\neq j}^{mp_n} \left(\rho_\lambda(|\Omega_{ij}^\diamond + \Delta_{ij}|) - \rho_\lambda(|\Omega_{ij}^\diamond|)\right)\,, \quad (113)$$

$$\begin{aligned} A_4 = (1-\alpha)m \sum_{q,\ell=1;q\neq\ell}^{p_n} &\left(\rho_\lambda(\|(\mathbf{\Omega}^\diamond)^{(q\ell)} + \mathbf{\Delta}^{(q\ell)}\|_F)\right. \\ &\left. - \rho_\lambda(\|(\mathbf{\Omega}^\diamond)^{(q\ell)}\|_F)\right)\,. \end{aligned} \quad (114)$$

Following [38, p. 502], we have

$$A_1 \geq \frac{\|\mathbf{\Delta}\|_F^2}{2(\|\mathbf{\Omega}^\diamond\| + \|\mathbf{\Delta}\|)^2} \geq \frac{\|\mathbf{\Delta}\|_F^2}{2\left(\beta_{\min}^{-1} + Rr_n\right)^2} \quad (115)$$

where we have used the fact that $\|\mathbf{\Omega}^\diamond\| = \|(\mathbf{\Sigma}^\diamond)^{-1}\| = \phi_{\max}((\mathbf{\Sigma}^\diamond)^{-1}) = (\phi_{\min}(\mathbf{\Sigma}^\diamond))^{-1} \leq \beta_{\min}^{-1}$ and $\|\mathbf{\Delta}\| \leq \|\mathbf{\Delta}\|_F = Rr_n$. We now consider A_2 in (112). We have

$$A_2 = \underbrace{\sum_{i,j=1;i\neq j}^{mp_n} [\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}^\diamond]_{ij}\Delta_{ji}}_{L_1} + \underbrace{\sum_{i=1}^{mp_n} [\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}^\diamond]_{ii}\Delta_{ii}}_{L_2} \quad (116)$$

To bound L_1, using Lemma 2, with probability $> 1 - 1/p_n^{\tau-2}$,

$$|L_1| \leq \|\mathbf{\Delta}^-\|_1 \max_{i,j}\left|[\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}^\diamond]_{ij}\right| \leq \|\mathbf{\Delta}^-\|_1\,\delta_{*n}\,. \quad (117)$$

Similarly, by Cauchy-Schwarz inequality, [42, Lemma 2] and (82),

$$|L_2| \le \|\mathbf{\Delta}^+\|_1 \, \delta_{*n} \le \delta_{*n} \sqrt{mp_n} \, \|\mathbf{\Delta}^+\|_F \le \|\mathbf{\Delta}^+\|_F r_n \, . \tag{118}$$

Therefore, with probability $> 1 - 1/p_n^{\tau-2}$,

$$|A_2| \le \|\mathbf{\Delta}^-\|_1 \, \delta_{*n} + \|\mathbf{\Delta}^+\|_F r_n \, . \tag{119}$$

We now derive a different bound on A_2. Define $\tilde{\mathbf{\Delta}} \in \mathbb{R}^{p_n \times p_n}$ with (i,j)th element $\tilde{\Delta}_{ij} = \|\mathbf{\Delta}^{(ij)}\|_F$, where $\mathbf{\Delta}^{(ij)}$ is defined from $\mathbf{\Delta}$ similar to (9). By the Cauchy-Schwarz inequality,

$$\|\mathbf{\Delta}^-\|_1 = \sum_{i,j=1; i\neq j}^{mp_n} |\Delta_{ij}| \le m \|\tilde{\mathbf{\Delta}}^-\|_1 + \underbrace{\Big(\sum_{k=1}^{p_n} \|\mathbf{\Delta}^{(kk)}\|_1 - \|\mathbf{\Delta}^+\|_1 \Big)}_{=:B} \, . \tag{120}$$

Then using $\sum_k \|\mathbf{\Delta}^{(kk)}\|_1 \le m \sum_k \tilde{\Delta}_{kk} \le m\sqrt{p_n} \, \|\tilde{\mathbf{\Delta}}^+\|_F$, we have

$$|L_2| + \delta_{*n} B \le \delta_{*n} \Big(\sum_{k=1}^{p_n} \|\mathbf{\Delta}^{(kk)}\|_1 \Big) \le \|\tilde{\mathbf{\Delta}}^+\|_F \sqrt{m} \, r_n \, .$$

Therefore, an alternative bound is

$$|A_2| \le m \|\tilde{\mathbf{\Delta}}^-\|_1 \, \delta_{*n} + \sqrt{m} \, \|\tilde{\mathbf{\Delta}}^+\|_F r_n \, . \tag{121}$$

For the rest of the proof we have two different approaches, one for lasso and log-sum and the other for SCAD and MCP penalties. The following applies to lasso and log-sum penalties.

For Lasso and Log-Sum Penalties: We now bound A_3 in (113). Let $\bar{\mathcal{E}}^\diamond$ denote the true enlarged edge-set corresponding to $\mathcal{E}^\diamond$ (referred to as $\mathcal{E}_y^\diamond$ in assumption (A2)) when one interprets multi-attribute model as a single-attribute model with mp_n nodes. Let $(\bar{\mathcal{E}}^\diamond)^c$ denote its complement. Using the mean-value theorem, we have ($\rho'_\lambda(u) = \frac{d\rho_\lambda(u)}{du}$)

$$\rho_\lambda(|\Omega_{ij}^\diamond + \Delta_{ij}|) = \rho_\lambda(|\Omega_{ij}^\diamond|) + \rho'_\lambda(|\tilde{\Omega}_{ij}|)(|\Omega_{ij}^\diamond + \Delta_{ij}| - |\Omega_{ij}^\diamond|) \tag{122}$$

where $|\tilde{\Omega}_{ij}| = |\Omega_{ij}^\diamond| + \nu(|\Omega_{ij}^\diamond + \Delta_{ij}| - |\Omega_{ij}^\diamond|)$ for some $\nu \in [0,1]$. We have

$$\begin{aligned} A_3 &= \alpha \sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} \rho'_\lambda(|\tilde{\Omega}_{ij}|)(|\Omega_{ij}^\diamond + \Delta_{ij}| - |\Omega_{ij}^\diamond|) \\ &\quad + \alpha \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} \rho_\lambda(|\Delta_{ij}|) \qquad (123) \\ &\ge -\alpha \sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} \rho'_\lambda(|\tilde{\Omega}_{ij}|)|\Delta_{ij}| + \alpha \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} C_\lambda |\Delta_{ij}| \\ &\quad \text{for } \ |\Delta_{ij}| \le \delta_\lambda, \qquad (124) \end{aligned}$$

where in the last step above, we used the triangle inequality and (49). Now use property (ii) of the penalty functions and $C_\lambda = \lambda/2$ to conclude that

$$A_3 \ge -\alpha\lambda_n \sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} |\Delta_{ij}| + \alpha(\lambda_n/2) \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} |\Delta_{ij}| \, . \tag{125}$$

Next we bound A_4 in (114). Considering the true edge-set $\mathcal{E}^\diamond$ for the multi-attribute graph, let $(\mathcal{E}^\diamond)^c$ denote its complement. If the edge $\{i,j\} \in (\mathcal{E}^\diamond)^c$, then $(\mathbf{\Omega}^\diamond)^{(ij)} = \mathbf{0}$, therefore, $\|(\mathbf{\Omega}^\diamond)^{(ij)} + \mathbf{\Delta}^{(ij)}\|_F - \|(\mathbf{\Omega}^\diamond)^{(ij)}\|_F = \|\mathbf{\Delta}^{(ij)}\|_F$. For $\{i,j\} \in \mathcal{E}^\diamond$, by the triangle inequality, $\|(\mathbf{\Omega}^\diamond)^{(ij)} + \mathbf{\Delta}^{(ij)}\|_F - \|(\mathbf{\Omega}^\diamond)^{(ij)}\|_F \ge -\|\mathbf{\Delta}^{(ij)}\|_F$. Mimicking the steps for bounding A_3, we have

$$\begin{aligned} A_4 &\ge -(1-\alpha)m\lambda_n \sum_{(i,j)\in\mathcal{E}^\diamond} \|\mathbf{\Delta}^{(ij)}\|_F \\ &\quad + (1-\alpha)m(\lambda_n/2) \sum_{(i,j)\in(\mathcal{E}^\diamond)^c} \|\mathbf{\Delta}^{(ij)}\|_F \, . \qquad (126) \end{aligned}$$

Split A_2 as $A_2 = \alpha A_2 + (1-\alpha)A_2$, apply bound (119) to αA_2 and (121) to $(1-\alpha)A_2$, use $\|\mathbf{\Delta}^-\|_1 = \|\mathbf{\Delta}^-_{\bar{\mathcal{E}}^\diamond}\|_1 + \|\mathbf{\Delta}^-_{(\bar{\mathcal{E}}^\diamond)^c}\|_1$ and $\|\tilde{\mathbf{\Delta}}^-\|_1 = \|\tilde{\mathbf{\Delta}}^-_{\bar{\mathcal{E}}^\diamond}\|_1 + \|\tilde{\mathbf{\Delta}}^-_{(\mathcal{E}^\diamond)^c}\|_1$. We have

$$\begin{aligned} \alpha A_2 + A_3 &\ge -\alpha|A_2| + \alpha\lambda_n(0.5\|\mathbf{\Delta}^-_{(\bar{\mathcal{E}}^\diamond)^c}\|_1 - \|\mathbf{\Delta}^-_{\bar{\mathcal{E}}^\diamond}\|_1) \\ &\ge \alpha(0.5\lambda_n - \delta_{*n}) \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} |\Delta_{ij}| \\ &\quad - \alpha(\lambda_n + \delta_{*n}) \sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} |\Delta_{ij}| - \alpha r_n \|\mathbf{\Delta}^+\|_F \, . \qquad (127) \end{aligned}$$

Since we pick $\lambda_n \ge \lambda_{n\ell}$ in Theorem 1, $0.5\lambda_n - \delta_{*n} \ge 0$ and therefore, the first term above can be neglected. Now $\sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} |\Delta_{ij}| \le \sqrt{m^2 s_n^\diamond} \|\mathbf{\Delta}\|_F$, by the Cauchy-Schwarz inequality, and $\|\mathbf{\Delta}^+\|_F \le \|\mathbf{\Delta}\|_F$. We then have

$$\alpha A_2 + A_3 \ge -\alpha \left((\lambda_n + \delta_{*n})\sqrt{m^2 s_n^\diamond} + r_n \right) \|\mathbf{\Delta}\|_F \, . \tag{128}$$

Similarly, we have

$$(1-\alpha)A_2 + A_4 \ge -(1-\alpha)m \times \left((\lambda_n + \delta_{*n})\sqrt{s_n^\diamond} + r_n \right) \|\mathbf{\Delta}\|_F \, . \tag{129}$$

From (128) and (129) we have

$$\begin{aligned} A_2 + A_3 + A_4 &\ge -\|\mathbf{\Delta}\|_F \left(\lambda_n m\sqrt{s_n^\diamond} + \delta_{*n} m\sqrt{s_n^\diamond} + mr_n \right) \\ &\ge -\|\mathbf{\Delta}\|_F \left(\lambda_n m\sqrt{s_n^\diamond} + (1+m)r_n \right) \\ &\ge -2(1+m)r_n \|\mathbf{\Delta}\|_F \qquad (130) \end{aligned}$$

where we used the fact that since $\lambda_n \le \lambda_{nu1}$, $\lambda_n m\sqrt{s_n^\diamond} \le (1+m)r_n$. Using (111), the bound (115) on A_1, bound (130) on $A_2 + A_3 + A_4$, and $\|\mathbf{\Delta}\|_F = Rr_n$, we have with probability $> 1 - 1/p_n^{\tau-2}$,

$$G(\mathbf{\Delta}) \ge \|\mathbf{\Delta}\|_F^2 \left[\frac{1}{2(\beta_{\min}^{-1} + Rr_n)^2} - \frac{2(1+m)}{R} \right] . \tag{131}$$

For the given choice of N_2, $Rr_n \le Rr_{N_2} \le 0.1/\beta_{\min}$ for $\bar{n} \ge N_2$. Also, $2(1+m)/R = \beta_{\min}^2/4$ by (81). Then for $\bar{n} \ge N_2$,

$$\frac{1}{2(\beta_{\min}^{-1}+Rr_n)^2} - \frac{2(1+m)}{R} \ge \beta_{\min}^2\left(\frac{1}{2.42}-\frac{1}{4}\right) > 0\,,$$

implying $G(\mathbf{\Delta}) > 0$. This proves (90). The choice of N_3 for log-sum penalty ensures that $|\Delta_{ij}| \le \delta_\lambda = \epsilon$ needed in (124) is satisfied with high probability (w.h.p.): if $Rr_n \le \epsilon$, then $|\Delta_{ij}| \le \|\mathbf{\Delta}\|_F \le Rr_n \le \epsilon$.

The following applies to SCAD and MCP penalties.

For SCAD and MCP Penalties: Here we address (122) differently. Using the triangle inequality, we have

$$\begin{aligned}|\tilde{\Omega}_{ij}| &\ge |\Omega^\diamond_{ij}| + \nu\big(|\Omega^\diamond_{ij}| - |\Delta_{ij}| - |\Omega^\diamond_{ij}|\big)\\ &\ge |\Omega^\diamond_{ij}| - |\Delta_{ij}|\,.\end{aligned} \tag{132}$$

Since $|\Delta_{ij}| \le \|\mathbf{\Delta}\|_F \le Rr_n$, the choice $\lambda_n = \lambda_{nu2}$ implies that $\lambda_n \ge Rr_n$, satisfying $|\Delta_{ij}| \le \lambda_n$. Therefore, $|\tilde{\Omega}_{ij}| \ge |\Omega^\diamond_{ij}| - \lambda_n$. For $\bar{n} \ge N_4$ and $\bar{n} \ge N_5$ for SCAD and MCP, respectively, $\rho'_\lambda(|\tilde{\Omega}_{ij}|) = 0$ (see (85) and (86)) if $\{i,j\} \in \bar{\mathcal{E}}^\diamond$, i.e, $|\Omega^\diamond_{ij}| \ne 0$, since in this case $|\tilde{\Omega}_{ij}| \ge (a+1)\lambda_n - \lambda_n = a\lambda_n$ for SCAD and $|\tilde{\Omega}_{ij}| \ge (b+1)\lambda_n - \lambda_n = b\lambda_n$ for MCP. Therefore,

$$\begin{aligned}A_3 &= \alpha \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} \rho_\lambda(|\Delta_{ij}|)\\ &\ge \alpha \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} C_\lambda|\Delta_{ij}| \ \text{ for } \ |\Delta_{ij}| \le \delta_\lambda\\ &= \alpha(\lambda_n/2) \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} |\Delta_{ij}|\,.\end{aligned} \tag{133}$$

Mimicking the steps for bounding A_3 above and under same conditions, we have

$$A_4 \ge (1-\alpha)m(\lambda_n/2) \sum_{(i,j)\in(\mathcal{E}^\diamond)^c} \|\mathbf{\Delta}^{(ij)}\|_F\,. \tag{134}$$

Thus,

$$\begin{aligned}\alpha A_2 + A_3 &\ge -\alpha|A_2| + 0.5\,\alpha\lambda_n\|\mathbf{\Delta}^-_{(\bar{\mathcal{E}}^\diamond)^c}\|_1\\ &\ge \alpha(0.5\lambda_n - \delta_{*n}) \sum_{(i,j)\in(\bar{\mathcal{E}}^\diamond)^c} |\Delta_{ij}|\\ &\quad - \alpha\delta_{*n} \sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} |\Delta_{ij}| - \alpha r_n\|\mathbf{\Delta}^+\|_F\,.\end{aligned} \tag{135}$$

Since we pick $\lambda_n = \max(Rr_n, \lambda_{nu1})$ in Theorem 1, $0.5\lambda_n - \delta_{*n} \ge 0$ and therefore, the first term above can be neglected. Now $\sum_{(i,j)\in\bar{\mathcal{E}}^\diamond} |\Delta_{ij}| \le \sqrt{m^2 s_n^\diamond}\|\mathbf{\Delta}\|_F$, by the Cauchy-Schwarz inequality, and $\|\mathbf{\Delta}^+\|_F \le \|\mathbf{\Delta}\|_F$. We then have

$$\alpha A_2 + A_3 \ge -\alpha\left(\delta_{*n}\sqrt{m^2 s_n^\diamond} + r_n\right)\|\mathbf{\Delta}\|_F\,. \tag{136}$$

By very similar arguments we also have

$$\begin{aligned}(1-\alpha)A_2 + A_4 &\ge -(1-\alpha)m\\ &\quad\times\left(\delta_{*n}\sqrt{s_n^\diamond} + r_n\right)\|\mathbf{\Delta}\|_F\,.\end{aligned} \tag{137}$$

From (136) and (137) we have

$$\begin{aligned}A_2 + A_3 + A_4 &\ge -\|\mathbf{\Delta}\|_F\left(\delta_{*n}m\sqrt{s_n^\diamond} + mr_n\right)\\ &\ge -(1+m)r_n\|\mathbf{\Delta}\|_F\end{aligned} \tag{138}$$

where we used the fact that $\delta_{*n}m\sqrt{s_n^\diamond} \le r_n$. Mimicking (131), we have with probability $> 1 - 1/p_n^{\tau-2}$, we have

$$\begin{aligned}G(\mathbf{\Delta}) &\ge \|\mathbf{\Delta}\|_F^2\left[\frac{1}{2(\beta_{\min}^{-1}+Rr_n)^2} - \frac{(1+m)}{R}\right]\\ &\ge \beta_{\min}^2\left(\frac{1}{2.42}-\frac{1}{8}\right) > 0\,,\end{aligned} \tag{139}$$

implying $G(\mathbf{\Delta}) > 0$. This proves (90). For the SCAD penalty, we need $|\Delta_{ij}| \le \delta_\lambda = \lambda_n$ in (133). Since $|\Delta_{ij}| \le \|\mathbf{\Delta}\|_F \le Rr_n$, the choice $\lambda_n = \lambda_{nu2}$ implies that $\lambda_n \ge Rr_n$, satisfying $|\Delta_{ij}| \le \lambda_n$. For the MCP we need $|\Delta_{ij}| \le \delta_\lambda = b\lambda_n$ in (133), which follows from $\lambda_n = \lambda_{nu2}$ and $|\Delta_{ij}| \le \lambda_n < b\lambda_n$ for $b > 1$. This completes the proof. ■

Proof of Lemma 3: Consider $h(\mathbf{\Omega}_y) = \mathcal{L}(\mathbf{\Omega}_y) - \frac{\mu}{2}\|\mathbf{\Omega}_y\|_F^2$ for some $\mu \ge 0$. The Hessian of $\mathcal{L}(\mathbf{\Omega}_y)$ w.r.t. $\text{vec}(\mathbf{\Omega}_y)$ is $\nabla^2\mathcal{L}(\mathbf{\Omega}_y) = \mathbf{\Omega}_y^{-1} \otimes \mathbf{\Omega}_y^{-1}$ with

$$\phi_{\min}(\nabla^2\mathcal{L}(\mathbf{\Omega}_y)) = \phi^2_{\min}(\mathbf{\Omega}_y^{-1}) = \frac{1}{\phi^2_{\max}(\mathbf{\Omega}_y)} = \frac{1}{\|\mathbf{\Omega}_y\|^2}\,. \tag{140}$$

Since $\nabla^2 h(\mathbf{\Omega}_y) = \mathbf{\Omega}_y^{-1} \otimes \mathbf{\Omega}_y^{-1} - \mu\boldsymbol{I}_{(mp)^2}$, it follows that $h(\mathbf{\Omega}_y)$ is positive semi-definite, hence convex, if

$$\|\mathbf{\Omega}_y\| \le \sqrt{\frac{1}{\mu}}\,. \tag{141}$$

One of the properties of the penalty functions $\rho_\lambda(u)$ is that $g(u) := \rho_\lambda(u) + \frac{\mu}{2}u^2$ is convex for some $\mu \ge 0$, and another property is that $\rho_\lambda(u)$ is non-decreasing on $\mathbb{R}_+$. Therefore, by the composition rules [60, Sec. 3.2.4], $g(|[\mathbf{\Omega}_y]_{ij}|)$ and $g(\|\mathbf{\Omega}_y^{(q\ell)}\|_F)$ are convex. Hence,

$$P_e(\mathbf{\Omega}_y) + \frac{\mu_e}{2}\|\mathbf{\Omega}_y\|_F^2 = \sum_{i\ne j}^{mp}\left(\rho_\lambda(|[\mathbf{\Omega}_y]_{ij}|) + \frac{\mu_e}{2}|[\mathbf{\Omega}_y]_{ij}|^2\right) \tag{142}$$

is convex for $\mu_e = \mu \ge 0$, and similarly,

$$\begin{aligned}P_g(\mathbf{\Omega}_y) + \frac{\mu_g}{2}\|\mathbf{\Omega}_y\|_F^2 &= m\sum_{q\ne\ell}^{p}\Big(\rho_\lambda(\|\mathbf{\Omega}_y^{(q\ell)}\|_F)\\ &\quad + \frac{\mu_g}{2m}\|\mathbf{\Omega}_y^{(q\ell)}\|_F^2\Big)\end{aligned} \tag{143}$$

is convex for $\mu_g = m\,\mu$, where μ is the value that renders $\rho_\lambda(u) + \frac{\mu}{2}u^2$ convex. Now express $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ as

$$\bar{\mathcal{L}}(\mathbf{\Omega}_y) = \alpha\bar{\mathcal{L}}_e(\mathbf{\Omega}_y) + (1-\alpha)\bar{\mathcal{L}}_g(\mathbf{\Omega}_y)\,, \tag{144}$$

$$\bar{\mathcal{L}}_e(\mathbf{\Omega}_y) = \mathcal{L}(\mathbf{\Omega}_y) - \frac{\mu}{2}\|\mathbf{\Omega}_y\|_F^2 + P_e(\mathbf{\Omega}_y) + \frac{\mu}{2}\|\mathbf{\Omega}_y\|_F^2\,, \tag{145}$$

$$\bar{\mathcal{L}}_g(\mathbf{\Omega}_y) = \mathcal{L}(\mathbf{\Omega}_y) - \frac{\mu}{2}\|\mathbf{\Omega}_y\|_F^2 + P_g(\mathbf{\Omega}_y) + \frac{\mu}{2}\|\mathbf{\Omega}_y\|_F^2\,. \tag{146}$$

By (141), (142) and (145), $\bar{\mathcal{L}}_e(\mathbf{\Omega}_y)$ is convex function of $\mathbf{\Omega}_y$ if $\|\mathbf{\Omega}_y\| \le \sqrt{\frac{1}{\mu}}$, and by (141), (143) and (146), $\bar{\mathcal{L}}_g(\mathbf{\Omega}_y)$ is convex in $\mathbf{\Omega}_y$ if $\|\mathbf{\Omega}_y\| \le \sqrt{\frac{1}{\mu_g}} = \sqrt{\frac{1}{m\mu}}$. Thus, for $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ to be strictly convex, using the (minimum) values of μ to make $\rho_\lambda(u) + \frac{\mu}{2}u^2$ convex, we require

$$\|\mathbf{\Omega}_y\| < \sqrt{\frac{1}{m\mu}} = \begin{cases} \infty & : \text{ Lasso} \\ \sqrt{\frac{a-1}{m}} & : \text{ SCAD} \\ \sqrt{\frac{\epsilon}{m\lambda_n}} & : \text{ log-sum} \\ \sqrt{\frac{b}{m}} & : \text{ MCP.} \end{cases} \tag{147}$$

The choice $\|\mathbf{\Omega}_y\| < \sqrt{\frac{1}{m\mu}}$ makes $\mathcal{L}(\mathbf{\Omega}_y) - \frac{\mu}{2}\|\mathbf{\Omega}_y\|_F^2$ positive definite, hence strictly convex. We take $\|\mathbf{\Omega}_y\| = 0.99\sqrt{\frac{1}{m\mu}}$, completing the proof. ■

Proof of Theorem 2: If $1/\beta_{\min} \le 0.99\,\theta$, then $\mathbf{\Omega}_y^\diamond \in \mathcal{B}$ since $\|\mathbf{\Omega}_y^\diamond\| \le 1/\beta_{\min}$ by assumption (A3). To establish that $\hat{\mathbf{\Omega}}_{y\lambda} \in \mathcal{B}$, consider ($\mathbf{\Delta}$ is as in the proof of Theorem 1)

$$\begin{aligned} \|\hat{\mathbf{\Omega}}_{y\lambda}\| &\le \|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\| + \|\mathbf{\Omega}_y^\diamond\| \\ &\le \|\mathbf{\Delta}\| + 1/\beta_{\min} \le \|\mathbf{\Delta}\|_F + 1/\beta_{\min} \\ &\le Rr_n + 1/\beta_{\min}\,. \end{aligned} \tag{148}$$

Therefore, $\hat{\mathbf{\Omega}}_{y\lambda} \in \mathcal{B}$. Thus, both $\hat{\mathbf{\Omega}}_{y\lambda}$ and $\mathbf{\Omega}_y^\diamond$ are feasible. The desired result then follows from Theorem 1 and (local) strict convexity of $\bar{\mathcal{L}}(\mathbf{\Omega}_y)$ over $\mathcal{B}$ implied by Lemma 3. ■

Proof of Theorem 3: We have $\|\hat{\mathbf{\Omega}}_{y\lambda}^{(q\ell)} - (\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F \le \|\hat{\mathbf{\Omega}}_{y\lambda} - \mathbf{\Omega}_y^\diamond\|_F \le \bar{\sigma}_n$ w.h.p. For the edge $\{q,\ell\} \in \mathcal{E}^\diamond$, we have

$$\begin{aligned} \|\hat{\mathbf{\Omega}}_{y\lambda}^{(q\ell)}\|_F &= \|(\mathbf{\Omega}_y^\diamond)^{(q\ell)} + \hat{\mathbf{\Omega}}_{y\lambda}^{(q\ell)} - (\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F \\ &\ge \|(\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F - \|\hat{\mathbf{\Omega}}_{y\lambda}^{(q\ell)} - (\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F \\ &\ge \nu_t - \bar{\sigma}_n \ge 0.6\,\nu_t \quad \text{for} \quad n \ge N_6 \\ &> \gamma_n\,. \end{aligned} \tag{149}$$

Thus, $\mathcal{E}^\diamond \subseteq \hat{\mathcal{E}}$. Now consider the set complements $(\mathcal{E}^\diamond)^c$ and $\hat{\mathcal{E}}^c$. For the edge $\{q,\ell\} \in (\mathcal{E}^\diamond)^c$, $\|(\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F = 0$. For $n \ge N_6$, w.h.p. we have

$$\begin{aligned} \|\hat{\mathbf{\Omega}}y\lambda^{(q\ell)}\|_F &= \|(\mathbf{\Omega}_y^\diamond)^{(q\ell)} + \hat{\mathbf{\Omega}}_{y\lambda}^{(q\ell)} - (\mathbf{\Omega}_y^\diamond)^{(q\ell)}\|_F \\ &\le 0 + \bar{\sigma}_n \le 0.4\,\nu_t < \gamma_n\,, \end{aligned} \tag{150}$$

implying that $\{q,\ell\} \in (\hat{\mathcal{E}}^\diamond)^c$. Thus, $(\mathcal{E}^\diamond)^c \subseteq \hat{\mathcal{E}}^c$, hence $\hat{\mathcal{E}} \subseteq \mathcal{E}^\diamond$, establishing $\hat{\mathcal{E}} = \mathcal{E}^\diamond$. ■

REFERENCES

[1] J. Whittaker, *Graphical Models in Applied Multivariate Statistics*. Hoboken, NJ, USA: Wiley, 1990.

[2] S. L. Lauritzen, *Graphical Models*. London, U.K.: Oxford Univ. Press, 1996.

[3] P. Bühlmann and S. van de Geer, *Statistics for High-Dimensional Data*. Berlin, Germany: Springer, 2011.

[4] M. J. Wainwright, *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge, U.K.: Cambridge Univ. Press, 2019.

[5] R. Dahlhaus, "Graphical interaction models for multivariate time series 1," *Metrika*, vol. 51, no. 2, pp. 157–172, Aug. 2000.

[6] M. Eichler, "Graphical modelling of multivariate time series," *Probab. Theory Rel. Fields*, vol. 153, nos. 1–2, pp. 233–268, Jun. 2012.

[7] P. Danaher, P. Wang, and D. M. Witten, "The joint graphical lasso for inverse covariance estimation across multiple classes," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 76, no. 2, pp. 373–397, Mar. 2014.

[8] D. R. Brillinger, "Remarks concerning graphical models for time series and point processes," *Brazilian Rev. Econometrics*, vol. 16, no. 1, pp. 1–23, Nov. 1996.

[9] A. Jung, G. Hannak, and N. Goertz, "Graphical LASSO based model selection for time series," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1781–1785, Oct. 2015.

[10] J. K. Tugnait, "On sparse high-dimensional graphical model learning for dependent time series," *Signal Process.*, vol. 197, pp. 1–18, Aug. 2022.

[11] J. K. Tugnait, "On conditional independence graph learning from multi-attribute Gaussian dependent time series," *IEEE Open J. Signal Process.*, vol. 6, pp. 705–721, 2025.

[12] J. K. Tugnait, "Sparse-group log-sum penalized graphical model learning for time series," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Singapore, May 2022, pp. 5822–5826.

[13] F. Takens, "Detecting strange attractors in turbulence," *Dyn. Syst. Turbulence*, vol. 898, no. 1, pp. 365–381, 1981.

[14] J. Frank, S. Mannor, and D. Precup, "Activity and gait recognition with time-delay embeddings," in *Proc. 24rth AAAI Conf. Artif. Intell.*, Jul. 2010, pp. 1581–1586.

[15] S. M. Hirsh, S. M. Ichinaga, S. L. Brunton, J. N. Kutz, and B. W. Brunton, "Structured time-delay models for dynamical systems with connections to frenet–serret frame," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 477, no. 2254, Oct. 2021, Art. no. 20210097.

[16] J. R. M. Hosking, "Fractional differencing," *Biometrika*, vol. 68, no. 1, pp. 165–176, Apr. 1981.

[17] I. N. Lobato, "Consistency of the averaged cross-periodogram in long memory series," *J. Time Ser. Anal.*, vol. 18, no. 2, pp. 137–155, Mar. 1997.

[18] J.-M. Bardet, G. Lang, G. Oppenheim, A. Philippe, and M. S. Taqqu, "Generators of long-range dependent processes: A survey," in *Theory and Applications of Long-range Dependence*. Boston, MA, USA: Birkhäuser, 2003, pp. 579–623.

[19] K. Shimotsu, "Gaussian semiparametric estimation of multivariate fractionally integrated processes," *J. Econometrics*, vol. 137, no. 2, pp. 277–310, Apr. 2007.

[20] G. Samorodnitsky, "Long range dependence," *Found. Trends Stochastic Syst.*, vol. 1, no. 3, pp. 163–257, 2007.

[21] S. Kechagias and V. Pipiras, "Definitions and representations of multivariate long-range dependent time series," *J. Time Ser. Anal.*, vol. 36, no. 1, pp. 1–25, Jan. 2015.

[22] V. Pipiras and M. Taqqu, *Long-Range Dependence and Self-Similarity*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[23] R. J. Wolstenholme and A. T. Walden, "An efficient approach to graphical modeling of time series," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3266–3276, Jun. 15, 2015.

[24] D. Schneider-Luftman, "P-value combiners for graphical modelling of EEG data in the frequency domain," *J. Neurosci. Methods*, vol. 271, pp. 92–106, Sep. 2016.

[25] Y. Matsuda, "A test statistic for graphical modelling of multivariate time series," *Biometrika*, vol. 93, no. 2, pp. 399–409, Jun. 2006.

[26] J. K. Tugnait, "Edge exclusion tests for graphical model selection: Complex Gaussian vectors and time series," *IEEE Trans. Signal Process.*, vol. 67, no. 19, pp. 5062–5077, Oct. 1, 2019.

[27] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, pp. 877–905, Jul. 2008.

[28] J. Songsiri, J. Dahl, and L. Vandenberghe, "Graphical models of autoregressive processes," in *Overview of Convex Optimization in Signal Processing and Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2009, pp. 89–116.

[29] M. Eichler, "Graphical modelling of dynamic relationships in multivariate time series," in *Handbook of Time Series Analysis: Recent Theoretical Developments and Applications*. Hoboken, NJ, USA: Wiley, 2006, pp. 335–372.

[30] J. Songsiri and L. Vandenberghe, "Topology selection in graphical models of autoregressive processes," *J. Mach. Learn. Res.*, vol. 11, pp. 2671–2705, Oct. 2010.

[31] E. Avventi, A. G. Lindquist, and B. Wahlberg, "ARMA identification of graphical models," *IEEE Trans. Autom. Control*, vol. 58, no. 5, pp. 1167–1178, May 2013.
[32] D. Alpago, M. Zorzi, and A. Ferrante, "Identification of sparse reciprocal graphical models," *IEEE Control Syst. Lett.*, vol. 2, no. 4, pp. 659–664, Oct. 2018.
[33] J. You, C. Yu, J. Sun, and J. Chen, "Generalized maximum entropy based identification of graphical ARMA models," *Automatica*, vol. 141, Jul. 2022, Art. no. 110319.
[34] C. Lam and J. Fan, "Sparsistency and rates of convergence in large covariance matrix estimation," *Ann. Statist.*, vol. 37, no. 6B, pp. 4254–4278, Dec. 2009.
[35] C.-H. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *Ann. Statist.*, vol. 38, no. 2, pp. 894–942, Apr. 2010.
[36] M. Kolar, H. Liu, and E. P. Xing, "Graph estimation from multi-attribute data," *J. Mach. Learn. Res.*, vol. 15, pp. 1713–1750, Jun. 2014.
[37] P. Ravikumar, M. J. Wainwright, G. Raskutti, and B. Yu, "High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence," *Electron. J. Statist.*, vol. 5, no. none, pp. 935–980, Jan. 2011.
[38] A. J. Rothman, P. J. Bickel, E. Levina, and J. Zhu, "Sparse permutation invariant covariance estimation," *Electron. J. Statist.*, vol. 2, no. none, pp. 494–515, Jan. 2008.
[39] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, Jul. 2008.
[40] T. Cai, W. Liu, and X. Luo, "A constrained ℓ_1 minimization approach to sparse precision matrix estimation," *J. Amer. Stat. Assoc.*, vol. 106, no. 494, pp. 594–607, Jun. 2011.
[41] T. Zhao, H. Liu, K. Roeder, J. Lafferty, and L. Wasserman, "The huge package for high-dimensional undirected graph estimation in R," *J. Mach. Learn. Res.*, vol. 13, pp. 1059–1062, Aug. 2012.
[42] J. K. Tugnait, "Sparse-group lasso for graph learning from multi-attribute data," *IEEE Trans. Signal Process.*, vol. 69, pp. 1771–1786, 2021.
[43] J. K. Tugnait, "Multi-attribute graph estimation with sparse-group non-convex penalties," *IEEE Access*, vol. 13, pp. 80174–80190, 2025.
[44] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *J. Comput. Graph. Statist.*, vol. 22, pp. 231–245, Sep. 2013.
[45] P.-L. Loh and M. J. Wainwright, "Support recovery without incoherence: A case for nonconvex regularization," *Ann. Statist.*, vol. 45, no. 6, pp. 2455–2482, Dec. 2017.
[46] H. Shu and B. Nan, "Estimation of large covariance and precision matrices from temporally dependent observations," *Ann. Statist.*, vol. 47, no. 3, pp. 1321–1350, Jun. 2019.
[47] M. Rudelson and R. Vershynin, "Hanson-wright inequality and sub-Gaussian concentration," *Electron. Commun. Probab.*, vol. 18, no. none, pp. 1–9, Jan. 2013.
[48] X. Chen, M. Xu, and W. B. Wu, "Regularized estimation of linear functionals of precision matrices for high-dimensional time series," *IEEE Trans. Signal Process.*, vol. 64, no. 24, pp. 6459–6470, Dec. 15, 2016.
[49] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its Oracle properties," *J. Amer. Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, Dec. 2001.
[50] J. K. Tugnait, "Graph learning from multivariate dependent time series via a multi-attribute formulation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Singapore, May 2022, pp. 4508–4512.
[51] J. K. Tugnait, "On graphical modeling of high-dimensional long-range dependent time series," in *Proc. 56th Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Oct. 2022, pp. 1098–1102.
[52] P. J. Antsaklis and A. N. Michel, *Linear Systems*. New York, NY, USA: McGraw-Hill, 1997.
[53] S. Basu and G. Michailidis, "Regularized estimation in sparse high-dimensional time series models," *Ann. Statist.*, vol. 43, no. 4, pp. 1535–1567, Aug. 2015.
[54] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 68, no. 1, pp. 49–67, Feb. 2006.
[55] J. Friedman, T. Hastie, and R. Tibshirani, "A note on the group lasso and a sparse group lasso," 2010, *arXiv:1001.0736*.
[56] P. Breheny and J. Huang, "Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors," *Statist. Comput.*, vol. 25, no. 2, pp. 173–187, Mar. 2015.
[57] H. Zou and R. Li, "One-step sparse estimates in nonconcave penalized likelihood models," *Ann. Statist.*, vol. 36, no. 4, pp. 1509–1533, Aug. 2008.
[58] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.
[59] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2011.
[60] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[61] J. Zhang, Y. Yang, and J. Ding, "Information criteria for model selection," *WIREs Comput. Statist.*, vol. 15, no. 5, p. e1607, Oct. 2023.
[62] R. Foygel and M. Drton, "Extended Bayesian information criteria for Gaussian graphical models," in *Proc. NIPS*, 2010, pp. 604–612.



JITENDRA K. TUGNAIT (Life Fellow, IEEE) received the B.Sc. degree (Hons.) in electronics and electrical communication engineering from Punjab Engineering College, Chandigarh, India, in 1971, the M.S. and E.E. degrees from Syracuse University, Syracuse, NY, USA, in 1973 and 1974, respectively, and the Ph.D. degree from the University of Illinois Urbana–Champaign, in 1978, all in electrical engineering.

From 1978 to 1982, he was an Assistant Professor of electrical and computer engineering with the University of Iowa, Iowa City, IA, USA. He was with the Long Range Research Division, Exxon Production Research Company, Houston, TX, USA, from June 1982 to September 1989. In September 1989, he joined the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL, USA, as a Professor, where he is currently a James B. Davis Professor. His current research interests include statistical signal processing and machine learning for signal processing.

Dr. Tugnait has served as an Associate Editor for IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON SIGNAL PROCESSING, IEEE SIGNAL PROCESSING LETTERS, and IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS; a Senior Area Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING; and a Senior Editor for IEEE WIRELESS COMMUNICATIONS LETTERS.