





# Learning Spatiotemporal Graphical Models From Incomplete Observations

Amirhossein Javaheri , Arash Amini , *Senior Member, IEEE*, Farokh Marvasti , *Life Senior Member, IEEE*, and Daniel P. Palomar , *Fellow, IEEE*

**Abstract**—This paper investigates the problem of learning a graphical model from incomplete spatio-temporal measurements. Our purpose is to analyze a time-varying graph signal represented by an incomplete data matrix, the rows and columns of which correspond to spatial and temporal features/measurements of the signal, respectively. In contrast to the conventional approaches which utilize either a directed or an undirected graphical model for data analysis, we propose a compound multi-relational model exploiting both directed and undirected structures. Our approach is based on statistical inference in which a spatio-temporal signal is considered as a random graph process to which we can apply maximum-a-posteriori estimation methods for model identification. We incorporate the Gaussian-Markov random field (GMRF) and the vector auto-regressive (VAR) models to capture both the (undirected) spatial correlations and the (directed) temporal dependencies. We propose an algorithm for joint estimation of the signal and the graphical models, from incomplete measurements. For this purpose, we formulate an optimization problem that we solve using the block successive upperbound minimization (BSUM) method. Our simulation results confirm the efficiency of the proposed method for signal recovery and graph learning.

**Index Terms**—Graph learning, graph signal recovery, incomplete data, Laplacian matrix, time-varying signal, vector auto regressive (VAR), Gaussian Markov random field (GMRF).

## I. INTRODUCTION

GRAPH-STRUCTURED models are widely used in signal processing and machine learning [1], [2], [3]. Indeed, graph signal processing (GSP) is now a trending research topic with many applications in social networks [4], internet and telecommunication networks [5], sensor networks [6], etc.

The primary step in graph signal processing is to find a graphical model that provides an efficient representation of the

signal. An undirected graph is used to illustrate the degree of similarities or mutual correlations (constraints) (e.g., factor graphs [7]), while a directed graph applies for modeling the causal or directional dependencies (e.g. Bayesian or belief networks (BN) [8]). There are a variety of methods which aim to learn the topology/structure of the graph modeling the signal. There is also an important category of methods for recovery of a graph signal from corrupted (noisy and incomplete) measurements. The first category called the *graph learning* methods [9], [10], [11], usually require complete statistics of the data, while in the second class of methods known as *graph signal recovery* methods [12], [13], knowledge of the graphical model describing the data is a postulate. However, in real-world applications, the observed data may be incomplete or prior information of the underlying graph may be unavailable. Therefore, this paper investigates the problem of learning a graph from incomplete measurements which can also be viewed as a blind graph signal recovery problem.

### A. Related Works

There are various approaches to graph learning from data, mainly classified as *undirected* and *directed* methods. The use of stochastic Gaussian Markov random field (GMRF), which is a graphical model for a multivariate Gaussian distribution with Markov property, is quite common to fit an undirected graph to the data. This approach often leads to the penalized log-likelihood estimation of the precision matrix  $\Theta$  as

$$\min_{\Theta \in \Omega_{\Theta}} \text{Tr}(\Theta \mathbf{S}) - \log \det(\Theta) + \alpha h(\Theta), \quad (1)$$

where  $\Omega_{\Theta}$  denotes the feasibility set,  $\mathbf{S}$  represents the statistics of the data (e.g., sample covariance matrix), and  $h(\Theta)$  is a regularization term (e.g., a sparsity promoting penalty function). The method in [14] known as the GLASSO, is an early work in this area which solves (1) for the set of positive definite precision matrices ( $\Omega_{\Theta} = \{\Theta \succ \mathbf{0}\}$ ). This work was later improved in [15] by introducing Laplacian structural constraints to  $\Omega_{\Theta}$ . A general framework for learning different classes of the Laplacian GMRFs (attractive GMRFs with Laplacian as the precision matrix [16]) with different structural constraints is also proposed in [17]. Additionally, a similar estimation approach has been employed for graph learning via a stochastic factor analysis model [18]. There have also been some recent methods that incorporate additional spectral constraints to problem

Manuscript received 21 May 2023; revised 12 November 2023; accepted 29 December 2023. Date of current version 14 March 2024. This work was supported by the Hong Kong GRF under Grant 16207820 and Grant 16206123. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Paolo Di Lorenzo. (*Corresponding author: Arash Amini.*)

Amirhossein Javaheri is with Sharif University of Technology, Tehran 1458889694, Iran, and also with The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong (e-mail: javaheri\_amirhossein@ee.sharif.edu; sajavaheri@connect.ust.hk).

Arash Amini and Farokh Marvasti are with Sharif University of Technology, Tehran 1458889694, Iran (e-mail: aamini@sharif.edu; marvasti@sharif.edu).

Daniel P. Palomar is with The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong (e-mail: palomar@ust.hk).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TSP.2024.3354572>, provided by the authors.

Digital Object Identifier 10.1109/TSP.2024.3354572

(1), to learn specific types of undirected graphs (e.g., regular,  $k$ -component, or bipartite graphs) [19]. The eigenvectors of the sample covariance matrix have been utilized in some works as the nominal eigenspace for graph learning, which is shown to improve the performance when the sample size is limited [20], [21]. Besides the stochastic approaches, there are several non-probabilistic algorithms for undirected graph learning. For instance, a signal-representation perspective for undirected graph learning is proposed in [10], in which, signal smoothness is used as a measure to learn a graph from the data [22]. Some methods also consider learning a parametric dictionary with graph-induced kernels for signal representation [23].

In contrast to the above approaches, an important but limited class of graph learning algorithms aim at learning a directed topology. The majority of these models are based on stochastic modeling. The Granger causality [24] or the structural equation model [25] can be used to formulate the causal (directional) dependencies between the components of the signal. The vector autoregressive (VAR) approach [26], [27] is another well-studied method for directed topology identification, modeling the directions of the temporal dependencies in a time-varying signal (stochastic process). In the VAR( $p$ ) model, the signal vector at time  $t$  ( $\mathbf{x}_t$ ) is stated in terms of a linear combination of the previous samples

$$\mathbf{x}_t = \sum_{i=1}^p \mathbf{A}^{(i)} \mathbf{x}_{t-i} + \epsilon_t, \quad (2)$$

Here,  $\mathbf{A}^{(i)}$  is known as the state evolution matrix of order  $i$ . This matrix is generally not symmetric and can be represented by the weighted adjacency matrix of a (generally) directed graph with possibly negative weights. There are, of course, some works that use undirected graphs to represent symmetric VAR model parameters, e.g., [28]. The signal  $\epsilon_t$ , called the excitation (innovation) process, is temporally white, i.e.,  $\epsilon_t$  is independent of  $\epsilon_{t-i}$  for  $i > 0$ . A similar concept is the family of causal graph processes (CGP) introduced in [29], in which causal relationships are incorporated into a directed graph topology and  $\mathbf{A}^{(i)}$ s are assumed to be polynomial functions of a common matrix  $\mathbf{A}$ . It is very common in the literature to simply assume the covariance matrix of the innovations is diagonal (spatially white). There are, however, some references that consider the excitation process to be non-white. For instance, [30] and [31] utilize a variant of the VAR model, in which the covariance matrix of the innovations represents spatial correlations.

The graph learning methods, both undirected and directed, rely on true statistics or measurements of the data. However, in practice, the data is often contaminated with noise or there may be missing entries in the data due to unavailable measurements. The graph signal recovery methods can restore signals from corrupted observations, exploiting various criteria such as spatio-temporal smoothness [32], total variation [33], [34], bandlimitedness and sparsity in the graph Fourier transform (GFT) domain [13], [35]. A recent work in [36] considers a multi-relational graphical model for probabilistic signal reconstruction using a Gaussian mixture model. There have also been some modern approaches based on graph neural networks

(GNN) [37] for signal recovery. These methods, however, need to know the topology of the underlying graph a priori. Hence, there have been some recent efforts to fill in the gap through simultaneous signal recovery and graph learning. For instance, [38] considers the joint inference of a directed network topology and the graph process in a structural VAR model. The authors in [39] investigate the problem of joint graph Laplacian inference and signal denoising. In [40], the problem of learning a graph from noisy and incomplete measurements (inpainting) is investigated, which employs the total variation metric. An algorithm for joint graph Laplacian estimation and signal denoising has been proposed in [41], which exploits long and short-term characteristics of the data. A recent approach is also proposed in [42] which exploits spatio-temporal smoothness for joint signal recovery and undirected graph learning. All these approaches are either tailored for learning undirected or directed graphs (to model either spatial or temporal dependencies). Nonetheless, learning a single directed/undirected graph model may not be sufficient to capture both the spatial correlations and the temporal dependencies in a spatio-temporal signal.

Thus, in this paper, we address this gap by proposing a joint signal and graph inference method from corrupted measurements of spatio-temporal data, using a multi-relational model composed of both undirected and directed graph structures.

## B. Contributions

The main contributions of this paper are as follows

- 1) We have adopted a maximum-a-posteriori estimation approach to formulate the problem of joint signal recovery and graph learning from noisy and incomplete measurements of spatio-temporal data under a multi-relational graphical model. Our structure is based on a VAR model where a directed graph is used to model the temporal dependencies via the state transition matrix with arbitrary (off-diagonal) entries, and a simple undirected graph is utilized to represent the spatial similarities via a Laplacian GMRF model for the innovations process.
- 2) Our learning algorithm is based on the block successive upperbound minimization (BSUM) approach, also called block majorization minimization (MM), with three sub-problems (update steps) for joint estimation of our proposed model parameters. We also prove the uniqueness of the solution to each subproblem, and subsequently the convergence of the proposed method. The simulation results demonstrate that the proposed method exhibits a faster rate of convergence as well as superior performance in simultaneous graph learning and signal recovery from corrupted data compared to some state-of-the-art methods.

## C. Outline and Notations

This paper is organized as follows. In Section II, some preliminaries and definitions are provided. Section II introduces the proposed model and formulates the problem of learning the model from noisy and incomplete data. We propose an iterative algorithm to solve the problem via three update-steps in

TABLE I  
 LIST OF NOTATIONS USED IN THE PAPER

Notation	Description
$\mathbf{A} / \mathbf{a} / \text{vec}(\mathbf{A})$	a matrix / a vector / vectorized form of a matrix
$\det(\mathbf{L}) /  \mathbf{L} $	determinant / pseudo determinant of $\mathbf{L}$
$\mathcal{L} / \mathcal{L}^*$	Laplacian operator / and its adjoint
$\ \mathbf{A}\ _{1,1} / \ \mathbf{A}\ _{1,\text{off}}$	$\ell_1$ norm of all / off-diagonal elements of $\mathbf{A}$
$\ \mathbf{A}\  / \ \mathbf{A}\ _F$	spectral / Frobenius norm of matrix $\mathbf{A}$
$\text{Tr}(\mathbf{A})$	trace of matrix $\mathbf{A}$
$\text{Diag}(\mathbf{a})$	diagonal matrix with elements from $\mathbf{a}$
$\mathbf{A} \odot \mathbf{B} / \mathbf{A} \otimes \mathbf{B}$	Hadamard / Kronecker product of $\mathbf{A}$ and $\mathbf{B}$
$\mathbf{A}_S$	sub-matrix of $\mathbf{A}$ with column indices in $S$
$D_{\mathbf{X}}f$	directional derivative of $f$ with respect to $\mathbf{X}$

Section IV. Here, we also analyze the conditions for uniqueness of the solutions to each subproblem and the convergence of the proposed method. Finally, the simulation results are given in V.

For clarity of presentation, the list of notations used in this paper is provided in Table I.

## II. PRELIMINARIES

We represent a graph with  $n$  vertices by the triplet  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$ , with  $\mathcal{V} = \{1, \dots, n\}$  being the set of graph vertices, and  $\mathcal{E}$  the set of graph edges; i.e.,  $\mathcal{E} = \{e_1, \dots, e_{|\mathcal{E}|}\} \subseteq \{(i, j) | i, j \in \mathcal{V}\}$ , and a pair  $(i, j) \in \mathcal{E}$  indicates a directed edge starting from vertex  $i$  and ending at vertex  $j$ . For undirected graphs, we simply denote each edge with  $e_k = \{i, j\}$ . The matrix  $\mathbf{W}$  in the graph triplet is the weighted adjacency matrix whose elements represent the weight of the edges; the elements of this matrix are commonly assumed to be non-negative, i.e.,  $W_{i,j} \geq 0$ . Obviously, we expect to have  $W_{i,j} = 0$  if  $(i, j) \notin \mathcal{E}$ . The support of the weight matrix is known as the adjacency or the connectivity matrix  $\mathbf{C} = \mathbb{1}_{\mathbf{W} \neq 0}$ . For a simple undirected graph,  $\mathbf{W}$  is symmetric with zero diagonals, whereas for a directed graph, it is in general, non-symmetric. The Laplacian matrix, or combinatorial graph Laplacian (CGL), for an undirected graph, is defined as  $\mathbf{L} = \mathbf{D} - \mathbf{W}$  where  $\mathbf{D}$  is the diagonal degree matrix of the graph, with  $D_{i,i} = \sum_j W_{i,j}$  being the degree of vertex  $i$ .

Learning a graph is equivalent to learning its Laplacian or weighted adjacency matrix. For a simple undirected graph,  $\mathbf{W}_{n \times n}$  consists of  $n(n-1)/2$  degrees of freedom which we denote by the vector  $\mathbf{w} \in \mathbb{R}^{n(n-1)/2}$ . The relationship between the vector  $\mathbf{w}$  and the Laplacian matrix can be stated as  $\mathbf{L} = \mathcal{L}(\mathbf{w})$  where  $\mathcal{L}: \mathbb{R}^{n(n-1)/2} \rightarrow \mathbb{R}^{n \times n}$  is called the Laplacian operator [19], [21]. An adjoint operator  $\mathcal{L}^*: \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n(n-1)/2}$  may also be defined that satisfies  $\langle \mathcal{L}(\mathbf{w}), \mathbf{M} \rangle = \langle \mathbf{w}, \mathcal{L}^*(\mathbf{M}) \rangle$  for any square matrix  $\mathbf{M}$ . When it comes to learning graphs from data, we often deal with a (two-dimensional) matrix of signal samples denoted by  $\mathbf{X} \in \mathbb{R}^{n \times m}$ . For spatio-temporal data, each column of the matrix represents one realization of a deterministic or a stochastic graph signal (spatial measurements) and each row encodes the variations of the graph signal at a given vertex over time (temporal measurements).

## III. PROBLEM STATEMENT

To capture both temporal and spatial dependencies (correlations) in a time-varying graph signal, we propose a spatio-temporal first-order VAR model as follows:

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \boldsymbol{\epsilon}_t, \quad p(\boldsymbol{\epsilon}_t | \mathbf{L}) \propto |\mathbf{L}|^{\frac{1}{2}} \exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_t^\top \mathbf{L} \boldsymbol{\epsilon}_t\right). \quad (3)$$

Here,  $\mathbf{A}$  represents the state transition matrix of the VAR model parameters, which can have an arbitrary (non-diagonal and asymmetric) structure, and  $\boldsymbol{\epsilon}_t$  is a zero-mean excitation (noise) process. We assume a Laplacian GMRF model for  $\boldsymbol{\epsilon}_t$  characterized with  $\mathbf{L} \succeq 0$  as the precision matrix. The matrix  $\mathbf{L}$  is the Laplacian of a simple undirected graph, which encodes the conditional dependence relations among the variables. In other words, a missing edge between nodes  $i$  and  $j$  indicates that  $\epsilon_{i,t}$  and  $\epsilon_{j,t}$  are independent, given all the other elements of  $\boldsymbol{\epsilon}_t$ . The Laplacian matrix is singular and the term  $|\mathbf{L}|$  denotes the pseudo-determinant (generalized determinant) defined as  $|\mathbf{L}| = \prod_{\lambda_i \neq 0} \lambda_i(\mathbf{L})$ . Throughout this paper, we assume that the undirected graph used in our model is connected and hence,  $\mathbf{L}$  has only one zero eigenvalue, i.e.,  $\text{rank}(\mathbf{L}) = n - 1$ . Thus, we may write  $|\mathbf{L}| = \det(\mathbf{L} + \mathbf{J})$ , where  $\mathbf{J} = (\frac{1}{n})\mathbf{1}\mathbf{1}^\top$ . In practice, the graphs we encounter have few edges and therefore, the matrices  $\mathbf{L}$  and  $\mathbf{A}$  are sparse. To promote sparsity, similar to [17], we assume an exponential prior with parameter  $\alpha_0$  for the edge weights of the undirected graph. Thus, we have the prior distribution  $p(\mathbf{w}) \propto \exp(-2\alpha_0 \mathbf{1}^\top \mathbf{w})$  on  $\mathbf{w} \geq 0$ . Hence,

$$p(\mathbf{L}) \propto \exp(-\alpha_0 \|\mathbf{L}\|_{1,\text{off}}), \quad (4)$$

where  $\|\mathbf{L}\|_{1,\text{off}}$  denotes the  $\ell_1$  norm of the off-diagonal elements of  $\mathbf{L}$ . There is also a division by 2 due to  $\mathbf{L} = \mathbf{L}^\top$ . Since all off-diagonal elements of  $\mathbf{L}$  are non-positive, we can also write  $\|\mathbf{L}\|_{1,\text{off}} = \text{Tr}(\mathbf{L}\mathbf{H}_{\text{off}})$  where  $\mathbf{H}_{\text{off}} = \mathbf{I} - \mathbf{1}\mathbf{1}^\top$ . It should be noted that although the  $\ell_1$ -norm is shown not to be effective in promoting sparsity in learning some types of graphical models [43], [44], our simulation results (presented in Section V-A) demonstrate that we can still apply the  $\ell_1$ -norm in our case, since we further scale and apply a threshold on the inferred Laplacian (weighted adjacency matrix).

Similarly, we assume a Laplace distribution (with parameter  $\alpha_1$ ) for the components of  $\mathbf{A}$ , to promote sparsity. Hence:

$$\log p(\mathbf{A}) = n^2 \log(\alpha_1/2) - \alpha_1 \|\mathbf{A}\|_{1,1}. \quad (5)$$

Now consider the proposed model expressed by (3). In terms of time (variable  $t$ ), the graph signal  $\mathbf{x}_t$  describes a first order Markov process. Similarly, based on the spatial dependencies (vertex domain), the signal defines a GMRF. While  $\mathbf{x}_t$  stands for the signal of interest in this work, we assume to have access to some incomplete noisy measurements  $\mathbf{y}_t$  from  $\mathbf{x}_t$ . Hence, the observations are obtained by

$$\mathbf{y}_t = \mathbf{m}_t \odot (\mathbf{x}_t + \mathbf{n}_t), \quad \mathbf{n}_t \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I}), \quad (6)$$

where the measurement noise component  $\mathbf{n}_t$ , is assumed to be additive and temporally-spatially i.i.d. Gaussian with zero mean



and covariance matrix  $\sigma_n^2 \mathbf{I}$ . Furthermore,  $\mathbf{m}_t$  is a known mask vector defining the sampling pattern of the graph signal at time  $t$ . Therefore, by horizontal concatenation of the vectors from  $t = 1$  to  $t = T$  (left to right), we obtain the following compact matrix form (where  $\mathbf{Y}_{i,t} = \mathbf{y}_t(i)$ ):

$$\mathbf{Y} = \mathbf{M} \odot (\mathbf{X} + \mathbf{N}) \quad (7)$$

Now, the problem considered in this paper is as follows: by observing the measurement matrix  $\mathbf{Y}$  and the sampling masks  $\mathbf{M}$ , we would like to estimate the VAR model parameters  $\mathbf{A}$ , the graph Laplacian  $\mathbf{L}$ , and the time snapshots of the graph signal  $\mathbf{X}$ . Let  $\Omega_{\mathbf{L}} = \{\mathbf{L} \succeq 0 \mid L_{ij} = L_{ji} \leq 0 \ (i \neq j), \mathbf{L} \cdot \mathbf{1} = \mathbf{0}, \text{rank}(\mathbf{L}) = n - 1\}$  be the set of feasible Laplacian matrices. We then employ the maximum-a-posteriori (MAP) rule to estimate  $\mathbf{X}$ ,  $\mathbf{L}$ , and  $\mathbf{A}$  by knowing  $\mathbf{M}$  and observing  $\mathbf{Y}$  in (7):

$$\begin{aligned} \mathbf{X}^*, \mathbf{L}^*, \mathbf{A}^* &= \underset{\mathbf{X}, \mathbf{L} \in \Omega_{\mathbf{L}}, \mathbf{A}}{\operatorname{argmax}} p(\mathbf{X}, \mathbf{L}, \mathbf{A} | \mathbf{Y}, \mathbf{M}) \\ &= \underset{\mathbf{X}, \mathbf{L} \in \Omega_{\mathbf{L}}, \mathbf{A}}{\operatorname{argmin}} \{-\log p(\mathbf{Y} | \mathbf{M}, \mathbf{X}, \mathbf{L}, \mathbf{A}) \\ &\quad - \log p(\mathbf{X} | \mathbf{L}, \mathbf{A}) - \log p(\mathbf{A}) - \log p(\mathbf{L})\}, \end{aligned} \quad (8)$$

where in the last equation, we have assumed  $\mathbf{A}$  to be independent from  $\mathbf{L}$ ;  $\mathbf{M}$  is also a fixed and known matrix. Because of the Markovian property of the proposed VAR model (in time-domain), starting from  $\mathbf{x}_0 = \mathbf{0}$  we have:

$$\begin{aligned} \log p(\mathbf{X} | \mathbf{A}, \mathbf{L}) &= \log \prod_{t=1}^T p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{A}, \mathbf{L}) \\ &= \sum_{t=1}^T \log p(\boldsymbol{\epsilon}_t | \mathbf{A}, \mathbf{L}) \\ &= C_0 + \frac{T}{2} \log \det(\mathbf{L} + \mathbf{J}) - \frac{1}{2} \sum_{t=1}^T \boldsymbol{\epsilon}_t^\top \mathbf{L} \boldsymbol{\epsilon}_t, \end{aligned}$$

where  $C_0 = -(n-1)/2 \log(2\pi)$ . By denoting the  $t$ -th canonical basis as  $\mathbf{e}_t$  with the convention  $\mathbf{e}_0 = \mathbf{0}$ , we can define  $\boldsymbol{\epsilon}_t = \mathbf{E} \mathbf{e}_t$ , and proceed as

$$\log p(\mathbf{X} | \mathbf{A}, \mathbf{L}) = C_0 + \frac{T}{2} \log \det(\mathbf{L} + \mathbf{J}) - \frac{1}{2} \operatorname{Tr}(\mathbf{L} \mathbf{E} \mathbf{E}^\top),$$

where

$$\mathbf{E} = \mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D}, \quad \mathbf{D}_T = \sum_{t=1}^T \mathbf{e}_{t-1} \mathbf{e}_t^\top. \quad (9)$$

Since  $\mathbf{Y}$  is independent of  $\mathbf{A}$  and  $\mathbf{L}$  given  $\mathbf{X}$ , we may write:

$$\begin{aligned} \log p(\mathbf{Y} | \mathbf{X}, \mathbf{L}, \mathbf{A}, \mathbf{M}) &= \log p(\mathbf{Y} | \mathbf{X}, \mathbf{M}) \\ &= C_1 - \frac{1}{2\sigma_n^2} \sum_{t=1}^T \|\mathbf{m}_t \odot (\mathbf{y}_t - \mathbf{x}_t)\|^2 \\ &= C_1 - \frac{1}{2\sigma_n^2} \|\mathbf{M} \odot \mathbf{Y} - \mathbf{M} \odot \mathbf{X}\|_F^2, \end{aligned} \quad (10)$$

where  $C_1$  is another constant. Now, using the Laplacian operator [19], we can restate  $\mathbf{L}$  as  $\mathbf{L} = \mathcal{L}(\mathbf{w})$ . Hence, by simple

steps, the MAP problem can be restated in terms of  $\mathbf{X}$ ,  $\mathbf{A}$ , and  $\mathbf{w}$  as follows

$$\begin{aligned} \mathbf{X}^*, \mathbf{A}^*, \mathbf{w}^* &= \underset{\mathbf{X}, \mathbf{A}, \mathbf{w}}{\operatorname{argmin}} f(\mathbf{X}, \mathbf{A}, \mathbf{w}), \quad \text{s.t. } \mathcal{L}(\mathbf{w}) \in \Omega_{\mathbf{L}} \\ f(\mathbf{X}, \mathbf{A}, \mathbf{w}) &\triangleq \frac{1}{\sigma_n^2} \|\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X}\|_F^2 + 2\alpha_1 \|\mathbf{A}\|_{1,1} \\ &\quad + \operatorname{Tr}(\mathcal{L}(\mathbf{w})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})^\top) \\ &\quad - T \log \det(\mathcal{L}(\mathbf{w}) + \mathbf{J}) + 2\alpha_0 \operatorname{Tr}(\mathcal{L}(\mathbf{w}) \mathbf{H}_{\text{off}}), \end{aligned} \quad (11)$$

where  $\mathbf{Y}_M = \mathbf{M} \odot \mathbf{Y}$ .

#### IV. PROPOSED ALGORITHM

The cost function in (11) is convex with respect to each block variable  $\mathbf{X}$ ,  $\mathbf{A}$ , and  $\mathbf{w}$ . Hence, we can use the block successive upperbound minimization (BSUM) [45] or the block MM [46] method to minimize the proposed objective function. The BSUM algorithm is indeed a generalization of the block-coordinate-descent (BCD) or the Gauss-Seidel method [47], which minimizes an upperbound of the original cost in each iteration. This way, starting from  $(\mathbf{X}^{(0)}, \mathbf{A}^{(0)}, \mathbf{w}^{(0)})$ , for  $j \geq 0$  we have the following iterations:

$$\begin{aligned} \mathbf{X}^{(j+1)} &= \underset{\mathbf{X}}{\operatorname{argmin}} f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}^{(j)}, \mathbf{A}^{(j)}, \mathbf{w}^{(j)}), \\ \mathbf{A}^{(j+1)} &= \underset{\mathbf{A}}{\operatorname{argmin}} f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{X}^{(j+1)}, \mathbf{A}^{(j)}, \mathbf{w}^{(j)}), \\ \mathbf{w}^{(j+1)} &= \underset{\mathbf{w}}{\operatorname{argmin}} f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{X}^{(j+1)}, \mathbf{A}^{(j+1)}, \mathbf{w}^{(j)}), \end{aligned} \quad (12)$$

where  $f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}^{(j)}, \mathbf{A}^{(j)}, \mathbf{w}^{(j)})$  is a convex upperbound or a majorization (as defined in [45]) for the function  $f_{\mathbf{X}}(\mathbf{X}) \triangleq f(\mathbf{X}, \mathbf{A} = \mathbf{A}^{(j)}, \mathbf{w} = \mathbf{w}^{(j)})$ . This majorizer also uses  $\mathbf{X}^{(j)}$  as a parameter. Similarly,  $f_{\mathbf{A}}^S$  and  $f_{\mathbf{w}}^S$  are convex functions of  $\mathbf{A}$  and  $\mathbf{w}$ , that majorize  $f_{\mathbf{w}}(\mathbf{w}) \triangleq f(\mathbf{X} = \mathbf{X}^{(j+1)}, \mathbf{A}, \mathbf{w} = \mathbf{w}^{(j)})$  and  $f_{\mathbf{A}}(\mathbf{A}) \triangleq f(\mathbf{X} = \mathbf{X}^{(j+1)}, \mathbf{A} = \mathbf{A}^{(j+1)}, \mathbf{w})$  respectively. To simplify the notations, from now on, we drop the super-scripts  $(j)$  and  $(j+1)$  from constant variables. Below, we separately study the updating rules for  $\mathbf{X}$ ,  $\mathbf{A}$ , and  $\mathbf{w}$ .

##### A. $\mathbf{X}$ -Subproblem

Assuming  $\mathbf{A}$  and  $\mathbf{w}$  to be fixed, the function in (11) can be restated in terms of  $\mathbf{X}$  as follows

$$\begin{aligned} f_{\mathbf{X}}(\mathbf{X}) &= \frac{1}{\sigma_n^2} \operatorname{Tr}((\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X})(\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X})^\top) \\ &\quad + \operatorname{Tr}(\mathcal{L}(\mathbf{w})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})^\top) + \text{const.} \end{aligned} \quad (13)$$

This function is convex with respect to  $\mathbf{X}$  as stated in the following theorem.

*Proposition 1:* The function  $f_{\mathbf{X}}(\mathbf{X})$  defined in (13) is convex with respect to  $\mathbf{X}$ . Furthermore, if  $\sigma_n < \infty$  and  $\sum_k M_{k,i} > 0, \forall i$  (i.e., there is at least one observation sample at each time snapshot), the function is strictly convex with a unique minimizer given by the following closed-form solution

$$\mathbf{X}^* = \operatorname{vec}^{-1}(\mathbf{G}^{-1} \mathbf{b}), \quad (14)$$

where

$$\begin{aligned}\mathbf{G} &= \frac{1}{\sigma_n^2} \text{Diag}(\text{vec}(\mathbf{M})) + \mathbf{H}^\top (\mathbf{I}_T \otimes \mathcal{L}(\mathbf{w})) \mathbf{H}, \\ \mathbf{H} &= \mathbf{I}_{nT} - \mathbf{D}^\top \otimes \mathbf{A}, \\ \mathbf{b} &= \frac{1}{\sigma_n^2} (\text{vec}(\mathbf{Y}_M)).\end{aligned}\quad (15)$$

*Proof:* Refer to Appendix A.  $\square$

Although  $f_{\mathbf{X}}(\mathbf{X})$  is convex and has a closed-form minimizer if  $\mathbf{G}$  is invertible, the computational complexity of inverting  $\mathbf{G}_{nT \times nT}$  is problematic in practice. Therefore, we introduce a motorize to reduce the computational load which can also be utilized in the BSUM algorithm.

*Lemma 1:* The function  $f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}_0)$  as

$$\begin{aligned}f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}_0) &= f_{\mathbf{X}}(\mathbf{X}) \\ &+ \text{vec}(\mathbf{X} - \mathbf{X}_0)^\top (\theta \mathbf{I} - \mathbf{G}) \text{vec}(\mathbf{X} - \mathbf{X}_0)\end{aligned}\quad (16)$$

defines a strictly convex majorization for  $f_{\mathbf{X}}(\mathbf{X})$  if  $\theta > \hat{\theta}_{\min} \triangleq \frac{1}{\sigma_n^2} + 2 \|\mathcal{L}(\mathbf{w})\| (1 + \|\mathbf{A}\|^2)$ .

*Proof:* Refer to Appendix B.  $\square$

Using this Lemma, the  $\mathbf{X}$ -update step in the BSUM algorithm can be obtained by setting  $\mathbf{X}_0 = \mathbf{X}^{(j)}$  as follows

$$\begin{aligned}\mathbf{X}^{(j+1)} &= \underset{\mathbf{X}}{\text{argmin}} f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}^{(j)}) \\ &= \mathbf{X}^{(j)} - \frac{1}{2\theta} \frac{\partial}{\partial \mathbf{X}} f_{\mathbf{X}}(\mathbf{X})|_{\mathbf{X}^{(j)}},\end{aligned}\quad (17)$$

where

$$\frac{\partial}{\partial \mathbf{X}} f_{\mathbf{X}}(\mathbf{X}) = 2 \left( \mathcal{L}(\mathbf{w}) \mathbf{E} - \mathbf{A}^\top \mathcal{L}(\mathbf{w}) \mathbf{E} \mathbf{D}^\top - \frac{\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X}}{\sigma_n^2} \right)$$

and  $\mathbf{E} = \mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D}$ . This is actually a gradient descent (GD) step with adaptive step-size.

### B. $\mathbf{A}$ -Subproblem

With  $\mathbf{X}$  and  $\mathbf{w}$  fixed, the cost function reduces to

$$\begin{aligned}f_{\mathbf{A}}(\mathbf{A}) &\triangleq \text{Tr}(\mathcal{L}(\mathbf{w})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})^\top) \\ &+ 2\alpha_1 \|\mathbf{A}\|_{1,1}.\end{aligned}\quad (18)$$

*Proposition 2:*  $f_{\mathbf{A}}(\mathbf{A})$  is convex with respect to  $\mathbf{A}$ , and the minimizer  $\mathbf{a}^* = \text{vec}(\mathbf{A}^*)$  of  $f_{\mathbf{A}}(\mathbf{A})$  is unique if the submatrix  $\mathbf{R}_{\mathcal{E}}$  is full column rank, where the equi-correlation set  $\mathcal{E}$  is

$$\mathcal{E} = \{1 \leq i \leq n : |\langle \mathbf{r}_i, \mathbf{d} - \mathbf{R} \mathbf{a}^* \rangle| = \alpha_1\},\quad (19)$$

$\mathbf{r}_i$  is the  $i$ th column of  $\mathbf{R} = (\mathbf{X} \mathbf{D})^\top \otimes \mathcal{L}(\mathbf{w})^{1/2}$ , and  $\mathbf{d} = \text{vec}(\mathcal{L}(\mathbf{w})^{1/2} \mathbf{X})$ .

*Proof:* Refer to Appendix C.  $\square$

The next step is to find a suitable majorization function  $f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0)$  for  $f_{\mathbf{A}}(\mathbf{A})$  that could be simply minimized.

*Lemma 2:* For any  $\mathbf{A}_0$  and  $\beta > \beta_{\min} = \|\mathbf{X} \mathbf{D}\|^2 \|\mathcal{L}(\mathbf{w})\|$ , the function  $f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0)$  defined below is a strictly convex majorizer for  $f_{\mathbf{A}}(\mathbf{A})$ .

$$\begin{aligned}f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0) &= f_{\mathbf{A}}(\mathbf{A}) + \beta \|\mathbf{A} - \mathbf{A}_0\|_F^2 \\ &- \text{Tr}(\mathcal{L}(\mathbf{w})(\mathbf{A} - \mathbf{A}_0) \mathbf{X} \mathbf{D} \mathbf{D}^\top \mathbf{X}^\top (\mathbf{A} - \mathbf{A}_0)^\top).\end{aligned}\quad (20)$$

*Proof:* Refer to Appendix D.  $\square$

We can simplify  $f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0)$  as follows

$$\begin{aligned}f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0) &= 2\alpha_1 \|\mathbf{A}\|_{1,1} \\ &+ \beta \left\| \mathbf{A} - \left( \mathbf{A}_0 - \frac{1}{\beta} \mathcal{L}(\mathbf{w})(\mathbf{A}_0 \mathbf{X} \mathbf{D} - \mathbf{X})(\mathbf{X} \mathbf{D})^\top \right) \right\|_F^2 + \text{const},\end{aligned}$$

where const is a term that does not involve  $\mathbf{A}$ . Hence, setting  $\mathbf{A}_0 = \mathbf{A}^{(j)}$ , a closed-form solution yields for the  $\mathbf{A}$  update step in the BSUM algorithm using the soft-thresholding operator  $\mathcal{S}$  [48] as

$$\begin{aligned}\mathbf{A}^{(j+1)} &= \underset{\mathbf{A}}{\text{argmin}} f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}^{(j)}) \\ &= \mathcal{S}_{\alpha_1/\beta} \left( \mathbf{A}^{(j)} - \frac{1}{\beta} \mathcal{L}(\mathbf{w})(\mathbf{A}^{(j)} \mathbf{X} \mathbf{D} - \mathbf{X})(\mathbf{X} \mathbf{D})^\top \right).\end{aligned}\quad (21)$$

### C. $\mathbf{w}$ -Subproblem

Assuming  $\mathbf{X}$  and  $\mathbf{A}$  to be fixed, we need to solve the following problem for the  $\mathbf{w}$ -update step,

$$\mathbf{w}^* = \underset{\mathbf{w} \in \Omega_{\mathbf{L}}}{\text{argmin}} f_{\mathbf{w}}(\mathbf{w})\quad (22)$$

where

$$f_{\mathbf{w}}(\mathbf{w}) = \text{Tr}(\mathcal{L}(\mathbf{w}) \mathbf{K}) - \log \det(\mathcal{L}(\mathbf{w}) + \mathbf{J})\quad (23)$$

and  $\mathbf{K} = \frac{1}{T} ((\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})^\top + 2\alpha_0 \mathbf{H}_{\text{off}})$ . Now, according to the following proposition, we may restate problem (22) as follows

$$\mathbf{w}^* = \underset{\mathbf{w} \in \Omega_{\mathbf{w}}}{\text{argmin}} f_{\mathbf{w}}(\mathbf{w}), \quad \Omega_{\mathbf{w}} = \{\mathbf{w} | \mathbf{w} \geq 0\}.\quad (24)$$

*Proposition 3:* The optimization problem in (22) is convex and equivalent to (24) if and only if  $\Omega_{\mathbf{w}} = \{\mathbf{w} | \mathbf{w} \geq 0\}$ . Furthermore, (24) has a unique minimizer if  $\mathbf{K} \succ 0$  or  $\alpha_0 > 0$ .

*Proof:* See Appendix E for the proof.  $\square$

*Lemma 3:* Assume  $\mathbf{w}_0 \geq 0$  and  $\tau > 0$  to be constant. Also define  $\mathbf{q} = \mathcal{L}^* ((\mathcal{L}(\mathbf{w}_0) + \mathbf{J})^{-1})$  and  $\mathbf{r} = \mathcal{L}^*(\mathbf{K})$ . A strictly convex majorization function for  $f_{\mathbf{w}}(\mathbf{w})$  denoted by  $f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{w}_0)$  can be obtained as

$$\begin{aligned}f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{w}_0) &= \tau (\mathbf{q} \odot \mathbf{w}_0^2, \mathbf{w} \odot \mathbf{w}_0 + (\mathbf{w}_0 + 1/\tau) \odot \mathbf{w} - 2) \\ &+ \langle \mathbf{w}, \mathbf{r} \rangle + \text{Tr}((\mathcal{L}(\mathbf{w}_0) + \mathbf{J})^{-1} \mathbf{J}) \\ &- \log \det(\mathcal{L}(\mathbf{w}_0) + \mathbf{J}) - n,\end{aligned}\quad (25)$$

where  $\odot$  and  $\oslash$ , respectively denote the element-wise power and division.

*Proof:* Refer to Appendix F for the proof.  $\square$

Finally, the BSUM step corresponding to  $\mathbf{w}$ -update is

$$\begin{aligned}\mathbf{w}^{(j+1)} &= \underset{\mathbf{w}}{\text{argmin}} f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{w}^{(j)}) \\ &= \mathbf{w}^{(j)} \odot \sqrt{(\tau \mathbf{w}^{(j)} \odot \mathbf{q} + \mathbf{q}) \oslash (\tau \mathbf{w}^{(j)} \odot \mathbf{q} + \mathbf{r})}.\end{aligned}\quad (26)$$

**Algorithm 1** The STSRGL (proposed) algorithm to solve (11) based on the BSUM method

**Input:**  $\mathbf{Y}$ ,  $\mathbf{M}$ , **Parameters:**  $\sigma_n$ ,  $\alpha_0$ ,  $\alpha_1$ ,  $\tau$ ,  
**Output:**  $\mathbf{X}^{(t)}$ ,  $\mathbf{A}^{(t)}$ ,  $\mathbf{L}^{(t)} = \mathcal{L}(\mathbf{w}^{(t)})$ .  
**Initialization:**  $\mathbf{X}^{(0)} = \mathbf{Y}$ ,  $\mathbf{A}^{(0)} = \mathbf{0}$ ,  $\mathbf{w}^{(0)} = \mathcal{P}_{\mathbf{w} \geq \mathbf{0}}(\mathbf{S}_Y^\dagger)$ ,  
 $j = 0$   
**repeat**  
  Obtain  $\mathbf{X}^{(j+1)}$  using (17) with  $(\mathbf{A}, \mathbf{w}) = (\mathbf{A}^{(j)}, \mathbf{w}^{(j)})$ .  
  Obtain  $\mathbf{A}^{(j+1)}$  via (21) with  $(\mathbf{X}, \mathbf{w}) = (\mathbf{X}^{(j+1)}, \mathbf{w}^{(j)})$ .  
  Obtain  $\mathbf{w}^{(j+1)}$  via (26) with  $(\mathbf{X}, \mathbf{A}) = (\mathbf{X}^{(j+1)}, \mathbf{A}^{(j+1)})$ .  
  Set  $j \leftarrow j + 1$   
**until** A stopping criterion is satisfied

#### D. The Overall Algorithm

Now, that we have obtained solutions to each subproblem of the BSUM algorithm, we can summarize the proposed method, called spatio-temporal signal recovery and graph learning (STSRGL), in Algorithm 1. For the initialization, we choose  $\mathbf{X}^{(0)} = \mathbf{Y}$ ,  $\mathbf{A}^{(0)} = \mathbf{0}$ , and  $\mathbf{w}^{(0)} = \mathcal{P}_{\mathbf{w} \geq \mathbf{0}}(\mathbf{S}_Y^\dagger)$ , where  $\mathbf{S}_Y = \frac{1}{T} \mathbf{Y} \mathbf{Y}^\top$ . The stopping condition is satisfied if the relative error between consecutive iterates falls below a threshold called the error tolerance, or the maximum number of iterations is reached.

#### E. Computational Complexity

The computational complexity of the  $\mathbf{X}$ -update step in (17) is  $\mathcal{O}(n^2T + T^2n + n^3)$  including the computation of  $\theta > \theta_{\min}$ . The update-step for  $\mathbf{A}$  (21) is similarly  $\mathcal{O}(n^2T + T^2n + n^3)$  complex and  $\mathcal{O}(n^3)$  operations are also needed to obtain  $\beta > \beta_{\min}$ . Moreover, given  $\mathbf{K}$ , the computational complexity of the  $\mathbf{w}$  update step is dominated by the evaluation of  $(\mathcal{L}(\mathbf{w}^{(k)}) + \mathbf{J})^{-1}$ , which requires  $\mathcal{O}(n^3)$  operations (the complexity of the remaining operations is  $\mathcal{O}(n^2)$ ). It also requires  $\mathcal{O}(n^2T + T^2n)$  in general to obtain  $\mathbf{K}$ . Thus, the overall complexity of each iteration of Algorithm 1 is  $\mathcal{O}(n^3 + n^2T + T^2n)$ . This may not be scalable to very large graphs. However, if we can cluster the nodes into several disjoint classes, then, the algorithm can be more efficiently applied locally to infer the graph and the signal within each cluster.

*Theorem 1:* For  $\alpha_0 > 0$ , the proposed method given in Algorithm 1 converges to a stationary point of problem (11).

*Proof:* As stated in Lemmas 1, 2, and 3, each subproblem of the BSUM algorithm is strictly convex (quasi-convex) and admits a unique minimizer. Moreover, we have

$$f(\mathbf{X}, \mathbf{A}, \mathbf{w}) = \frac{1}{\sigma_n^2} \|\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X}\|_F^2 + 2\alpha_1 \|\mathbf{A}\|_{1,1} + T f_{\mathbf{w}}(\mathbf{w}),$$

where  $f_{\mathbf{w}}(\mathbf{w})$  is defined in (23). Using Proposition 3 for  $\alpha_0 > 0$ , we conclude that  $f(\mathbf{X}, \mathbf{A}, \mathbf{w})$  in (11) is lower-bounded by  $Th(z^*)$  with  $h(z)$  defined in (33). This function is also continuous in  $\mathbf{X}$ ,  $\mathbf{A}$ , and  $\mathbf{w} \geq \mathbf{0}$  and thus, the sub-level sets  $\{(\mathbf{X}, \mathbf{A}, \mathbf{w}) | f(\mathbf{X}, \mathbf{A}, \mathbf{w}) \leq f_0\}$  are compact for any  $f_0$ . Hence,

using Theorem 2 in [45], it is concluded that the proposed algorithm converges to the set of coordinate-wise minima of (11). That means the directional derivatives along any feasible block direction  $\mathbf{z}_0 = (\mathbf{X}_0, \mathbf{A}_0, \mathbf{w}_0)$ , at any limit point, i.e.,  $D_{\mathbf{X}_0} f(\hat{\mathbf{z}})$ ,  $D_{\mathbf{A}_0} f(\hat{\mathbf{z}})$ , and  $D_{\mathbf{w}_0} f(\hat{\mathbf{z}})$  are all non-negative, where  $\hat{\mathbf{z}}$  denotes an arbitrary limit point. Now, we may write

$$f(\mathbf{X}, \mathbf{A}, \mathbf{w}) = h_{\mathbf{X}}(\mathbf{X}) + h_{\mathbf{A}}(\mathbf{A}) + h_{\mathbf{w}}(\mathbf{w}) + g(\mathbf{X}, \mathbf{A}, \mathbf{w}),$$

with  $h_{\mathbf{X}}(\mathbf{X}) = \frac{1}{\sigma_n^2} \|\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X}\|_F^2$ ,  $h_{\mathbf{A}}(\mathbf{A}) = 2\alpha_1 \|\mathbf{A}\|_{1,1}$ ,  $h_{\mathbf{w}}(\mathbf{w}) = -T \log \det(\mathcal{L}(\mathbf{w}) + \mathbf{J}) + 2\alpha_0 \text{Tr}(\mathcal{L}(\mathbf{w}) \mathbf{H}_{\text{off}})$ , and  $g(\mathbf{X}, \mathbf{A}, \mathbf{w}) = \text{Tr}(\mathcal{L}(\mathbf{w})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})(\mathbf{X} - \mathbf{A} \mathbf{X} \mathbf{D})^\top)$ .

Since  $g(\cdot)$  is smooth, one can show that

$$\begin{aligned} D_{\mathbf{z}_0} f(\hat{\mathbf{z}}) &= \lim_{\mu \rightarrow 0^+} \frac{f(\hat{\mathbf{z}} + \mu \mathbf{z}_0) - f(\hat{\mathbf{z}})}{\mu} \\ &= D_{\mathbf{X}_0} f(\hat{\mathbf{z}}) + D_{\mathbf{A}_0} f(\hat{\mathbf{z}}) + D_{\mathbf{w}_0} f(\hat{\mathbf{z}}) \geq 0. \end{aligned}$$

Hence, it is concluded that  $f$  is regular at any coordinate-wise minimum, implying that every limit-point is also stationary [45].  $\square$

## V. SIMULATION RESULTS

In this section, we present the simulation results of the proposed algorithm for inference in graphical models and recovery of corrupted (noisy and incomplete) observations for both synthetic and real data. In Sec. V-A, we compare the proposed method with some of the state-of-the-art methods in the literature with synthetic data. The results on real data shall be presented in Sec. V-B.

#### A. Synthetic Data

We set  $n = 100$  and  $T = 1000$  in our experiments and generate synthetic data: we construct the Laplacian matrix  $\mathbf{L}$  using the Stochastic Block Model (also known as the Modular Graph) and generate random samples of the  $\epsilon_t$  signal according to a multivariate Gaussian distribution (GMRF). In this undirected graphical model, the nodes are divided into a number of clusters/blocks (4 clusters in our case), and the edges are formed by setting inter-cluster and the intra-cluster probabilities. Finally, the generated Laplacian matrix is uniformly scaled such that  $\text{Tr}(\mathbf{L}) = n$ . We generate random samples  $\epsilon_t$  via

$$\epsilon_t = \sqrt{\mathbf{L}^\dagger} \boldsymbol{\nu}_t, \quad \boldsymbol{\nu}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

where  $\mathbf{L}^\dagger$  represents the pseudo-inverse of the Laplacian matrix.

Next, we use the Laplace distribution to generate the elements of the state transition matrix  $\mathbf{A}$  in equation (3), and then use the adjacency matrix of the undirected graph in our model to mask the components of  $\mathbf{A}$ . Finally, we divide the matrix by its operator norm. Now by generating time samples  $\mathbf{x}_t$  (initialized with  $\mathbf{x}_0 = \mathbf{0}$ ) according to equation (3), the matrix of the original data  $\mathbf{X}$  is constructed by column-wise concatenation of the vectors  $\mathbf{x}_t$  from  $t = 1$  to  $t = T$ . We further normalize the data matrix (each row is centralized and scaled by its standard deviation). The final measurements are constructed by  $\mathbf{Y} = \mathbf{M} \odot (\mathbf{X} + \mathbf{N})$ , where  $\mathbf{M}$  is a random binary sampling matrix (mask) following rate parameter SR and  $\mathbf{N}$  is the noise

matrix filled with i.i.d. Gaussian random variables with zero mean and variance  $\epsilon_n$ . Indeed, the matrices  $\mathbf{Y}$  and  $\mathbf{M}$  are the inputs for learning the graphical models and recovering the data matrix.

To examine the graph learning performance, we measure the relative error and the F-score criteria for the Laplacian and the state transition matrices. If  $\mathbf{B}^* \in \mathbb{R}^{n \times n}$  is either the Laplacian or the state transition of a graph, and  $\hat{\mathbf{B}} \in \mathbb{R}^{n \times n}$  represents its estimated version, the relative error and the F-score values are given by

$$\text{RelErr} = \frac{\|\mathbf{B}^* - \hat{\mathbf{B}}\|_F}{\|\mathbf{B}^*\|_F}, \quad \text{F-score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}.$$

In the above equation, TP, FP, and FN stand for the number of connections in the original graph that are correctly detected, the number of connections in the estimated graph that do not exist in the original one, and the number of connections from the original graph missing in the estimated one, respectively.

We also use SNR and NMSE criteria to measure the signal recovery performance. If we denote the ground-truth data matrix with  $\mathbf{X}^* \in \mathbb{R}^{n \times T}$ , its recovered/estimated version by  $\hat{\mathbf{X}} \in \mathbb{R}^{n \times T}$  and their  $i$ th columns by  $\mathbf{x}_i^*$  and  $\hat{\mathbf{x}}_i$ , respectively, then, we have

$$\text{SNR} = 20 \log_{10} \left( \frac{\|\mathbf{X}^*\|_F}{\|\mathbf{X}^* - \hat{\mathbf{X}}\|_F} \right), \quad \text{NMSE} = \frac{1}{T} \sum_{i=1}^T \frac{\|\mathbf{x}_i^* - \hat{\mathbf{x}}_i\|^2}{\|\mathbf{x}_i^*\|^2}.$$

To provide more robustness, we average the results over 50 different random realizations. In addition, we separately report the results of the Laplacian matrix inference, the state transition matrix inference, and the signal recovery. In the following simulations, we have chosen  $\tau = 50$ ,  $\alpha_0 = 0.1T$ ,  $\alpha_1 = 20$ , and  $\sigma_n = 0.1$ .

1) *Inference of the Laplacian Matrix:* In this subsection, we present the results of the Laplacian matrix inference from noisy and incomplete data. For this purpose, we compare the simulation results of the proposed algorithm with several state-of-the-art undirected graph learning methods, consisting of CGL<sup>1</sup> [17], the GSP-Log and GSP-L2 versions of the GSP toolbox<sup>2</sup> [22], the GLE-MM method in [21], the nonconvex graph learning method introduced in [44], named NGL<sup>3</sup>, and the GL-LRSS method [41] for joint signal and graph Laplacian inference based on low-rank and spatio-temporal smoothness. For the sake of comparison, we include two versions of the proposed algorithm: in the first one, we only solve the Laplacian estimation subproblem with  $\mathbf{A} = \mathbf{0}$  and  $\mathbf{X} = \mathbf{Y}$  (referred to as STSRGL L-sub), while in the second one, we run the complete BSUM method with all the steps according to Algorithm 1 (referred to as STSRGL). For a fair comparison, we scale the output Laplacian matrices obtained by all the algorithms in order to have  $\text{Tr}(\mathbf{L}) = n$ . We then, eliminate edges with weights below a threshold value.

Fig. 1, represents the results of the Laplacian matrix estimation in the Stochastic-Block model in terms of RelErr and

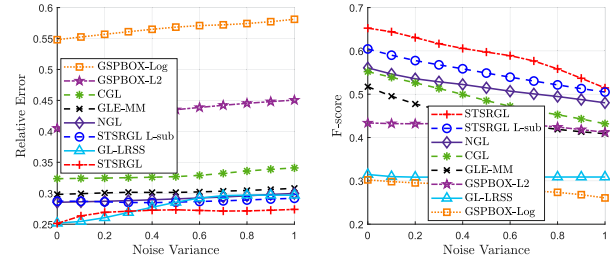


Fig. 1. Relative error and F-score performance of the Laplacian matrix ( $\mathbf{L}$ ) estimation under the synthetic data model at different  $\epsilon_n$  values with fixed SR = 0.8.

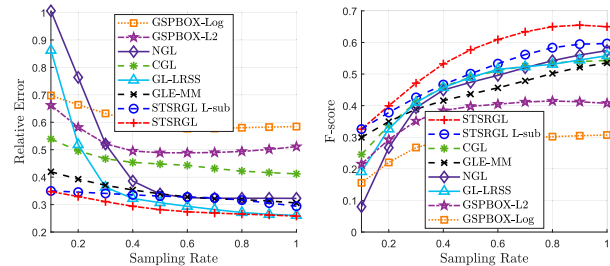


Fig. 2. Relative error and F-score performance of the Laplacian matrix ( $\mathbf{L}$ ) estimation under the synthetic data model at different SR values with fixed  $\epsilon_n = 0.01$ .

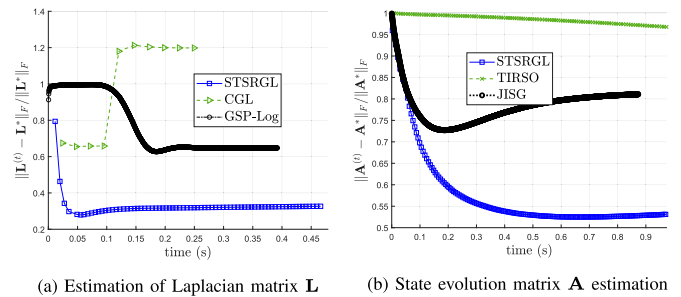


Fig. 3. Convergence plots of different algorithms for inference of the Laplacian/state evolution matrix at SR = 0.8 and  $\epsilon_n = 0.01$ . The vertical axis shows the error with respect to the ground-truth (denoted with asterisk), and the horizontal axis shows the iteration times in seconds.

F-score at different noise levels ( $\epsilon_n$ ) and the fixed sampling rate of SR = 0.8. Fig. 2 also shows the same results for different sampling rates (SR) assuming  $\epsilon_n = 0.01$  (fixed). As depicted, both versions of the proposed algorithm perform suitably in estimating the graph Laplacian matrix, especially in terms of the F-score criterion. Furthermore, the convergence rate of the STSRGL method in Fig. 3 (expressed in CPU time) for the sampling rate SR = 0.8 and  $\epsilon_n = 0.01$  is observed to be faster than the other techniques.

We further apply these methods for learning the Laplacian of different random graphs, namely Barabasi Albert, Erdos-Renyi, and kNN geometric, for the sampling rate SR = 0.8 and noise variance  $\epsilon_n = 0.01$ . The results are provided in Table II.

2) *Inference of the State Transition Matrix:* In this part, we present the results of learning the state transition matrix  $\mathbf{A}$  from

<sup>1</sup><https://github.com/STAC-USC/GraphLearning>

<sup>2</sup><https://github.com/epfl-lts2/gspbox>

<sup>3</sup><https://github.com/mirca/sparseGraph>



TABLE II  
PERFORMANCE OF THE LAPLACIAN MATRIX ESTIMATION FOR RANDOM GRAPH MODELS FROM INCOMPLETE MEASUREMENTS, AT SR = 0.8 AND  $\epsilon_n = 0.01$

	Barbasi-Albert		Erdos-Renyi		kNN	
	F-score	RelErr	F-score	RelErr	F-score	RelErr
CGL	0.4293	0.8721	0.5517	0.3932	0.3899	0.5669
GSPBOX-Log	0.4468	0.8373	0.3709	0.5783	0.5718	0.5604
GLE-MM	0.3522	0.7662	0.5094	0.3018	0.2873	0.4134
NGL	0.4797	1.1093	0.4998	0.3507	0.4764	0.7145
GL-LRSS	0.4804	0.8483	0.6017	0.3083	0.4989	0.5815
STSRGL L-sub	0.3784	0.7743	0.5694	0.3408	0.5782	0.3990
STSRGL	<b>0.5109</b>	<b>0.6267</b>	<b>0.6587</b>	<b>0.2716</b>	<b>0.6911</b>	<b>0.3300</b>

the corrupted observations  $\mathbf{Y}$ . For this purpose, the performance of the proposed algorithm is examined in terms of RelErr and F-score compared to some of the state-of-the-art methods. These include the method in the JISG algorithm in [38] for joint inference of signals and graphs (consisting of two alternating graph learning and signal recovery steps), and also two recent methods in [49] called TISO and TIRSO<sup>4</sup>, for online learning of a VAR model. We also use the classic ordinary least squares regression method in the VAR toolbox<sup>5</sup> [50], for comparison. Similar to the previous part, we simulate two variants of the proposed algorithm in this experiment and report the results. In the first variant, we consider  $\mathbf{w}^{(0)} = \mathcal{P}_{\mathbf{w} \geq 0}(\mathbf{S}_Y^\dagger)$  and  $\mathbf{X} = \mathbf{Y}$ , and find only the solution to the  $\mathbf{A}$ -subproblem. In the second variant, we run the complete BSUM method with all its steps according to the proposed Algorithm 1. The first variant is called the STSRGL A-sub and the second variant is called the STSRGL in the following figures. For the JISG algorithm, we consider the output  $\mathbf{A}^{(1)}$  as the desired matrix. For a fair and accurate comparison, we normalize the output matrix of all the algorithms so that the element with the largest magnitude equals 1. Next, we discard (set to zero) the small elements in the normalized matrix  $\mathbf{A}$  (below a given threshold). Finally the matrix is scaled to have unit operator norm.

Figs. 4 and 5 demonstrate the performance of the proposed algorithms compared to other methods in estimating the state transition matrix under different sampling rates and noise levels. As expected, learning the graph simultaneously when recovering the signal, improves the performance of the  $\mathbf{A}$ -subproblem, specifically, at higher sampling rates and low noise levels. Evidently, the proposed STSRGL method also outperforms the other state-of-the-art algorithms in learning the directed graph matrix  $\mathbf{A}$ .

3) *Data Matrix Recovery*: Here, we examine the recovery performance of the original data matrix  $\mathbf{X}$  in terms of SNR and NMSE criteria. In this experiment, we simulate and report the results for three versions of the proposed algorithm. In the first version, we execute only the  $\mathbf{X}$ -subproblem of the proposed method, and consider  $\mathbf{w}^{(0)} = \mathcal{P}_{\mathbf{w} \geq 0}(\mathbf{S}_Y^\dagger)$  and  $\mathbf{A} = \mathbf{0}$ . In the second version, we again run the  $\mathbf{X}$ -subproblem with the same choice for the Laplacian matrix, except that here

<sup>4</sup><https://github.com/uia-wisenet/OnlineTopologyId>

<sup>5</sup><https://github.com/ambropo/VAR-Toolbox>

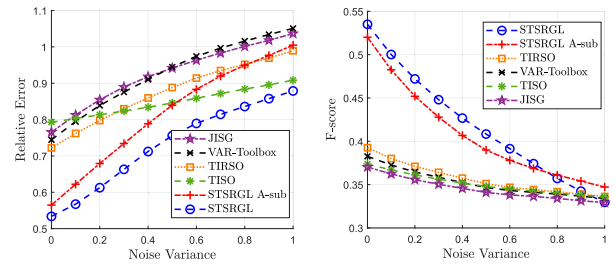


Fig. 4. Relative error and F-score curves of state evolution matrix ( $\mathbf{A}$ ) estimation in the synthetic model, at different  $\epsilon_n$  values for SR = 0.8.

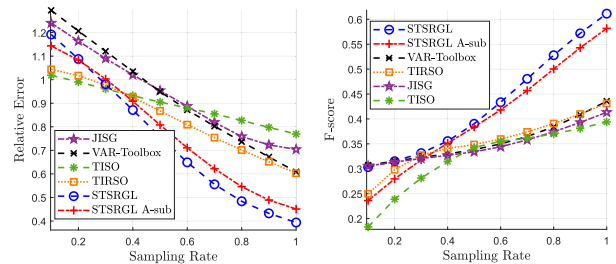


Fig. 5. Relative error and F-score curves of state evolution matrix ( $\mathbf{A}$ ) estimation in the synthetic model, at different SR values for  $\epsilon_n = 0.01$ .

we assume  $\mathbf{A} = \mathbf{I}$ . In the third version, we run the complete BSUM method with all the steps according to the proposed Algorithm 1. We denote the first, the second, and the third versions by STSRGL X-sub, STSRGL X-sub ( $\mathbf{A} = \mathbf{I}$ ), and STSRGL, respectively. We compare the results of our proposed methods with several benchmark signal recovery (matrix completion) algorithms. These algorithms include the SOFT-IMPUTE method<sup>6</sup> for matrix completion via nuclear norm regularization [51], the JISG method, the time-varying graph signal reconstruction method (TVGS)<sup>7</sup> [52], and the method in [53] named as Graph-Tikhonov, which uses Tikhonov regularization with the assumption of spatio-temporal smoothness. The latter two methods require knowledge of the graph Laplacian matrix to model the signal; hence, we primarily use the CGL algorithm to learn the Laplacian matrix from incomplete observations  $\mathbf{Y}$ , and provide the estimated output as the input Laplacian to these algorithms.

Fig. 6 shows the data matrix recovery performance of the aforementioned methods in terms of the SNR and NMSE criteria at various noise levels under SR = 0.8. Fig. 7 also shows the same results for different values of the sampling rate with  $\epsilon_n = 0.01$ . As we can see, learning the graph of the proposed signal model (estimating  $\mathbf{L}$  and  $\mathbf{A}$  matrices) and simultaneously recovering the signal, improves the performance of the  $\mathbf{X}$ -subproblem and increases the reconstruction quality of the corrupted samples. This improvement is of course more evident at higher sampling rates and lower noise levels.

4) *Additional Tests*: Fig. 8, demonstrates the effect of the number of time snapshots  $T$  on the performance of the utilized

<sup>6</sup><https://CRAN.R-project.org/package=softImpute>

<sup>7</sup>[http://gu.ee.tsinghua.edu.cn/codes/Timevarying\\_GS\\_Reconstruction.zip](http://gu.ee.tsinghua.edu.cn/codes/Timevarying_GS_Reconstruction.zip)



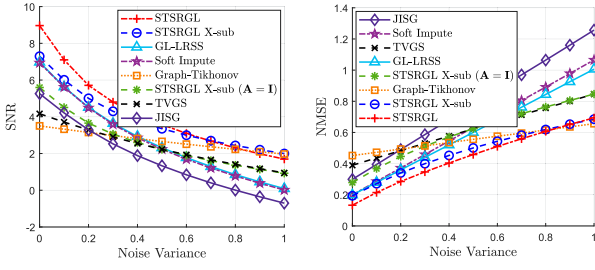


Fig. 6. SNR and NMSE curves of data matrix ( $\mathbf{X}$ ) reconstruction in the synthetic model, at different values of  $\epsilon_n$  for  $\text{SR} = 0.8$ .

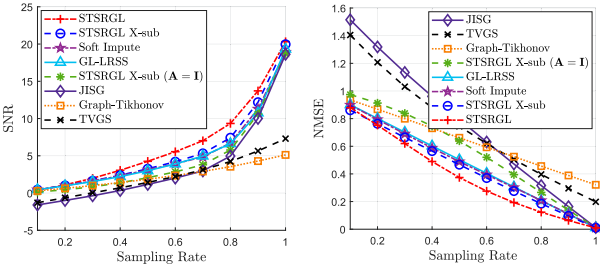


Fig. 7. SNR and NMSE curves of data matrix ( $\mathbf{X}$ ) reconstruction in the synthetic model, at different values of  $\text{SR}$  for  $\epsilon_n = 0.01$ .

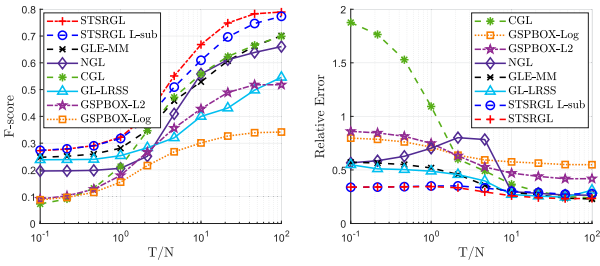


Fig. 8. Performance of the Laplacian ( $\mathbf{L}$ ) estimation algorithms in terms of the ratio  $T/N$ , where noisy and incomplete measurements of synthetic data are used ( $\text{SR} = 0.8$  and  $\epsilon_n = 0.01$ ).

algorithms for learning the Laplacian matrix from noisy and incomplete data, where  $N = 100$  is fixed and  $T$  varies between  $0.1N$  to  $100N$ . The performance is measured in terms of F-score and relative error. Fig. 9 also depicts the performance results for inference of the state transition matrix ( $\mathbf{A}$ ). It is evident that the proposed STSRGL algorithm, has superior performance (specifically in terms of relative error), in both undirected and directed graph learning from data with different number of time snapshots. Moreover, is implied from Fig. 10, that the complexity of the proposed method in terms of CUP run-time, is comparable to the benchmark algorithms used in our simulations. In fact, the proposed STSRGL L-sub and the STSRGL A-sub methods (which are respectively, implementations of the subproblems for learning the Laplacian / state transition matrix) are roughly the least complex methods.

Finally, the quality performance of the proposed STSRGL algorithm with respect to the hyper-parameters, i.e., the parameters  $\alpha_0/T$ ,  $\alpha_1$ , and  $\tau$ , are shown in Figs. 11 to 13.

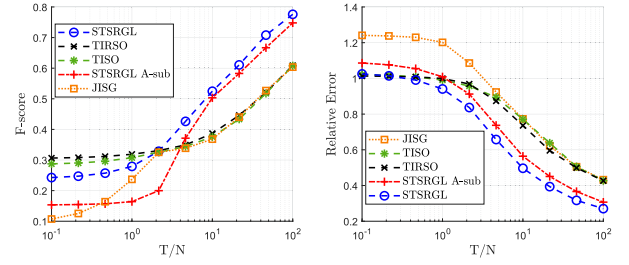


Fig. 9. Performance of the state transition matrix ( $\mathbf{A}$ ) estimation algorithms in terms of the ratio  $T/N$ , where noisy and incomplete measurements of synthetic data are used ( $\text{SR} = 0.8$  and  $\epsilon_n = 0.01$ ).

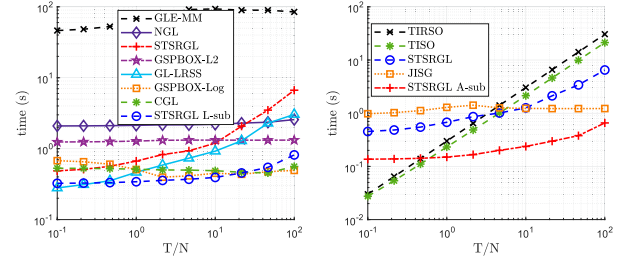


Fig. 10. CPU time (complexity) of the estimations algorithms versus the ratio  $T/N$ , for inference of the Laplacian matrix  $\mathbf{L}$  (left), and the state transition matrix (right). Noisy and incomplete measurements of synthetic data are used ( $\text{SR} = 0.8$  and  $\epsilon_n = 0.01$ ).

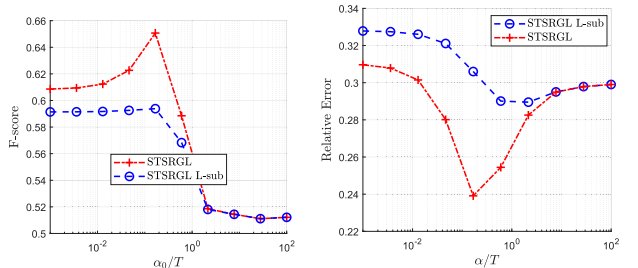


Fig. 11. Performance result of the proposed STSRGL method for Laplacian ( $\mathbf{L}$ ) estimation in terms of the parameter  $\alpha_0$  (the ratio  $\alpha_0/T$ ).

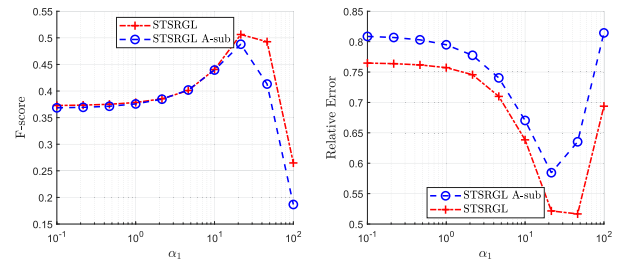


Fig. 12. Performance result of the proposed STSRGL method for the estimation of the state transition matrix ( $\mathbf{A}$ ), in terms of the parameter  $\alpha_1$ .

## B. Real Data

In this part, we present the simulation results of the proposed algorithm on real spatio-temporal data (signals). The spatio-temporal signals are a class of time-varying graph signals in

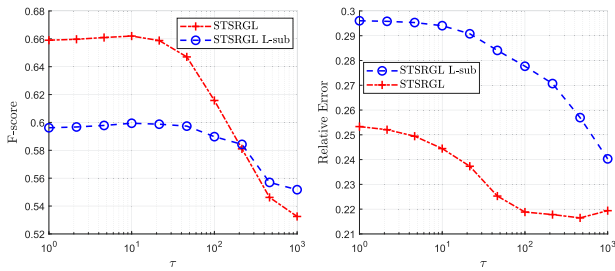


Fig. 13. Performance result of the proposed STSRGL method for Laplacian estimation ( $\mathbf{A}$ ) in terms of the parameter  $\tau$ .

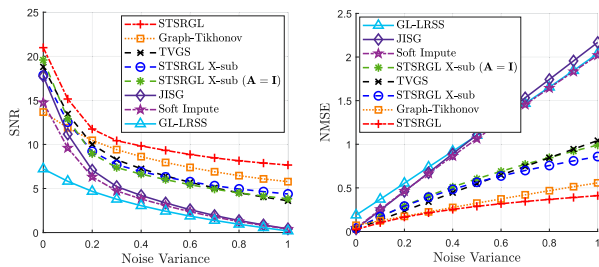


Fig. 14. SNR and NMSE performance of the algorithms in reconstructing the US temperature data matrix ( $\mathbf{X}$ ) at different values of  $\epsilon_n$  for  $\text{SR} = 0.8$ .

which we have a sequence of measurements over time for each vertex of the graph (corresponding to a geographical or a spatial point). We then, normalize the data matrix as described in the previous section by subtracting the mean value from each row and normalizing the rows. The observation matrix is then constructed via  $\mathbf{Y} = \mathbf{M} \odot (\mathbf{X} + \mathbf{N})$  with  $\mathbf{M}$  and  $\mathbf{N}$  generated randomly. Since the ground-truth value of the Laplacian or the state transition matrix is unknown in this scenario, we only provide the results of graph signal recovery methods. We use the same algorithms applied in Section V.A.3, to compare and measure the performance of the proposed algorithm, and finally report the results in terms of SNR and NMSE.

1) *US Temperature Data*: The US temperature dataset<sup>8</sup> includes the average daily measurements of the temperature recorded from 45 US states over 16 years (from 2000 to 2015). From this dataset, we choose the first 450 columns which forms a  $45 \times 450$  sub-matrix; we then, normalize the data matrix by subtracting the mean value from each row and normalizing the rows. The corrupted measurements are obtained as before. The NMSE and SNR performance results of signal recovery from corrupted measurements are shown in Figs. 14 and 15 at different noise levels and sampling rates, respectively. These figures demonstrate the efficiency of the proposed algorithm for spatio-temporal signal recovery compared to several state-of-the-art methods.

2) *PM2.5 Concentration Data*: Here, we experiment on the air pollution index data for PM2.5 concentration in the state of California<sup>9</sup>. The air pollution dataset contains measurements by

<sup>8</sup><http://www.esrl.noaa.gov/psd>

<sup>9</sup><https://www.epa.gov/outdoor-air-quality-data>

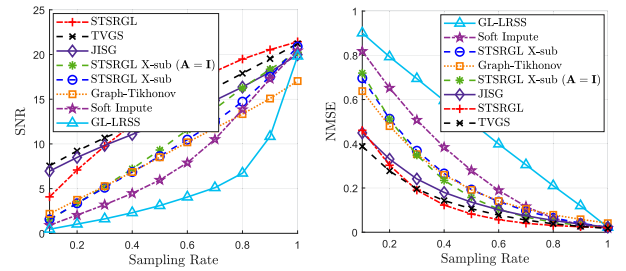


Fig. 15. SNR and NMSE performance of the algorithms in reconstructing the US temperature data matrix ( $\mathbf{X}$ ) at different values of  $\text{SR}$  for  $\epsilon_n = 0.01$ .

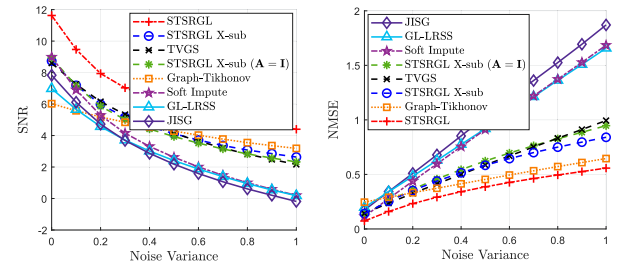


Fig. 16. SNR and NMSE performance of the algorithms in reconstructing the PM2.5 data matrix ( $\mathbf{X}$ ) at different values of  $\epsilon_n$  for  $\text{SR} = 0.8$ .

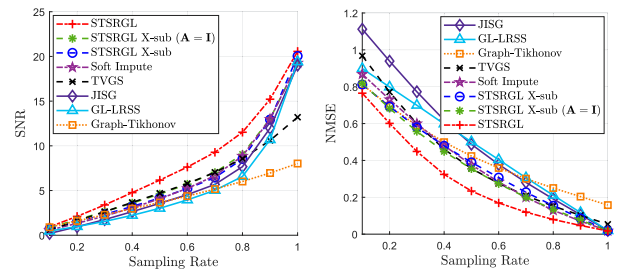


Fig. 17. SNR and NMSE performance of the algorithms in reconstructing the PM2.5 data matrix ( $\mathbf{X}$ ) at different values of  $\text{SR}$  for  $\epsilon_n = 0.01$ .

93 stations (in California) within 300 days starting from January 1, 2015. Hence, the data matrix is of dimension  $93 \times 300$ .

As shown in Figs. 16 and 17, the STSRGL algorithm outperforms the state-of-the-art methods for real spatio-temporal signal recovery. It has efficient performance in estimating the state transition matrix that models the dynamic behavior of the signal, compared to choosing  $\mathbf{A} = \mathbf{0}$ , or  $\mathbf{A} = \mathbf{I}$ , in STSRGL X-sub method. Consequently, simultaneous inference of graphs and signals has improved the reconstruction quality of corrupted samples of the data matrix.

## VI. CONCLUSION

In this paper, we proposed a method to learn a graph-based model from noisy and incomplete data. We incorporated a multi-relational graphical model exploiting both directed and undirected structures. Our model is a spatio-temporal vector auto-regressive (VAR) model in which the excitation process is defined over a Gaussian Markov random field (GMRF), and the state transition matrix is also an arbitrary matrix. We applied

a maximum-a-posteriori estimation method for joint inference of the underlying graphical model and the signal. We proposed an algorithm to solve the optimization problem using the block successive upperbound minimization (BSUM) method. We provided proof of convergence for the proposed method and also analyzed the conditions for the uniqueness of the solution. We finally examined the performance of the proposed method on synthetic and real data. The simulation results confirm the effectiveness of the proposed method for joint signal and graph inference of spatio-temporal or time-varying graph signals.

#### APPENDIX A PROOF OF PROPOSITION 1

*Proof:* Using  $\mathbf{E} = \mathbf{X} - \mathbf{A}\mathbf{X}\mathbf{D}$ , and the properties of the Kronecker product [54], we have:

$$\text{Tr}(\mathcal{L}(\mathbf{w})\mathbf{E}\mathbf{E}^\top) = \text{vec}(\mathbf{E})^\top (\mathbf{I}_T^\top \otimes \mathcal{L}(\mathbf{w}))^\top \text{vec}(\mathbf{E}). \quad (27)$$

Moreover,  $\text{vec}(\mathbf{E})$  can be restated as  $\mathbf{H}\text{vec}(\mathbf{X})$  where  $\mathbf{H} = \mathbf{I}_{nT} - \mathbf{D}^\top \otimes \mathbf{A}$ . Similarly, we may write

$$\begin{aligned} \|\mathbf{Y}_M - \mathbf{M} \odot \mathbf{X}\|_F^2 &= \|\text{vec}(\mathbf{Y}_M)\|_2^2 + \|\text{vec}(\mathbf{M} \odot \mathbf{X})\|_2^2 \\ &\quad - 2\text{vec}(\mathbf{Y}_M)^\top \text{vec}(\mathbf{M} \odot \mathbf{X}). \end{aligned} \quad (28)$$

Finally, using  $\text{vec}(\mathbf{M} \odot \mathbf{X}) = \text{Diag}(\text{vec}(\mathbf{M}))\text{vec}(\mathbf{X})$  and  $\text{vec}(\mathbf{Y}_M)^\top \text{vec}(\mathbf{M} \odot \mathbf{X}) = \text{vec}(\mathbf{Y}_M)^\top \text{vec}(\mathbf{X})$ , we obtain

$$\begin{aligned} f_{\mathbf{X}}(\mathbf{X}) &= \frac{1}{\sigma_n^2} \text{vec}(\mathbf{X})^\top (\text{Diag}(\text{vec}(\mathbf{M}))\text{vec}(\mathbf{X}) - 2\text{vec}(\mathbf{Y}_M)) \\ &\quad + \text{vec}(\mathbf{X})^\top \mathbf{H}^\top (\mathbf{I}_T \otimes \mathcal{L}(\mathbf{w})) \mathbf{H} \text{vec}(\mathbf{X}) + \text{const} \\ &= \text{vec}(\mathbf{X})^\top \mathbf{G} \text{vec}(\mathbf{X}) - 2\mathbf{b}^\top \text{vec}(\mathbf{X}) + \text{const}, \end{aligned} \quad (29)$$

where  $\mathbf{G}$ ,  $\mathbf{H}$  and  $\mathbf{b}$  are given in (15).

Now, since  $\mathcal{L}(\mathbf{w})^{1/2}$  exists, for any  $\mathbf{x} \in \mathbb{R}^{nT}$  we have

$$\mathbf{x}^\top \mathbf{G} \mathbf{x} = \frac{1}{\sigma_n^2} \|\text{Diag}(\text{vec}(\mathbf{M}))\mathbf{x}\|_2^2 + \left\| (\mathbf{I} \otimes \mathcal{L}(\mathbf{w})^{1/2}) \mathbf{H} \mathbf{x} \right\|_2^2,$$

which is always non-negative. Thus,  $\mathbf{G}$  is positive semi-definite (and symmetric) with all real non-negative eigenvalues. Hence, the function (13) is always convex with respect to  $\mathbf{X}$ .

To prove the second statement, let  $\mathbf{P} = \mathbf{H}^\top (\mathbf{I}_T \otimes \mathcal{L}(\mathbf{w})) \mathbf{H}$ . Then, we have  $\mathbf{G} = \frac{1}{\sigma_n^2} \text{Diag}(\text{vec}(\mathbf{M})) + \mathbf{P}$ , where  $\mathbf{P}$  is a symmetric block-Toeplitz matrix of size  $nT \times nT$  with  $\mathcal{L}(\mathbf{w}) + \mathbf{A}^\top \mathcal{L}(\mathbf{w}) \mathbf{A}$  as the main diagonal,  $-\mathbf{A}^\top \mathcal{L}(\mathbf{w})$  as the upper diagonal and  $-\mathcal{L}(\mathbf{w}) \mathbf{A}$  as the lower diagonal blocks. It is easy to verify that each row in the sub-matrix  $\mathbf{P}_{in:(i+1)n,:}$  is linearly independent from the rows in all other sub-matrices  $\mathbf{P}_{jn:(j+1)n,:}$ ,  $j \neq i$  since they have different supports (the set of nonzero indices). Now, for  $\mathbf{G}$  to be of full row rank, it suffices for each sub-matrix  $\mathbf{G}_{in:(i+1)n,:}$  to have full row rank. Moreover,  $\mathbf{G}_{in:(i+1)n,:} = \mathbf{P}_{in:(i+1)n,:} + \frac{1}{\sigma_n^2} \text{Diag}(\mathbf{m}_i)$ , where  $\mathbf{m}_i = \text{vec}(\mathbf{M})_{in:(i+1)n}$  is the  $i$ -th column of  $\mathbf{M}$ . Hence, a sufficient condition for  $\mathbf{G}$  to be of full row rank is that each diagonal block  $\mathcal{L}(\mathbf{w}) + \mathbf{A}^\top \mathcal{L}(\mathbf{w}) \mathbf{A} + \frac{1}{\sigma_n^2} \text{Diag}(\mathbf{m}_i)$  has full rank, i.e., its minimum eigenvalue is greater than zero. This happens if  $\mathbf{m}_i$  has at least one non-zero entry ( $\sum_k M_{k,i} > 0$ ) and  $\sigma_n < \infty$  (since  $\forall \mathbf{v} \neq \mathbf{0} \in \text{Null}(\mathcal{L}(\mathbf{w})) = \{\beta \mathbf{1}, \beta \neq 0\}$ , the term  $1/\sigma_n^2 \mathbf{v}^\top \text{Diag}(\mathbf{m}_i) \mathbf{v}$  is strictly positive). Finally,  $\mathbf{G}$

will be invertible if it has full (row) rank. In this case, the function (13) can be uniquely minimized via the following closed-form solution

$$\begin{aligned} \mathbf{X}^* &= \underset{\text{vec}(\mathbf{X})}{\text{argmin}} \text{vec}(\mathbf{X})^\top \mathbf{G} \text{vec}(\mathbf{X}) - 2\mathbf{b}^\top \text{vec}(\mathbf{X}) \\ &= \text{vec}^{-1}(\mathbf{G}^{-1} \mathbf{b}). \end{aligned} \quad (30)$$

This concludes the proof.  $\square$

#### APPENDIX B PROOF OF LEMMA 1

*Proof:* If  $\theta > \theta_{\min} = \lambda_{\max}(\mathbf{G})$ , then,  $\text{vec}(\mathbf{X} - \mathbf{X}_0)^\top (\theta \mathbf{I} - \mathbf{G}) \text{vec}(\mathbf{X} - \mathbf{X}_0)$  is strictly positive for  $\mathbf{X} \neq \mathbf{X}_0$ , implying that  $f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}_0) \geq f_{\mathbf{X}}(\mathbf{X})$  with equality achieved only at  $\mathbf{X} = \mathbf{X}_0$ . Moreover, we have  $D_{\mathbf{X}} f_{\mathbf{X}}(\mathbf{X}_0) = D_{\mathbf{X}} f_{\mathbf{X}}^S(\mathbf{X}_0; \mathbf{X}_0)$ . Hence, the function  $f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}_0)$  given in (16) is a majorization function for  $f_{\mathbf{X}}(\mathbf{X})$  which always admits a unique minimizer if  $\theta > \lambda_{\max}(\mathbf{G})$ . This function can be simplified as follows

$$\begin{aligned} f_{\mathbf{X}}^S(\mathbf{X}; \mathbf{X}_0) &= f_{\mathbf{X}}(\mathbf{X}) + \text{vec}(\mathbf{X} - \mathbf{X}_0)^\top (\theta \mathbf{I} - \mathbf{G}) \text{vec}(\mathbf{X} - \mathbf{X}_0) \\ &= \theta \left\| \text{vec}(\mathbf{X} - \mathbf{X}_0) + \frac{\mathbf{G} \text{vec}(\mathbf{X}_0) - \mathbf{b}}{\theta} \right\|^2 + \text{const}. \end{aligned}$$

Now, it is trivial that the above quadratic function is strictly convex with a unique minimizer specified by  $\text{vec}(\mathbf{X}_0) - \frac{1}{\theta} (\mathbf{G} \text{vec}(\mathbf{X}_0) - \mathbf{b})$  where  $\mathbf{G} \text{vec}(\mathbf{X}_0) - \mathbf{b} = \text{vec}(\frac{1}{2} \frac{\partial}{\partial \mathbf{X}} f_{\mathbf{X}}(\mathbf{X})|_{\mathbf{X}_0})$ .

We may also obtain an upper-bound for  $\|\mathbf{G}\|$  as follows

$$\begin{aligned} \|\mathbf{G}\| &= \max_{\|\mathbf{x}\|=1} \|\mathbf{G} \mathbf{x}\| \\ &= \max_{\|\mathbf{x}\|=1} \left\| \frac{1}{\sigma_n^2} \text{Diag}(\text{vec}(\mathbf{M})) \mathbf{x} + \mathbf{H}^\top (\mathbf{I} \otimes \mathcal{L}(\mathbf{w})) \mathbf{H} \mathbf{x} \right\| \\ &\leq \max_{\|\mathbf{x}\|=1} \frac{1}{\sigma_n^2} \|\text{Diag}(\text{vec}(\mathbf{M})) \mathbf{x}\| + \|\mathbf{H}^\top (\mathbf{I} \otimes \mathcal{L}(\mathbf{w})) \mathbf{H} \mathbf{x}\| \\ &\leq \max_{\|\mathbf{x}\|=1} \frac{1}{\sigma_n^2} \|\mathbf{x}\| + \|\mathbf{H}\|^2 \|\mathbf{I} \otimes \mathcal{L}(\mathbf{w})\| \|\mathbf{x}\| \\ &= \frac{1}{\sigma_n^2} + \|\mathbf{H}\|^2 \|\mathcal{L}(\mathbf{w})\|, \end{aligned} \quad (31)$$

where  $\mathbf{H} = \mathbf{I} - \mathbf{D}^\top \otimes \mathbf{A}$ . The second inequality results from  $\|\mathbf{M}_1 \mathbf{M}_2\| \leq \|\mathbf{M}_1\| \|\mathbf{M}_2\|$  and the last equality is also obtained from  $\|\mathbf{M}_1 \otimes \mathbf{M}_2\| = \|\mathbf{M}_1\| \|\mathbf{M}_2\|$ . We also have

$$\begin{aligned} \|\mathbf{H}\|^2 &= \max_{\|\mathbf{x}\|=1} \|\mathbf{x} - \mathbf{D}^\top \otimes \mathbf{A} \mathbf{x}\|^2 \\ &\leq \max_{\|\mathbf{x}\|=1} 2 \left( \|\mathbf{x}\|^2 + \|\mathbf{D}^\top \otimes \mathbf{A} \mathbf{x}\|^2 \right) \\ &= 2 \left( 1 + \|\mathbf{D}^\top \otimes \mathbf{A}\|^2 \right) \\ &= 2 \left( 1 + \|\mathbf{D}^\top\|^2 \|\mathbf{A}\|^2 \right) = 2 \left( 1 + \|\mathbf{A}\|^2 \right), \end{aligned} \quad (32)$$

where the first inequality results from the Cauchy-Schwarz. Hence, we conclude that  $\|\mathbf{G}\|_2 = \lambda_{\max}(\mathbf{G}) \leq \frac{1}{\sigma_n^2} + 2 \|\mathcal{L}(\mathbf{w})\| \left( 1 + \|\mathbf{A}\|^2 \right)$ . Therefore, it only suffices to choose  $\theta > \hat{\theta}_{\min} = \frac{1}{\sigma_n^2} + 2 \|\mathcal{L}(\mathbf{w})\| \left( 1 + \|\mathbf{A}\|^2 \right)$ .

The advantage of using  $\hat{\theta}_{\min}$  instead of  $\lambda_{\max}(\mathbf{G})$  as lower bound for  $\theta$ , is that we can compute  $\hat{\theta}_{\min}$  with  $\mathcal{O}(n^3)$  operations which is considerably less complex than that of  $\lambda_{\max}(\mathbf{G})$  (which is  $\mathcal{O}(n^3 T^3)$  complex).  $\square$

APPENDIX C  
PROOF OF PROPOSITION 2

*Proof:* Using vectorial representation, we have  $f_{\mathbf{A}}(\mathbf{A}) = \text{vec}(\mathbf{A})^{\top} \mathbf{F} \text{vec}(\mathbf{A}) - 2\text{vec}(\mathbf{C})^{\top} \text{vec}(\mathbf{A}) + 2\alpha_1 \|\text{vec}(\mathbf{A})\|_1$ , with  $\mathbf{F} = (\mathbf{X}\mathbf{D}(\mathbf{X}\mathbf{D})^{\top}) \otimes \mathcal{L}(\mathbf{w})$  and  $\mathbf{C} = \mathcal{L}(\mathbf{w})\mathbf{X}(\mathbf{X}\mathbf{D})^{\top}$ . Now, since,  $\mathcal{L}(\mathbf{w})^{1/2} \succeq 0$  exists, we can rewrite the problem of minimizing  $f_{\mathbf{A}}(\mathbf{A})$  as the following LASSO problem

$$\text{vec}(\mathbf{A}^*) = \underset{\text{vec}(\mathbf{A})}{\text{argmin}} \frac{1}{2} \|\mathbf{R}\text{vec}(\mathbf{A}) - \mathbf{d}\|_2^2 + \alpha_1 \|\text{vec}(\mathbf{A})\|_1,$$

where  $\mathbf{R} = (\mathbf{X}\mathbf{D})^{\top} \otimes \mathcal{L}(\mathbf{w})^{1/2}$  and  $\mathbf{d} = \text{vec}(\mathcal{L}(\mathbf{w})^{1/2}\mathbf{X})$ . Then, if we use Lemma 2 in [55], the proof is complete.  $\square$

APPENDIX D  
PROOF OF LEMMA 2

*Proof:* Define  $\beta_{\min} = \inf\{\beta \geq 0 : \beta \|\mathbf{V}\|_F^2 - \text{Tr}(\mathcal{L}(\mathbf{w})\mathbf{V}\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}\mathbf{V}^{\top}) > 0 \quad \forall \mathbf{V} \neq \mathbf{0} \in \mathbb{R}^{n \times n}\}$ . On one hand, we have

$$\begin{aligned} \text{Tr}(\mathcal{L}(\mathbf{w})\mathbf{V}\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}\mathbf{V}^{\top}) \\ = \text{vec}(\mathbf{V})^{\top} [(\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}) \otimes \mathcal{L}(\mathbf{w})] \text{vec}(\mathbf{V}). \end{aligned}$$

On the other hand,  $\beta \|\mathbf{V}\|_F^2 = \text{vec}(\mathbf{V})^{\top} \beta \mathbf{I} \text{vec}(\mathbf{V})$ . Thus,  $\beta \|\mathbf{V}\|_F^2 - \text{Tr}(\mathcal{L}(\mathbf{w})\mathbf{V}\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}\mathbf{V}^{\top})$  is positive for all  $\mathbf{V} \neq \mathbf{0}$ , if and only if the matrix  $\beta \mathbf{I} - (\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}) \otimes \mathcal{L}(\mathbf{w})^{\top}$  is positive definite, or equivalently  $\beta > \beta_{\min} = \lambda_{\max}(\mathbf{F})$  where  $\mathbf{F} = (\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}) \otimes \mathcal{L}(\mathbf{w})^{\top}$ . By setting  $\mathbf{V} = \mathbf{A} - \mathbf{A}_0$ , we conclude that  $f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0) > f_{\mathbf{A}}(\mathbf{A}) \quad \forall \mathbf{A} \neq \mathbf{A}_0$ , and  $f_{\mathbf{A}}^S(\mathbf{A}_0; \mathbf{A}_0) = f_{\mathbf{A}}(\mathbf{A}_0)$ . Moreover, since the additional term in  $f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A})$  is quadratic and smooth, we have  $D_{\mathbf{A}} f_{\mathbf{A}}(\mathbf{A}_0) = D_{\mathbf{A}} f_{\mathbf{A}}^S(\mathbf{A}_0; \mathbf{A}_0)$ . Consequently,  $f_{\mathbf{A}}^S(\mathbf{A}; \mathbf{A}_0)$  is a majorization function [45] for  $f_{\mathbf{A}}(\mathbf{A})$ . We may write

$$f_{\mathbf{A}}^S(\mathbf{A}, \mathbf{A}_0) = \frac{1}{2} \|\text{vec}(\mathbf{A}) - \text{vec}(\mathbf{A}_0)\|_2^2 + \frac{\alpha_1}{\beta} \|\text{vec}(\mathbf{A})\|_1,$$

which implies that  $f_{\mathbf{A}}^S(\mathbf{A}, \mathbf{A}_0)$  is strictly convex and admits a unique minimizer. Now, using the Kronecker property, we have  $\lambda_{\max}(\mathbf{F}) = \lambda_{\max}(\mathbf{X}\mathbf{D}\mathbf{D}^{\top}\mathbf{X}^{\top}) \lambda_{\max}(\mathcal{L}(\mathbf{w})) = \|\mathbf{X}\mathbf{D}\|_2^2 \|\mathcal{L}(\mathbf{w})\|$ . Hence, we may use  $\|\mathbf{X}\mathbf{D}\|_2^2 \|\mathcal{L}(\mathbf{w})\|$  as  $\beta_{\min}$  which takes  $\mathcal{O}(n^3)$  operations to obtain (much simpler than  $\mathcal{O}(n^3 T^3)$  for  $\lambda_{\max}(\mathbf{F})$ ).  $\square$

APPENDIX E  
PROOF OF PROPOSITION 3

*Proof:* The convexity of (22) can be deduced using the convexity of  $-\log \det(\Phi)$ , the affinity of  $\Phi = \mathcal{L}(\mathbf{w}) + \mathbf{J}$  and the affinity of  $\text{Tr}(\mathcal{L}(\mathbf{w})\mathbf{K}) = \langle \mathbf{w}, \mathcal{L}^*(\mathbf{K}) \rangle$  with respect to  $\mathbf{w}$ . Let  $\mathbf{L} = \mathcal{L}(\mathbf{w})$ . According to the definition of the Laplacian operator [21], we have  $\mathcal{L}(\mathbf{w}) = \mathbf{E}\text{Diag}(\mathbf{w})\mathbf{E}^{\top}$  where  $\mathbf{E} = [\xi_1, \dots, \xi_{n(n-1)/2}] \in \mathbb{R}^{n \times n(n-1)/2}$ . The vector  $\xi_k$  for  $k = i - j + \frac{i-1}{2}(2n-j)$ ,  $i > j$ , has  $-1$  at the  $i$ -th position,  $+1$  at the  $j$ -th position, and zeros elsewhere. In this definition,  $w_k = -L_{ij}$ . It can be easily verified that  $\mathcal{L}(\mathbf{w})$  is symmetric. Moreover, since  $\xi_i^{\top} \mathbf{1} = \mathbf{0}$ ,  $\forall i$ , we have  $\mathcal{L}(\mathbf{w})\mathbf{1} = \mathbf{0}$ . Furthermore, the constraints  $L_{i,i} \geq 0$  and  $L_{i,j} \leq 0$ ,  $i \neq j$  are satisfied iff  $w_i \geq 0$ . Besides, for any  $\mathbf{x} \in \mathbb{R}^n$  we may

write  $\mathbf{x}^{\top} \mathcal{L}(\mathbf{w})\mathbf{x} = \mathbf{x}^{\top} (\sum_i w_i \xi_i \xi_i^{\top}) \mathbf{x} = \sum_i w_i (\xi_i^{\top} \mathbf{x})^2$ . Consequently, the constraint  $\mathcal{L}(\mathbf{w}) \succeq 0$  holds iff  $w_i \geq 0$ . Therefore,  $\mathcal{L}(\mathbf{w}) + \mathbf{J}$  is (symmetric) positive semi-definite if  $\mathbf{w} \geq 0$ . Furthermore,  $\text{dom}(f_{\mathbf{w}}) = \{\mathbf{w} \mid \text{rank}(\mathcal{L}(\mathbf{w}) + \mathbf{J}) = n\}$  due to the term  $-\log \det(\mathcal{L}(\mathbf{w}) + \mathbf{J})$ . Since  $\text{rank}(\mathbf{J}) = 1$ ,  $\mathbf{w} \in \text{dom}(f_{\mathbf{w}})$  iff  $\text{rank}(\mathcal{L}(\mathbf{w})) = n - 1$ . Therefore, it only suffices to define the feasible set as  $\Omega_{\mathbf{w}} = \{\mathbf{w} \mid \mathbf{w} \geq 0\}$  to satisfy the Laplacian constraints specified by  $\Omega_{\mathbf{L}}$ . Moreover,  $f_{\mathbf{w}}(\mathbf{w})$  is lower bounded by

$$\begin{aligned} f_{\mathbf{w}}(\mathbf{w}) &= -\log \left( \prod_{i=2}^n \lambda_i(\mathcal{L}(\mathbf{w})) \right) + \langle \mathbf{w}, \mathcal{L}^*(\mathbf{K}) \rangle \\ &\geq -(n-1) \log \left( \frac{1}{n-1} \sum_{i=2}^n \lambda_i(\mathcal{L}(\mathbf{w})) \right) + \langle \mathbf{w}, \mathcal{L}^*(\mathbf{K}) \rangle \\ &= (n-1) \left( \log(n-1) - \log \text{Tr}(\mathcal{L}(\mathbf{w})) \right) + \langle \mathbf{w}, \mathcal{L}^*(\mathbf{K}) \rangle \\ &\geq -(n-1) \log \left( \sum_{i=1}^{n(n-1)/2} w_i \right) + r_{\min} \left( \sum_{i=1}^{n(n-1)/2} w_i \right) \\ &\quad + (n-1) \log \frac{n-1}{2}, \end{aligned}$$

where the first inequality is due to the Jensen's inequality and the last inequality is obtained from  $r_i = [\mathcal{L}^*(\mathbf{K})]_i \geq r_{\min}$ , where  $r_{\min} = \min_i r_i$ . If  $\mathbf{K} \succ 0$  or  $\alpha_0 > 0$ , then  $r_i = [\mathcal{L}^*(\mathbf{K})]_i = \xi_i^{\top} \mathbf{K} \xi_i > 0$ , and hence,  $r_{\min}$  would be positive. Now, define

$$h(z) = -(n-1) \log z + r_{\min} z + (n-1) \log \frac{n-1}{2} \quad (33)$$

for  $z = \sum_i w_i > 0$ . It can be easily verified that  $h(z)$  is strictly convex with a unique minimizer specified by  $z^* = (n-1)/r_{\min}$ . Thus, we conclude that  $f_{\mathbf{w}}(\mathbf{w}) \geq h(z) \geq h(z^*)$  and  $\lim_{z \rightarrow +\infty} h(z) = +\infty$ . Hence, similar to [43], it can be shown that  $f_{\mathbf{w}}(\mathbf{w})$  has a unique minimizer for  $\mathbf{w} \geq 0$ .  $\square$

APPENDIX F  
PROOF OF LEMMA 3

*Proof:* Using the definition of the Laplacian operator [21], we may write

$$\mathcal{L}(\mathbf{w}) + \mathbf{J} = \mathbf{E}\text{Diag}(\mathbf{w})\mathbf{E}^{\top} + \mathbf{J} = \mathbf{C}\text{Diag}(\tilde{\mathbf{w}})\mathbf{C}^{\top}, \quad (34)$$

where  $\mathbf{C} = [\mathbf{E}, \mathbf{1}]$  and  $\tilde{\mathbf{w}} = [\mathbf{w}^{\top}, 1/n]^{\top}$ . Now, using the concavity of the log function, a majorization function for  $-\log \det(\mathcal{L}(\mathbf{w}) + \mathbf{J})$  can be achieved as:

$$\begin{aligned} -\log \det(\mathcal{L}(\mathbf{w}) + \mathbf{J}) &\leq \text{Tr}(\mathbf{Q}_0(\mathbf{C}\text{Diag}(\tilde{\mathbf{w}})\mathbf{C}^{\top})^{-1}) \\ &\quad - \log \det(\mathcal{L}(\mathbf{w}_0) + \mathbf{J}) - n, \end{aligned} \quad (35)$$

where  $\mathbf{Q}_0 = \mathcal{L}(\mathbf{w}_0) + \mathbf{J}$ , and the equality happens only at  $\mathbf{w} = \mathbf{w}_0$ . Another majorizer for the first term in the right-hand side of (35) can be obtained using Lemma 4 in [21]:

$$\begin{aligned} \text{Tr}(\mathbf{Q}_0(\mathbf{C}\text{Diag}(\tilde{\mathbf{w}})\mathbf{C}^{\top})^{-1}) \\ \leq \text{Tr}(\mathbf{Q}_0^{1/2} \mathbf{Q}_0^{-1} \mathbf{C}\text{Diag}(\tilde{\mathbf{w}})^2 \text{Diag}(\tilde{\mathbf{w}})^{-1} \mathbf{C}^{\top} \mathbf{Q}_0^{-1} \mathbf{Q}_0^{1/2}) \\ = \langle \mathbf{w}_0^2 \oslash \mathbf{w}, \mathcal{L}^*(\mathbf{Q}_0^{-1}) \rangle + \text{Tr}(\mathbf{Q}_0^{-1} \mathbf{J}), \end{aligned} \quad (36)$$



where in the last equation, we used the property  $\text{Tr}(\mathcal{L}(\mathbf{w}), \mathbf{H}) = \langle \mathbf{w}, \mathcal{L}^*(\mathbf{H}) \rangle$ . Define  $g_{\mathbf{w}}(\mathbf{w}; \mathbf{w}_0) = \text{Tr}(\mathcal{L}(\mathbf{w})\mathbf{K}) + \langle \mathbf{w}_0^{\circ 2} \circ \mathbf{w}, \mathcal{L}^*(\mathbf{Q}_0^{-1}) \rangle$  and

$$\mathbf{r} = \mathcal{L}^*(\mathbf{K}), \quad \mathbf{q} = \mathcal{L}^*(\mathbf{Q}_0^{-1}) = \mathcal{L}^*((\mathcal{L}(\mathbf{w}_0) + \mathbf{J})^{-1}).$$

We can now decompose  $g_{\mathbf{w}}(\mathbf{w}; \mathbf{w}_0)$  into separable functions of  $w_i$  as follows

$$g_{\mathbf{w}}(\mathbf{w}; \mathbf{w}_0) = \sum_{i=1}^{n(n-1)/2} g_{w_i}(w_i; w_{0i}) = r_i w_i + q_i \frac{w_{0i}^2}{w_i}, \quad (37)$$

where  $r_i = [\mathbf{r}]_i$ , and  $q_i = [\mathbf{q}]_i$ . Consider the function  $h(x) = x + \frac{1}{x} - 2$ . It can be easily verified that  $h(x) \geq 0$  for  $x > 0$ , with equality achieved only at  $x = 1$ . Moreover, from  $\mathcal{L}(\mathbf{w}^{(j)} + \mathbf{J}) \succ 0$  and  $\mathbf{K} \succeq 0$ , it is implied that  $q_i > 0$  and  $r_i \geq 0$  (using the definition of the  $\mathcal{L}^*$  operator [19]). Hence, we may propose the following majorization function for  $g_{w_i}(w_i; w_{0i})$

$$\begin{aligned} g_{w_i}^S(w_i; w_{0i}) &= g_{w_i}(w_i; w_{0i}) + \tau q_i w_{0i}^2 h(w_i/w_{0i}) \\ &= r_i w_i + \tau q_i w_{0i}^2 \left( \frac{w_i}{w_{0i}} + \frac{w_{0i} + 1/\tau}{w_i} - 2 \right), \end{aligned}$$

where  $\tau > 0$  is constant. Together with (35) and (36), we finally obtain the following majorization function for  $f_{\mathbf{w}}(\mathbf{w})$

$$\begin{aligned} f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{w}_0) &= \sum_i g_{w_i}^S(w_i; w_{0i}) + \text{Tr}(\mathbf{Q}_0^{-1}\mathbf{J}) \\ &\quad - \log \det(\mathcal{L}(\mathbf{w}_0) + \mathbf{J}) - n, \end{aligned}$$

which is simplified to (25). One can easily verify that  $f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{w}_0)$  satisfies the properties of the majorization function (see [45]). It is also straightforward to show that the second partial derivatives of  $f_{\mathbf{w}}^S(\mathbf{w}; \mathbf{w}_0)$  are all positive, hence, the function is strictly convex.  $\square$

## REFERENCES

- [1] A. Ortega, P. Frossard, J. Kovacević, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proc. IEEE*, vol. 106, no. 5, pp. 808–828, May 2018.
- [2] A. G. Marques, N. Kiyavash, J. M. F. Moura, D. Van De Ville, and R. Willett, "Graph signal processing: Foundations and emerging directions [From the Guest Editors]," *IEEE Signal Process. Mag.*, vol. 37, no. 6, pp. 11–13, Nov. 2020.
- [3] X. Dong, D. Thanou, L. Toni, M. Bronstein, and P. Frossard, "Graph signal processing for machine learning: A review and new perspectives," *IEEE Signal Process. Mag.*, vol. 37, no. 6, pp. 117–127, Nov. 2020.
- [4] W. Campbell, C. Dagli, and C. Weinstein, "Social network analysis with content and graphs," *Lincoln Lab. J.*, vol. 20, no. 1, pp. 62–81, 2013.
- [5] H. Tamura, K. Nakano, M. Sengoku, and S. Shinoda, "On applications of graph/network theory to problems in communication systems," *ECTI Trans. Comput. Inf. Technol.*, vol. 5, no. 1, pp. 15–21, Jan. 1970.
- [6] X. Piao, Y. Hu, Y. Sun, B. Yin, and J. Gao, "Correlated spatio-temporal data collection in wireless sensor networks based on low rank matrix approximation and optimized node sampling," *Sensors*, vol. 14, no. 12, pp. 23137–23158, Dec. 2014.
- [7] H. A. Loeliger, "An introduction to factor graphs," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Jan. 2004.
- [8] A. Lalitha, X. Wang, O. Kilinc, Y. Lu, T. Javidi, and F. Koushanfar, "Decentralized Bayesian learning over graphs," May 2019, *arXiv:1905.10466*.
- [9] G. B. Giannakis, Y. Shen, and G. V. Karanikolas, "Topology identification and learning over graphs: Accounting for nonlinearities and dynamics," *Proc. IEEE*, vol. 106, no. 5, pp. 787–807, May 2018.
- [10] X. Dong, D. Thanou, M. Rabbat, and P. Frossard, "Learning graphs from data: A signal representation perspective," *IEEE Signal Process. Mag.*, vol. 36, no. 3, pp. 44–63, May 2019.
- [11] F. Xia et al., "Graph learning: A survey," *IEEE Trans. Artif. Intell.*, vol. 2, no. 2, pp. 109–127, Apr. 2021.
- [12] S. Chen, R. Verma, A. Singh, and J. Kovacevic, "Signal recovery on graphs: Fundamental limits of sampling strategies," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 4, pp. 539–554, Dec. 2016.
- [13] P. D. Lorenzo, S. Barbarossa, and P. Banelli, "Sampling and recovery of graph signals," in *Cooperative and Graph Signal Processing*, P. M. Djuric and C. Richard, Eds., Cambridge, MA, USA: Academic Press, 2018, pp. 261–282.
- [14] 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.
- [15] B. Lake and J. Tenenbaum, "Discovering structure by learning sparse graphs," in *Proc. 32nd Annu. Meeting Cogn. Sci. Soc.*, Portland, OR, USA, Cognitive Science Society, Aug. 2010, pp. 778–784.
- [16] H. Rue and L. Held, *Gaussian Markov Random Fields: Theory and Applications*, Number 104 in Monographs on Statistics and Applied Probability. Boca Raton, FL, USA: Chapman & Hall, 2005.
- [17] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under Laplacian and structural constraints," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 825–841, Sep. 2017.
- [18] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning Laplacian matrix in smooth graph signal representations," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6160–6173, Dec. 2016.
- [19] S. Kumar, J. Ying, J. V. de M. Cardoso, and D. P. Palomar, "A unified framework for structured graph learning via spectral constraints," *J. Mach. Learn. Res.*, vol. 21, no. 22, pp. 1–60, 2020.
- [20] S. Segarra, A. G. Marques, G. Mateos, and A. Ribeiro, "Network topology inference from spectral templates," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 467–483, Sep. 2017.
- [21] L. Zhao, Y. Wang, S. Kumar, and D. P. Palomar, "Optimization algorithms for graph Laplacian estimation via ADMM and MM," *IEEE Trans. Signal Process.*, vol. 67, no. 16, pp. 4231–4244, Aug. 2019.
- [22] V. Kalofolias, "How to learn a graph from smooth signals," in *Proc. 19th Int. Conf. Artif. Intell. Statist.*, May 2016, pp. 920–929.
- [23] H. P. Margetic, D. Thanou, and P. Frossard, "Graph learning under sparsity priors," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2017, pp. 6523–6527.
- [24] C. W. J. Granger, "Some recent development in a concept of causality," *J. Econometrics*, vol. 39, no. 1, pp. 199–211, Sep. 1988.
- [25] D. Kaplan, *Structural Equation Modeling*, 2nd ed. Newbury Park, CA, USA: SAGE, 2009.
- [26] J. Songsiri and L. Vandenberghe, "Topology selection in graphical models of autoregressive processes," *J. Mach. Learn. Res.*, vol. 11, no. 91, pp. 2671–2705, 2010.
- [27] A. Bolstad, B. D. Van Veen, and R. Nowak, "Causal network inference via group sparse regularization," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2628–2641, Jun. 2011.
- [28] E. Isufi, A. Loukas, N. Perraudin, and G. Leus, "Forecasting time series with VARMA recursions on graphs," *IEEE Trans. Signal Process.*, vol. 67, no. 18, pp. 4870–4885, Sep. 2019.
- [29] J. Mei and J. M. F. Moura, "Signal processing on graphs: Causal modeling of unstructured data," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2077–2092, Apr. 2017.
- [30] C. K. Wikle and C. Noel, "A dimension-reduced approach to space-time Kalman filtering," *Biometrika*, vol. 86, no. 4, pp. 815–829, 1999.
- [31] F. Sigrist, H. R. Künsch, and W. A. Stahel, "An autoregressive spatio-temporal precipitation model," *Procedia Environ. Sci.*, vol. 3, pp. 2–7, 2011.
- [32] X. Mao, K. Qiu, T. Li, and Y. Gu, "Spatio-temporal signal recovery based on low rank and differential smoothness," *IEEE Trans. Signal Process.*, vol. 66, no. 23, pp. 6281–6296, Dec. 2018.
- [33] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacević, "Signal recovery on graphs: Variation minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4609–4624, Sep. 2015.
- [34] P. Berger, G. Hannak, and G. Matz, "Graph signal recovery via primal-dual algorithms for total variation minimization," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 842–855, Sep. 2017.
- [35] S. H. Safavi, M. Khatua, N. M. Cheung, and F. Torkamani-Azar, "On sparse graph Fourier transform," Nov. 2018, *arXiv:1811.08609*.
- [36] Q. Lu and G. B. Giannakis, "Probabilistic reconstruction of spatio-temporal processes over multi-relational graphs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 7, pp. 166–176, 2021.
- [37] S. Chen, M. Li, and Y. Zhang, "Sampling and recovery of graph signals based on graph neural networks," Nov. 2020, *arXiv:2011.01412*.

- [38] V. N. Ioannidis, Y. Shen, and G. B. Giannakis, "Semi-blind inference of topologies and dynamical processes over dynamic graphs," *IEEE Trans. Signal Process.*, vol. 67, no. 9, pp. 2263–2274, May 2019.
- [39] S. Dong, P. A. Absil, and K. A. Gallivan, "Graph learning for regularized low-rank matrix completion," in *Proc. 23rd Int. Symp. Math. Theory Netw. Syst. (MTNS)*, 2018, pp. 1–8.
- [40] P. Berger, G. Hannak, and G. Matz, "Efficient graph learning from noisy and incomplete data," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 105–119, 2020.
- [41] Y. Liu, W. Guo, K. You, L. Zhao, T. Peng, and W. Wang, "Graph learning for spatiotemporal signals with long- and short-term characterization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 699–713, 2020.
- [42] A. Javaheri, A. Amini, F. Marvasti, and D. P. Palomar, "Joint signal recovery and graph learning from incomplete time-series," 2023, *arXiv:2312.16940*.
- [43] J. Ying, J. V. de M. Cardoso, and D. P. Palomar, "Does the 1-norm learn a sparse graph under Laplacian constrained graphical models?" Jun. 2020, *arXiv:2006.14925*.
- [44] J. Ying, J. V. de M. Cardoso, and D. P. Palomar, "Nonconvex sparse graph learning under Laplacian constrained graphical model," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 7101–7113.
- [45] M. Razaviyayn, M. Hong, and Z. Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, Jan. 2013.
- [46] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, Feb. 2017.
- [47] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *J. Optim. Theory Appl.*, vol. 109, no. 3, pp. 475–494, Jun. 2001.
- [48] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *J. Roy. Statist. Soc. Ser. B (Methodol.)*, vol. 58, no. 1, pp. 267–288, 1996.
- [49] B. Zaman, L. M. L. Ramos, D. Romero, and B. Beferull-Lozano, "Online topology identification from vector autoregressive time series," *IEEE Trans. Signal Process.*, vol. 69, pp. 210–225, 2021.
- [50] A. Cesa-Bianchi, "VAR-toolbox," version 2, 2020. [Online]. Available: <https://github.com/ambropo/VAR-Toolbox>
- [51] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *J. Mach. Learn. Res.*, vol. 11, no. 80, pp. 2287–2322, 2010.
- [52] K. Qiu, X. Mao, X. Shen, X. Wang, T. Li, and Y. Gu, "Time-varying graph signal reconstruction," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 870–883, Sep. 2017.
- [53] N. Perraudin, A. Loukas, F. Grassi, and P. Vandergheynst, "Towards stationary time-vertex signal processing," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 3914–3918.
- [54] A. J. Laub, *Matrix Analysis for Scientists and Engineers*. Philadelphia, PA, USA: SIAM, 2005.
- [55] R. J. Tibshirani, "The Lasso problem and uniqueness," *Electron. J. Statist.*, vol. 7, pp. 1456–1490, Jan. 2013.



**Amirhossein Javaheri** received the B.Sc. and M.Sc. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2014 and 2016, respectively. He is currently pursuing the dual Ph.D. program in electronic and computer engineering with Sharif University of Technology and The Hong Kong University of Science and Technology. His current research interests include graph learning and signal processing, optimization, machine learning, and data analytics.



**Arash Amini** (Senior Member, IEEE) received the B.Sc. degree in electrical engineering (communications and signal processing) and the B.Sc. degree in petroleum engineering (reservoir), and the M.Sc. and Ph.D. degrees in electrical engineering (communications and signal processing) from Sharif University of Technology, Tehran, Iran, in 2005, 2007, and 2011, respectively. He was a Researcher with the Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland, from 2011 to 2013. He joined as an Assistant Professor with Sharif University of Technology, in 2013, where he has been currently an Associate Professor, since 2018. He has served as an Associate Editor of IEEE SIGNAL PROCESSING LETTERS from 2014 to 2018.



**Farokh Marvasti** (Life Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from Rensselaer Polytechnic Institute, USA, in 1970, 1971, and 1973, respectively. He has worked, consulted, and taught with Bell Labs, University of California Davis, Illinois Institute of Technology, University of London, and Kings College London. He was one of the editors and an Associate Editor of IEEE TRANSACTIONS ON COMMUNICATIONS and IEEE TRANSACTIONS ON SIGNAL PROCESSING, from 1990 to 1997. He has published two books and five book chapters, about 200 Journal papers and several hundred conference papers. He was a Professor with Sharif University of Technology and the Director of Advanced Communications Research Institute (ACRI) before he retired in 2021. He spent his sabbatical leave at the Communications and Information Systems Group of University College London (UCL) in 2013. He received a distinguished award from the Iranian Academy of Sciences in 2014 and a five year term Chair position from Iranian National Science Foundation in 2015. He was also appointed as a Distinguished Researcher by IEEE Iran Chapter in 2018. He has organized two concerts in London, East Meets West in 2003 and Gathering of the Birds in 2023.



**Daniel P. Palomar** (Fellow, IEEE) received the B.Sc. and the Ph.D. degrees in electrical engineering from the Technical University of Catalonia (UPC), Barcelona, Spain, in 1998 and 2003, respectively. He was a Fulbright Scholar with Princeton University, from 2004 to 2006. He is a Professor with the Department of Electronic and Computer Engineering and the Department of Industrial Engineering and Decision Analytics at The Hong Kong University of Science and Technology (HKUST), Hong Kong, where he joined in 2006. He previously held several research appointments, namely, at King's College London (KCL), London, U.K.; Stanford University, Stanford, CA, USA; Telecommunications Technological Center of Catalonia (CTTC), Barcelona, Spain; Royal Institute of Technology (KTH), Stockholm, Sweden; University of Rome "La Sapienza," Rome, Italy; and Princeton University, Princeton, NJ, USA. His current research interests include applications of optimization theory, graph methods, and signal processing in financial systems and big data analytics. He was a recipient of a 2004/06 Fulbright Research Fellowship, the 2004, 2015, and 2020 (as a co-author) Young Author Best Paper Awards by the IEEE Signal Processing Society, the 2015–16 HKUST Excellence Research Award, the 2002/03 best Ph.D. prize in Information Technologies and Communications by the Technical University of Catalonia (UPC), the 2002/03 Rosina Ribalta first prize for the Best Doctoral Thesis in Information Technologies and Communications by the Epson Foundation, and the 2004 prize for the best Doctoral Thesis in Advanced Mobile Communications by the Vodafone Foundation and COIT. He has been a Guest Editor of IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING 2016 Special Issue on "Financial Signal Processing and Machine Learning for Electronic Trading," an Associate Editor of IEEE TRANSACTIONS ON INFORMATION THEORY and of IEEE TRANSACTIONS ON SIGNAL PROCESSING, a Guest Editor of *IEEE Signal Processing Magazine* 2010 Special Issue on "Convex Optimization for Signal Processing," IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS 2008 Special Issue on "Game Theory in Communication Systems," and IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS 2007 Special Issue on "Optimization of MIMO Transceivers for Realistic Communication Networks."