

# Graph Learning for Spatiotemporal Signals with Long- and Short-Term Characterization

Yueliang Liu, *Student Member, IEEE*, Wenbin Guo, *Member, IEEE*, Kangyong You, Lei Zhao, Tao Peng, Wenbo Wang, *Senior Member, IEEE*

**Abstract**—Mining natural associations from high-dimensional spatiotemporal signals plays an important role in various fields including biology, climatology, and financial analysis. However, most existing works have mainly studied time-independent signals without considering the correlations of spatiotemporal signals that achieve high learning accuracy. This paper aims to learn graphs that better reflect underlying data relations by leveraging the long- and short-term characteristics of spatiotemporal signals. First, a spatiotemporal signal model is presented that considers both spatial and temporal relations. In particular, we integrate a low-rank representation and a Gaussian Markov process to describe the temporal correlations. Then, the graph learning problem is formulated as a joint low-rank component estimation and graph Laplacian inference. Accordingly, we propose a low rank and spatiotemporal smoothness-based graph learning method (GL-LRSS), which introduces a spatiotemporal smoothness prior into time-vertex signal analysis. By jointly exploiting the low rank of long-time observations and the smoothness of short-time observations, the overall learning performance can be effectively improved. Experiments on both synthetic and real-world datasets demonstrate substantial improvements in the learning accuracy of the proposed method over the state-of-the-art low-rank component estimation and graph learning methods.

**Index Terms**—Graph learning, spatiotemporal signal, graph signal, low rank, spatiotemporal smoothness.

## I. INTRODUCTION

APPLICATIONS in a variety of fields, from finance and sociology to transportation and sensor networks, rely on statistics, modeling, and processing of spatiotemporal signals. These signals often represent long time series measured over a certain spatial range. Examples include biomedical imaging data [1], video sequences [2], social interactions among individuals [3], and environmental sensing [4]. The usually complex spatiotemporal correlations and interactions can hinder the analysis of spatiotemporal signals.

Graphs can be useful for data analysis due to their ability to provide flexible descriptions in irregular domains. In recent years, graph signal processing (GSP) [5] has provided an engineering paradigm for processing spatiotemporal signals on graphs, establishing time-varying graph signals, based on the

spectral graph theory [6]. For analysis and learning, data may be suitably represented by a graph, and the graph Laplacian matrix, which is equivalent to graph topology, can be used to solve many problems including graph signal compression [7], graph signal reconstruction [8], and graph filtering [9]. Although graph-based methods have been successfully applied, the graph structure is not always available, and straightforward representations (e.g., geographical  $k$ -nearest neighbors) may not adequately capture intrinsic relations among data. Therefore, efficient graph learning methods should be developed to improve the quality and efficiency of data analysis (e.g., trend identification). Extracting underlying relations from observed spatiotemporal signals is essential for their analysis.

In many cases, the collected spatiotemporal signals are highly redundant and thus strongly correlated. To learn a high-quality graph from these spatiotemporal signals, their correlation properties must be thoroughly studied. Recent studies [10]–[12] provided effective ways of characterizing correlation properties by assuming spatiotemporal signals to be approximately low-rank and have short-term stability. The corresponding results showed methodological superiority in signal processing tasks. However, most existing graph learning methods neglect the long-term correlation of signals, for example, by modeling the spatiotemporal signals locally [13] or by treating the successive signals independently [14], [15]. Although the mentioned graph learning methods achieve satisfying results, there is still much room for improvement. Therefore, in this paper, we propose an enhanced graph learning method that fully leverages long- and short-term correlations in spatiotemporal signals.

### A. Related works

Our work involves joint graph learning and low-rank component estimation. Several approaches have been proposed to address these two problems, and detailed surveys are available in [16]–[18]. However, graph learning and low-rank approximation have not been jointly studied.

For low-rank component estimation, various methods have approximated spatiotemporal signals as low-rank matrices [19], [20], achieving satisfactory results by assuming that the matrix collecting the time sequences is approximately low-rank. Recently, GSP approaches were proposed to recover low-rank components by using spectral graph regularization [16], [21], [22]. These approaches incorporated graph smoothness on low-rank matrices and improved both clustering and recovery performance. However, the graphs are predefined based on

This work was supported by the National Natural Science Foundation of China (61271181, 61571054), the Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory Foundation and BUPT Excellent Ph.D. Students Foundation (Grant CX2018101) (*Corresponding author: Wenbin Guo.*)

Y. Liu and W. Guo are with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory, Shijiazhuang 050000, China (e-mail: {liuyueliang, gw}@bupt.edu.cn).

the geometric distance in these methods, possibly undermining the accuracy for subsequent analyses.

For graph learning, early studies provided graphical models by neighborhood selection per node [23]. For stability under noise, graphical lasso methods have been used to estimate the inverse covariance or precision matrix [24]–[26]. The fast-growing GSP allows solving graph learning problems by implementing methods related to Gaussian Markov Random Fields (GMRFs) with the precision matrix defined by a graph Laplacian. For smoothing the graph signals, smoothness-based methods have been adopted during graph inference. Dong *et al.* [14] first proposed a valid combinatorial graph Laplacian (CGL) learning method under a smooth graph representation. Then, Kalofolias [15] reformulated the problem in terms of the adjacency matrix and proposed a computationally efficient algorithm. To generalize the restriction of the precision matrix to be a CGL, Egilimez *et al.* [27] identified a GMRF model whose precision matrix could be any of multiple types of graph Laplacians. Alternative smoothness-based approaches have also been effective [13], [28], [29], with the methodological implementation in [13] and [28] being based on space-time modeling and edge selection, respectively, whereas a theoretical analysis of the reconstruction error was provided in [29]. These methods learned graphs from smooth graph signals, while a few other works added assumptions on the graph dynamics for time-varying graph learning. For instance, dynamic graphs have been learned by assuming that the graph structure changes smoothly over time [30], whereas a method considering the sparseness of the graph variation was also proposed in [31].

Another family of graph learning approaches adopts a physics perspective for modeling graph signals. In these cases, observations are modeled considering a physical process for the graph, such as diffusion [32]–[35] and causality [36]–[38]. Segarra *et al.* [32] and Pasdeloup *et al.* [33] identified a graph from stationary observations assumed to be generated by a diffusion process. Shafipour *et al.* [34] generalized this assumption by exploring a graph learning method that could be applied to non-stationary graph signals. Thanou *et al.* [35] proposed graph learning considering that the graph signals result from heat diffusion. Causality-based methods focus on the asymmetric adjacency matrix corresponding to a directed graph. Mei *et al.* [36] considered a causal graph process to characterize a time series and applied it to temperature analysis. Under a structural equation model, Baingana *et al.* [37] proposed a recursive least-squares estimator to track both the signal state and graph topology. Similarly, Shen *et al.* [38] described nonlinear dependencies of signals via structural vector autoregressive models and developed an efficient estimator to infer a sparse graph. While these graph learning methods can provide meaningful graphs from time series, long-term correlations (i.e., low rank) have been neglected.

### B. Contributions

For high-quality graph learning, we propose a method that considers the low rank and local smoothness of spatiotemporal signals. Low-rank component estimation allows to improve the quality of the learned graph, with the low-rank compo-

nent being better estimated from a refined graph. The main contributions of this study can be summarized as follows:

- 1) To the best of our knowledge, this is the first model of spatiotemporal signals integrating a low-rank representation and a first-order Gaussian Markov process.
- 2) We introduce a spatiotemporal smooth prior to the time-varying graph signal to facilitate graph learning.
- 3) Graph learning is formulated as a joint graph refinement and low-rank component estimation problem solved using the proposed graph learning method based on low-rank approximation and spatiotemporal smoothness (GL-LRSS), which applies the alternating direction method of multipliers (ADMM) and alternating minimization.
- 4) We provide visual and quantitative comparisons with state-of-the-art low-rank component estimation and graph learning methods. The extensive experimental results on synthetic and real-world datasets demonstrate the superiority and effectiveness of the proposed GL-LRSS.

### C. Comparison with state-of-the-art methods

Regarding graph signal representation, the proposed GL-LRSS extends smoothness-based graph learning. Although the methods in [12] and [14] are the most related to the proposed GL-LRSS, they neglect the local temporal correlations of spatiotemporal signals. Specifically, the smoothness-based method (e.g., [14], [15], [27]) uses a GMRF model which is mainly suitable for time-independent signals, and the method in [12] uses a low-rank signal model which lacks local signal characterization. The proposed GL-LRSS adopts a different model from these similar methods in the following aspects:

- The novel model for spatiotemporal signals considers local and global correlations. By combining a low-rank representation and a first-order Gaussian Markov process, the proposed model can describe multiple types of time correlations.
- Although both the method in [12] and the proposed GL-LRSS aim to jointly estimate the graph structure and low-rank components, the method in [12] obtains its final optimization by directly combining the objective functions of two estimation subproblems. Our method, on the other hand, formulates the optimization problem based on Bayesian inference and introduces a new regularization term called spatiotemporal smoothness for graph learning.

The remainder of this paper is organized as follows. Section II presents the notation and preliminaries of GSP. In Section III, we propose the low-rank graph-based model and the corresponding spatiotemporal smoothness prior. In Section IV, we formulate the graph learning problem as a joint low-rank component and graph topology estimation and propose the GL-LRSS to solve the optimization problem alternately. In Section V, the GL-LRSS performance on both synthetic and real-world datasets is reported and compared with that of baseline methods. A discussion is presented in Section VI, and we draw conclusions in Section VII.

## II. NOTATION AND PRELIMINARIES

### A. Notations

Throughout this paper, lowercase letters (e.g.,  $\alpha$ ,  $\beta$ ), lowercase boldface letters (e.g.,  $\mathbf{x}$ ,  $\mathbf{u}$ ), and uppercase boldface letters

TABLE I  
LIST OF SYMBOLS AND THEIR MEANING

Symbols	Meaning
$\mathcal{G} \mid \mathbf{L} \mid \mathcal{L}^N$	weighted graph   graph Laplacian matrix   set of CGLs
$\mathcal{V} \mid \mathcal{E}$	vertex set   edge set
$N \mid M$	number of vertices   number of time instants
$\mathbf{I} \mid \mathbf{W} \mid \mathbf{D}$	identity matrix   adjacency matrix   degree matrix
$\mathbf{U} \mid \mathbf{\Lambda}$	eigenvector matrix   eigenvalue matrix of $\mathbf{L}$
$\mathbf{0} \mid \mathbf{1}$	column vector of zeros   column vector of ones
$\mathbf{X}^{-1} \mid \mathbf{X}^\dagger$	inverse of $\mathbf{X}$   pseudo-inverse of $\mathbf{X}$
$\mathbf{X}^T \mid \mathbf{x}^T$	transpose of $\mathbf{X}$   transpose of $\mathbf{x}$
$(\mathbf{X})_{ij}$	entry of $\mathbf{X}$ at $i$ -th row and $j$ -th column
$\mathbf{x}_i$	$i$ -th entry of $\mathbf{x}$
$\geq (\leq)$	element-wise greater (less) than or equal to operator
$\mathbf{X} \succeq \mathbf{0}$	$\mathbf{X}$ is a positive semidefinite matrix
$\text{tr} \mid \text{vec}$	trace operator   vectorization operator
$\otimes \mid \langle \cdot, \cdot \rangle$	Kronecker product operator   inner product operator
$\text{diag}(\mathbf{x})$	diagonal matrix formed by elements of $\mathbf{x}$
$p(\mathbf{x})$	probability density function of random vector $\mathbf{x}$
$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$	zero-mean multivariate Gaussian with covariance $\mathbf{\Sigma}$
$\ \mathbf{X}\ _*$	nuclear norm of $\mathbf{X}$
$\ \mathbf{x}\ _1 \mid \ \mathbf{X}\ _1$	sum of absolute values of all elements ( $l_1$ -norm)
$\ \mathbf{x}\ _2^2 \mid \ \mathbf{X}\ _F^2$	sum of squared values of elements

(e.g.,  $\mathbf{X}, \mathbf{L}$ ) denote scalars, vectors, and matrices, respectively. Unless otherwise stated, calligraphic capital letters (e.g.,  $\mathcal{E}$  and  $\mathcal{L}$ ) represent sets. Additional notation is listed in Table I.

### B. Graph Laplacian

We consider an undirected weighted graph with non-negative edge weights and no self-loops. Let  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$  be an  $N$ -vertex weighted graph, where  $\mathcal{V} = (v_1, \dots, v_N)$  is the vertex set and  $\mathcal{E}$  is the edge set. The adjacency matrix  $\mathbf{W}$  is an  $N \times N$  symmetric matrix. The CGL of  $\mathcal{G}$  is defined as  $\mathbf{L} = \mathbf{D} - \mathbf{W}$ , where diagonal matrix  $\mathbf{D}$  denotes the degree matrix with its  $i$ th diagonal entry indicating the degree of vertex  $i$  (i.e.,  $\text{diag}(\mathbf{D})_i = \sum_{j=1}^N W_{ij}$ ).  $\mathcal{L}$  is the set of all valid  $N \times N$  CGLs for matrix  $\mathbf{L}$ :

$$\mathcal{L} = \left\{ \mathbf{L} \mid \mathbf{L} \succeq \mathbf{0}, (\mathbf{L})_{ij} = (\mathbf{L})_{ji} \leq 0, i \neq j, \text{ and } \mathbf{L} \cdot \mathbf{1} = \mathbf{0} \right\}. \quad (1)$$

As the CGL is a real symmetric positive semidefinite matrix, its eigenvalues are non-negative. Let the CGL eigendecomposition be  $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ , where  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)$  and  $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N]$  are matrices containing the eigenvalues and eigenvectors, respectively. The graph frequency spectrum is defined by the ascending array of eigenvalues  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$ , referred to as the graph frequency, and orthogonal eigenvectors  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N$  are the harmonics associated with the graph frequencies. In addition, the CGL of a connected graph always has a zero eigenvalue (i.e.,  $\lambda_1 = 0$ ) corresponding to eigenvector  $\mathbf{u}_1 = 1/\sqrt{N} \cdot \mathbf{1}$ .

### C. Smooth Graph Signals

For graph signal  $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$ , where  $x_i$  is attached to vertex  $v_i$ , its frequency component is defined by the graph Fourier transform denoted as  $\hat{\mathbf{x}} = \mathbf{U}^T \mathbf{x}$ . The frequency components corresponding to higher eigenvalues indicate larger variations between the signals of the vertices, whereas those corresponding to small eigenvalues are relatively smooth. Many real-world datasets have graph signals

that change smoothly between the connected vertices. Such smoothness property indicates the graph signal variation with respect to the underlying graph. To quantify the smoothness of signal  $\mathbf{x}$ , a typical metric can be written in the graph Laplacian quadratic form [6]:

$$S(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{(i,j) \in \mathcal{I}} (\mathbf{W})_{i,j} [x_j - x_i]^2, \quad (2)$$

where  $\mathcal{I} = \{(i,j) \mid (v_i, v_j) \in \mathcal{E}\}$  is the set of index pairs of connected vertices. Eq. (2) measures the total variation of connected vertices associated with edge set  $\mathcal{E}$ . In the vertex domain, smaller values of Eq. (2) indicate higher signal smoothness in the graph.

### D. Correlations in Spatiotemporal Signals

Spatiotemporal signals can be viewed as time-varying graph signals in a graph structure of the observation sites. These signals are usually highly redundant and thus strongly correlated. Global and local consistency principles have been identified for data description [40], unveiling long- and short-term correlations.

*Long-term correlation:* The global consistency indicates that spatiotemporal signals are usually correlated globally [11], [19]. Such correlation describes the space and time commonalities over a long time and can be interpreted as temporal sequences of the form  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M$  generated from limited patterns. Hence, spatiotemporal signals  $\mathbf{X}$  can be approximately low-rank [16], [20].

*Short-term correlation:* Spatiotemporal signals can be locally correlated [13], [40], as observations from a site can be correlated across neighboring time instants for the temporal sequences to vary smoothly over time. Likewise, at a given instant, nearby observation sites can exhibit spatial correlations corresponding to similar values. These two types of short-term correlations are respectively determined by temporal smoothness and spatial smoothness.

GSP methods are based upon spatial and temporal smoothness. Although spatial smoothness has been widely applied [8], [15], [16], few studies have leveraged temporal smoothness [41], [42]. By combining spatial and temporal smoothness, we introduce the concept of spatiotemporal smoothness and propose the GL-LRSS, which also considers long-term correlations. Spatiotemporal smoothness describes short-term characteristics of time-varying graph signals.

## III. LONG- AND SHORT-TERM CHARACTERIZATION OF SPATIOTEMPORAL SIGNALS

### A. Signal Representation

Spatiotemporal signals exhibit global and local correlations over the long and short terms, respectively. To describe these correlations, we propose a model that characterizes spatiotemporal signals from the local and global perspectives.

Consider an  $N$ -vertex graph with graph Laplacian matrix  $\mathbf{L} \in \mathbb{R}^{N \times N}$ . A spatiotemporal signal can be expressed by matrix  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M] \in \mathbb{R}^{N \times M}$ , where  $M$  is the number of time instants. In the proposed model, the observed signal is modeled as

$$\mathbf{Y} = \mathbf{X} + \mathbf{N}, \quad (3)$$

where  $\mathbf{N}$  denotes the additive Gaussian white noise.

1) *Short-term signal characterization*: Considering dependencies in the neighboring space and time, we characterize the observed signal from a local viewpoint as follows:

$$\mathbf{y}_t = \mathbf{x}_t + \mathbf{n}_t, \quad (4)$$

$$\mathbf{x}_t = R\mathbf{x}_{t-1} + \mathbf{v}_t, \quad (5)$$

where  $\mathbf{y}_t \in \mathbb{R}^N$  is the observation at the  $t$ th time instant, and  $\mathbf{n}_t \in \mathbb{R}^N$  denotes the multivariate Gaussian noise with zero mean and covariance  $\sigma_n^2 \mathbf{I}_N$ . The state transition matrix  $R$  is defined as a general diagonal matrix  $R = \text{diag}(c_1, c_2, \dots, c_N)$ , where  $c_i$  denotes the coefficient of the  $i$ th observation site and ranges from 0 to 1. Each  $c$  represents the autocorrelation coefficient that describes the time correlation of data with a delayed copy (one-time lag in this model) of itself, and can be obtained in advance.

To represent signals residing on graphs and identify structures in data, we introduce graph-based process variable  $\mathbf{v}_t$ :

$$\mathbf{v}_t = \mathbf{U}_{(r)} \mathbf{z}_t, \quad (6)$$

where  $\mathbf{U}_{(r)} \in \mathbb{R}^{N \times r}$  contains the first  $r$  ( $r \leq N$ ) eigenvectors of the graph Laplacian matrix, and  $\mathbf{z}_t \in \mathbb{R}^r$  is assumed to follow a multivariate Gaussian distribution,  $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \Lambda_{(r)}^\dagger)$ , with precision matrix  $\Lambda_{(r)}^\dagger$  being the Moore-Penrose pseudoinverse of the matrix that contains the first  $r$  eigenvalues. This definition leads to a smooth graph signal representation and provides an intuitive relationship between the graph structure and graph signal. According to Eq. (6), the assumption about  $\mathbf{z}_t$  and the basis vector  $\mathbf{U}_{(r)}$  leads to a multivariate Gaussian distribution for  $\mathbf{v}_t$  (i.e.,  $\mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \tilde{\mathbf{L}}^\dagger)$ , with  $\tilde{\mathbf{L}}^\dagger = \mathbf{U}_{(r)} \Lambda_{(r)}^\dagger \mathbf{U}_{(r)}^T$ ), such that the representation of time-varying signals reflects the graph topology. Furthermore, as is shown in Section III-B-1, the short-term characterizations in Eqs. (5) and (6) lead to the local smoothness of spatiotemporal signals.

2) *Long-term signal characterization*: As mentioned in Section II-D, spatiotemporal signals are approximately low-rank in practice. Thus, it is realistic and efficient to treat spatiotemporal signals from a global viewpoint. Considering the spatiotemporal signals across  $M$  time instant  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M]$  with initialization  $\mathbf{x}_0 = \mathbf{v}_0$ , we obtain the matrix form of Eq. (4) given by Eq. (3).

For a convenient signal representation of  $\mathbf{X}$ , a normal distribution  $\mathcal{N}(\mu, \sigma_R^2)$  is used to model temporal correlation coefficients  $c_1, \dots, c_N$ , such that correlation matrix  $R$  can be decomposed as  $R = \mu \mathbf{I} + \Delta R$ , where  $\mu \mathbf{I}$  corresponds to the mean and  $\Delta R$  represents the fluctuations around  $\mu \mathbf{I}$ . By applying both the decomposition of  $R$  and Eqs. (6) to (5), the spatiotemporal signal  $\mathbf{X}$  in Eq. (3) becomes

$$\mathbf{X} = \mathbf{U}_{(r)} \mathbf{Z} + \Phi, \quad (7)$$

where  $\mathbf{Z} = [\mu \mathbf{z}_0 + \mathbf{z}_1, \dots, \mu^M \mathbf{z}_0 + \mu^{M-1} \mathbf{z}_1 + \dots + \mathbf{z}_M]$ , and  $\Phi$  is a complex perturbation term related to  $\Delta R$ . Mathematically, Eq. (7) is the matrix expression of Eq. (5) that establishes a relation between the long- and short-term signals. As discussed in Section III-B-2, spatiotemporal signals under the long-term characterization are approximately low-rank.

## B. Long- and Short-Term Properties in Signal Representation

Under the local and global signal representations described in Section III-A, we explore the long- and short-term properties (i.e., low-rank property and spatiotemporal smoothness, respectively) of spatiotemporal signals.

1) *Short-term property*: As discussed in Section III-A-1, the model given by Eq. (5) is analogous to a first-order vector autoregressive model. It naturally promotes the temporal smoothness of the signal. On the other hand, as small eigenvalues correspond to smooth eigenvectors on the graph, the selection of  $\mathbf{U}_{(r)}$  in Eq. (6) as the basis vector supports the spatial smoothness of the signal. Therefore, the graph signal under the short-term characterizations in Eqs. (5) and (6) exhibit spatiotemporal smoothness. Moreover, we show in Section IV-A that our proposed method enforces such spatiotemporal smoothness property in graph learning. The definition of spatiotemporal smoothness is given next.

**Definition 1** (Spatiotemporal smoothness). *The weighted time differences of spatiotemporal signals are smooth with respect to the graph structure. Based on (2), spatiotemporal smoothness can be defined as*

$$S(\mathbf{x}_t - R\mathbf{x}_{t-1}) = (\mathbf{x}_t - R\mathbf{x}_{t-1})^T \mathbf{L} (\mathbf{x}_t - R\mathbf{x}_{t-1}) = \mathbf{v}_t^T \mathbf{L} \mathbf{v}_t. \quad (8)$$

Considering the signals across multiple time instants, the corresponding spatiotemporal smoothness in *Definition 1* can be expressed in a matrix form as follows.

**Definition 2** (Weighted difference operator). *The weighted difference operator of graph signal  $\mathbf{X}$  is  $\mathcal{D}(\mathbf{X}) = \mathbf{X} - R\mathbf{X}\mathbf{B}$ , where  $\mathbf{B}$  is a shift operator given by*

$$\mathbf{B} = \begin{bmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & 0 & \ddots & \\ & & & \ddots & 1 \\ & & & & 0 \end{bmatrix}_{M \times M}, \quad (9)$$

The weighted difference signal is equal to  $\mathcal{D}(\mathbf{X}) = [\mathbf{x}_1, \mathbf{x}_2 - R\mathbf{x}_1, \mathbf{x}_3 - R\mathbf{x}_2, \dots, \mathbf{x}_M - R\mathbf{x}_{M-1}]$ . Hence, the matrix expression of spatiotemporal smoothness is

$$S(\mathcal{D}(\mathbf{X})) = \sum_{t=1}^M S(\mathbf{x}_t - R\mathbf{x}_{t-1}) \stackrel{(a)}{=} \text{tr} \left( \mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}) \right), \quad (10)$$

where (a) follows from *Definition 2* and the smoothness metric  $S(\cdot)$  in (2).

2) *Long-term property*: Besides the above mentioned short-term properties, the following discussion shows the approximately low-rank property of  $\mathbf{X}$  considering Eq. (7).

The first term in Eq. (7) resembles the formulation of principal component analysis (PCA), which is the most popular technique for approximating low-rank components. According to conventional PCA, the low-rankness of  $\mathbf{X}$  mainly depends on the number of basis vectors, i.e., the number of columns of  $\mathbf{U}_{(r)}$ . However, due to the perturbation in Eq. (7), the rank of  $\mathbf{X}$  is also affected by  $\Phi$ . Specifically, a small  $\sigma_R$  tends to reduce the effect of  $\Phi$ , and hence  $\mathbf{X}$  is more likely to be low-rank, whereas a large  $\sigma_R$  tends to weaken the low-rank property of  $\mathbf{X}$ . When  $\sigma_R = 0$  (i.e.,  $\mathbf{X} = \mathbf{U}_{(r)} \mathbf{Z}$ ),  $\mathbf{X}$  is a low-rank matrix. Therefore, the proposed model given by (3) and

(7) can be viewed as an approximately low-rank representation that roughly characterizes long-term correlations of the signal. To precisely constrain the low-rank property of  $\mathbf{X}$ , we introduce a nuclear norm for the optimization problem.

#### IV. GRAPH LEARNING BASED ON LOW RANK AND SPATIOTEMPORAL SMOOTHNESS (GL-LRSS)

In this section, we propose an efficient graph learning method by jointly exploiting the local smoothness and global correlation of spatiotemporal signals. We first formulate the graph learning problem and then propose an optimization algorithm, GL-LRSS, which is based on the ADMM and alternating minimization. Finally, the computational complexity of the proposed algorithm is briefly analyzed.

##### A. Problem Formulation

As mentioned in Section III-A, the graph structural information is encoded in the covariance of process variable  $\mathbf{v}_t$ . In terms of graph structure recovery, our method can be regarded as inverse covariance matrix estimation. For probabilistic inference, we first introduce the following weighted difference observation:

$$\mathbf{d}_t = \mathbf{y}_t - R\mathbf{y}_{t-1} = \mathbf{v}_t + \mathbf{n}_t - R\mathbf{n}_{t-1}, \quad (11)$$

with initialization  $\mathbf{d}_1 = \mathbf{y}_1$ . Based on the distribution of Gaussian noise, the conditional probability of  $\mathbf{d}_t$  given  $\mathbf{v}_t$  satisfies

$$\mathbf{d}_t | \mathbf{v}_t \sim \mathcal{N}(\mathbf{v}_t, \sigma_n^2 (\mathbf{I}_N + RR^T)). \quad (12)$$

Given the weighted difference observation  $\mathbf{d}_t$  and the Gaussian prior distribution of  $\mathbf{v}_t$ , we can compute a maximum a posteriori (MAP) estimate of core component  $\mathbf{v}_t$ . Specifically, by applying Bayes' rule, the MAP estimate of  $\mathbf{v}_t$  is given by

$$\begin{aligned} \mathbf{v}_{t, \text{MAP}}(\mathbf{d}_t) &:= \arg \max_{\mathbf{v}_t \in \mathbb{R}^N} p(\mathbf{v}_t | \mathbf{d}_t) = \arg \max_{\mathbf{v}_t \in \mathbb{R}^N} p(\mathbf{d}_t | \mathbf{v}_t) p(\mathbf{v}_t) \\ &= \arg \min_{\mathbf{v}_t \in \mathbb{R}^N} (-\log p_E(\mathbf{d}_t - \mathbf{v}_t) - \log p_V(\mathbf{v}_t)) \\ &= \arg \min_{\mathbf{v}_t \in \mathbb{R}^N} (\mathbf{d}_t - \mathbf{v}_t)^T \mathbf{W}^{-1} (\mathbf{d}_t - \mathbf{v}_t) + \alpha \mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t, \end{aligned} \quad (13)$$

where  $\mathbf{W} = \mathbf{I}_N + RR^T$  and  $\alpha$  is a constant parameter proportional to the noise variance,  $\sigma_n^2$ . However, the objective function in Eq. (13) is difficult to process [13], especially for the unknown correlation matrix  $R$ . To obtain a solution, a relaxation procedure can be adopted for the problem. By leveraging the diagonal property of matrix  $R$  and inequality

$$(\mathbf{d}_t - \mathbf{v}_t)^T \mathbf{W}^{-1} (\mathbf{d}_t - \mathbf{v}_t) \geq \lambda_{\min}(\mathbf{W}^{-1}) \|\mathbf{d}_t - \mathbf{v}_t\|_2^2, \quad (14)$$

we obtain a relaxed MAP estimation:

$$\mathbf{v}_{t, \text{MAP}}(\mathbf{d}_t) := \arg \min_{\mathbf{v}_t \in \mathbb{R}^N} \|\mathbf{d}_t - \mathbf{v}_t\|_2^2 + \alpha \mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t. \quad (15)$$

In Eq. (15), the Laplacian quadratic term is the same as that of Eq. (8). Therefore, it verifies that the proposed method enforces the spatiotemporal smoothness in graph learning.

Considering observations at  $M$  time instants  $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_M] \in \mathbb{R}^{N \times M}$ , we focus on 1) learning the graph Laplacian matrix that is equivalent to the graph structure and 2) improving the low-rank component estimation. By imposing

additional constraints on the graph Laplacian  $\mathbf{L}$  and the low-rank component  $\mathbf{X}$ , we propose to solve the problem of (15) using the following objective function, given in a matrix form:

$$\begin{aligned} (\text{P1}) \quad & \min_{\mathbf{X} \in \mathbb{R}^{N \times M}, \mathbf{L}} Q_1(\mathbf{L}, \mathbf{X}) \\ \text{s.t.} \quad & Q_1(\mathbf{L}, \mathbf{X}) = \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) \\ & + \beta \|\mathbf{L}\|_F^2 + \gamma \|\mathbf{X}\|_*, \\ & \mathbf{L} \in \mathcal{L}, \text{tr}(\mathbf{L}) = N, \end{aligned}$$

where  $\alpha$ ,  $\beta$ , and  $\gamma$  are positive regularization parameters corresponding to the regularization terms. The first regularization term,  $\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}))$ , induces the spatiotemporal smoothness encoded in Eq. (10). Together with the trace constraint that aims to avoid trivial solutions, the second regularization term,  $\|\mathbf{L}\|_F^2$ , controls the sparsity of the off-diagonal entries in  $\mathbf{L}$  (i.e., the edge weights of the graph). To promote long-term correlations, we impose nuclear norm  $\|\mathbf{X}\|_*$ , which is defined as the sum of the singular values of  $\mathbf{X}$  and corresponds to the convex envelope of  $\text{rank}(\mathbf{X})$ . The last Laplacian constraint guarantees that the learned graph Laplacian is a valid CGL that satisfies Eq. (1).

Note that in problem (P1), we particularly introduce two regularization terms, i.e.,  $\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}))$  and  $\|\mathbf{X}\|_*$  to characterize the correlation properties of spatiotemporal signals. Although these two terms promote the correlation of spatiotemporal signals from the local and global perspectives, respectively, they compensate each other to deduce a meaningful graph, as detailed below.

- Regularization term  $\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}))$  is derived from the proposed signal representation for graph learning. This term encodes spatial and temporal correlations of  $\mathbf{X}$  in graph Laplacian  $\mathbf{L}$  and weighted difference operator  $\mathcal{D}$ , respectively, while enforcing the weighted difference signal to be smooth on the graph. Unlike differential smoothness [11], this term contains a general correlation matrix  $R$  that considers the varying temporal evolution of data at distinct observation sites. As demonstrated in real experiments, when proper matrix  $R$  is known a priori, the graph learning performance can be further improved.
- As data from many applications have the low-rank property, we utilized  $\|\mathbf{X}\|_*$  to improve the low-rank approximation. The nuclear norm directly forces spatiotemporal signal  $\mathbf{X}$  to achieve a low rank, thus compensating for the limitation of the proposed model in terms of long-term signal characterization, where the low-rank property of the spatiotemporal signal is partially depicted through the term resembling PCA in Eq. (7). Moreover, nuclear norm  $\|\mathbf{X}\|_*$  can increase the graph learning performance, as verified experimentally.

In the optimization of (P1), the graph Laplacian interacts with the low-rank component. Our hypothesis is that accurate low-rank component estimation improves the quality of the learned graph, which in turn improves the low-rank component estimation. Therefore, we adopted an alternating minimization framework that iteratively refines the graph topology and esti-

---

**Algorithm 1 : Graph learning based on low rank and spatiotemporal smoothness (GL-LRSS)**


---

**Input:** Observations  $\mathbf{Y}$ , local correlation  $R$ , regularization parameters  $\alpha, \beta, \gamma$ , maximum iteration  $K$ , threshold  $\varepsilon$ .

- 1: Initialization:  $\mathbf{X}^0 = \mathbf{Y}$ ,  $k = 1$ ;
- 2: **repeat**
- 3:   1) Graph topology refinement:  
 $\mathbf{L}^{k+1} = G(\mathbf{X}^k, \mathbf{Y})$  by (19)-(21)
- 4:   2) Low-rank component estimation:  
 $\mathbf{X}^{k+1} = C(\mathbf{L}^{k+1}, \mathbf{Y})$  by (25)-(27)
- 5:   3)  $(\hat{\mathbf{L}}, \hat{\mathbf{X}}) = (\mathbf{L}^k, \mathbf{X}^k)$ ,  $k = k + 1$ ;
- 6: **until**  $k = K$  or  $|Q_1(\mathbf{L}^k, \mathbf{X}^k) - Q_1(\mathbf{L}^{k+1}, \mathbf{X}^{k+1})| < \varepsilon$

**Output:** Refined graph  $\hat{\mathbf{L}}$ , low-rank component  $\hat{\mathbf{X}}$ .

---

mates the low-rank components. This hypothesis was validated through experiments on synthetic dataset.

### B. Optimization algorithm

The formulation in (P1) establishes a biconvex optimization problem, that is, it is a convex problem with respect to  $\mathbf{L}$  when  $\mathbf{X}$  is fixed and vice versa. We propose the GL-LRSS to solve the optimization problem via alternating optimization. At each step, one variable is optimized while keeping the other variables constant. The iterative procedure is given by

1.  $G(\mathbf{X}, \mathbf{Y}) \triangleq \arg \min_{\mathbf{L}} Q_1(\mathbf{L}, \mathbf{X})$ , (S<sub>L</sub>)  
s.t.  $\mathbf{L} \in \mathcal{L}$ ,  $\text{tr}(\mathbf{L}) = N$ .
2.  $C(\mathbf{L}, \mathbf{Y}) \triangleq \arg \min_{\mathbf{X} \in \mathbb{R}^{N \times M}} Q_1(\mathbf{L}, \mathbf{X})$ . (S<sub>X</sub>)

By iteratively refining the graph from the low-rank representation and estimating the low-rank component using the learned graph, we obtain the final solution of (P1) by alternating minimization. The detailed procedure for solving (P1) is shown as follows.

1) *Graph refinement in subproblem (S<sub>L</sub>):* Notice that (S<sub>L</sub>) is a strictly convex problem under convex constraints, because the Hessian matrix of the objective function,  $2\beta\mathbf{I}_N$ , is positive definite. To solve this constrained convex problem, we apply the ADMM method [44]. Specifically, we reformulate problem (P1) with respect to the graph Laplacian  $\mathbf{L}$  as follows:

$$\begin{aligned} \min_{\mathbf{L} \in \mathcal{L}^*} \quad & \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) + \beta \|\mathbf{L}\|_F^2, \\ \text{s.t.} \quad & \mathbf{L} - \mathbf{Z} = 0, \mathbf{Z} \in \mathcal{L}^*, \end{aligned} \quad (16)$$

where  $\mathbf{Z}$  is an auxiliary variable matrix and  $\mathcal{L}^*$  is expressed as

$$\mathcal{L}^* = \{\mathbf{L} | \mathbf{L} \succeq 0, L_{ji} = L_{ij} \leq 0, i \neq j, \text{ and } \mathbf{L} \cdot \mathbf{1} = \mathbf{0}, \text{tr}(\mathbf{L}) = N\}. \quad (17)$$

Therefore, the augmented Lagrangian of (16) is given by

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{L}, \mathbf{Z}, \mathbf{\Xi}) = & \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) + \beta \|\mathbf{L}\|_F^2 \\ & + \langle \mathbf{\Xi}, \mathbf{Z} - \mathbf{L} \rangle + \frac{\rho}{2} \|\mathbf{Z} - \mathbf{L}\|_F^2, \end{aligned} \quad (18)$$

where  $\mathbf{\Xi}$  is the Lagrange multiplier,  $\langle \cdot, \cdot \rangle$  denotes the inner product of matrices, and  $\rho > 0$  is a prescribed penalty

parameter. We use the following formulas to update  $\mathbf{L}$ ,  $\mathbf{Z}$ , and  $\mathbf{\Xi}$  to find a saddle point of (18)

$$\mathbf{L}^{k+1} = \arg \min_{\mathbf{L} \in \mathcal{L}^*} \mathcal{L}_\rho(\mathbf{L}, \mathbf{Z}^k, \mathbf{\Xi}^k), \quad (19)$$

$$\mathbf{Z}^{k+1} = \arg \min_{\mathbf{Z} \in \mathcal{L}^*} \mathcal{L}_\rho(\mathbf{L}^{k+1}, \mathbf{Z}, \mathbf{\Xi}^k), \quad (20)$$

$$\mathbf{\Xi}^{k+1} = \mathbf{\Xi}^k + \rho(\mathbf{Z}^{k+1} - \mathbf{L}^{k+1}). \quad (21)$$

Setting the derivatives of Eqs. (19) and (20) with respect to  $\mathbf{L}$  and  $\mathbf{Z}$ , respectively, equal to zero, we obtain the following solutions:

$$\mathbf{L}^{k+1} = \frac{\rho \mathbf{Z}^k + \mathbf{\Xi}^k - \alpha \mathcal{D}(\mathbf{X}) \mathcal{D}(\mathbf{X})^T}{2\beta + \rho}, \quad \mathbf{Z}^{k+1} = P_{\mathcal{L}^*} \left( \mathbf{L}^{k+1} - \frac{1}{\rho} \mathbf{\Xi}^k \right), \quad (22)$$

where  $P_{\mathcal{L}^*}(\cdot)$  denotes the Euclidean projection onto set  $\mathcal{L}^*$ .

2) *Low-rank component estimation in subproblem (S<sub>X</sub>):* The first two terms of (P1) are differentiable, and the third term of  $\mathbf{X}$  is proximal. We apply ADMM to solve problem (S<sub>X</sub>). First, we provide an equivalent formulation of (P1) with respect to  $\mathbf{X}$ :

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{P} \in \mathbb{R}^{N \times M}} \quad & \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) + \gamma \|\mathbf{P}\|_*, \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{P}. \end{aligned} \quad (23)$$

The objective function is split into two parts by introducing the linear equality constraint. Then, the augmented Lagrangian of (23) is given by

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{X}, \mathbf{P}, \mathbf{Q}) = & \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) \\ & + \gamma \|\mathbf{P}\|_* + \langle \mathbf{Q}, \mathbf{X} - \mathbf{P} \rangle + \frac{\rho}{2} \|\mathbf{X} - \mathbf{P}\|_F^2, \end{aligned} \quad (24)$$

where  $\mathbf{Q}$  is the Lagrange multiplier. Based on the augmented Lagrangian in (24), the solution is obtained iteratively, as follows:

$$\mathbf{X}^{k+1} = \arg \min_{\mathbf{X} \in \mathbb{R}^{N \times M}} \mathcal{L}_\rho(\mathbf{X}, \mathbf{P}^k, \mathbf{Q}^k), \quad (25)$$

$$\mathbf{P}^{k+1} = \arg \min_{\mathbf{P} \in \mathbb{R}^{N \times M}} \mathcal{L}_\rho(\mathbf{X}^{k+1}, \mathbf{P}, \mathbf{Q}^k), \quad (26)$$

$$\mathbf{Q}^{k+1} = \mathbf{Q}^k + \rho(\mathbf{X}^{k+1} - \mathbf{P}^{k+1}). \quad (27)$$

According to (24), the subproblem of (25) can be rewritten as

$$\begin{aligned} \mathbf{X}^{k+1} = & \arg \min_{\mathbf{X} \in \mathbb{R}^{N \times M}} \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) \\ & + \frac{\rho}{2} \|\mathbf{X} - \mathbf{P}^k + \mathbf{Q}^k / \rho\|_F^2. \end{aligned} \quad (28)$$

The expression in (28) is a differentiable convex optimization problem that admits a closed-form solution. Using  $\text{vec}(\mathbf{AXB}) = (\mathbf{B}^T \otimes \mathbf{A}) \text{vec}(\mathbf{X})$ , the optimal update of  $\mathbf{X}^{k+1}$  is given by

$$\text{vec}(\mathbf{X}^{k+1}) = (2\mathbf{T}_d \mathbf{T}_d^T + 2\alpha \tilde{\mathbf{L}} + \rho \mathbf{I}_{MN})^{-1} (\text{vec}(\rho \mathbf{P}^k - \mathbf{Q}^k) + \tilde{\mathbf{Y}}), \quad (29)$$

where the operator  $\text{vec}(\cdot)$  stacks the columns of an  $M \times N$  matrix into a vector of dimension  $MN$ , and parameters  $\tilde{\mathbf{L}}$  and  $\tilde{\mathbf{Y}}$  are represented by  $\mathbf{T}_d(\mathbf{I}_M \otimes \mathbf{L})\mathbf{T}_d^T$  and  $2\mathbf{T}_d \mathbf{T}_d^T \text{vec}(\mathbf{Y})$ , respectively, with  $\mathbf{T}_d$  expressed as

$$\mathbf{T}_d = \begin{bmatrix} \mathbf{I}_N & -R & & & \\ & \mathbf{I}_N & -R & & \\ & & \mathbf{I}_N & \ddots & \\ & & & \ddots & -R \\ & & & & \mathbf{I}_N \end{bmatrix}_{NM \times NM}. \quad (30)$$

---

**Algorithm 2 : Method for solving subproblem (28)**


---

**Input:**  $\mathbf{Y}, \mathbf{R}, \mathbf{B}, \mathbf{L}^{k+1}, \mathbf{P}^k, \mathbf{Q}^k, \alpha, \rho, K$ , error tolerance  $\delta$ .

- 1: Initialization:  $\mathbf{X}_0 = \mathbf{0}; \Delta\mathbf{X}_0 = -\nabla f_X(\mathbf{X}_0)$ ;
- 2: **repeat**
- 3:   1) Dynamic stepsize selection:
- 4:    $\mu = -\frac{\text{tr}\{(\Delta\mathbf{X}_m)^T \nabla f_X(\mathbf{X}_m)\}}{\text{tr}\{(\Delta\mathbf{X}_m)^T [\nabla f_X(\Delta\mathbf{X}_m) + \psi]\}},$   
with  $\psi = 2\mathcal{D}(\mathbf{Y}) - 2\mathbf{R}\mathcal{D}(\mathbf{Y})\mathbf{B}^T + \rho\mathbf{P}^k - \mathbf{Q}^k$ ;
- 5:   2) Conjugate direction update:
- 6:    $\mathbf{X}_{m+1} = \mathbf{X}_m + \mu\Delta\mathbf{X}_m$ ;
- 7:    $\Delta\mathbf{X}_{m+1} = -\nabla f_X(\mathbf{X}_{m+1}) + \theta\Delta\mathbf{X}_m$ ;
- 8:    $m = m + 1$ ;
- 9: **until**  $m$  reaches maximum number of iterations

**Output:** Recovered  $\mathbf{X}$ .

---

A detailed derivation of (29) is outlined in Appendix A. The solution in (29) requires the calculation of the inverse of an  $MN \times MN$  matrix. Thus, for a large number of vertices or time instants, this procedure is expected to be time-consuming. Instead, the conjugate gradient method [45] can be adopted to efficiently obtain a solution. Let  $f_X(\cdot)$  represent the objective function in (28). The algorithm mainly updates the stepsize and searching direction in each iteration. Denoting the search direction of the  $m$ th iteration as  $\Delta\mathbf{X}^m$ , the optimal stepsize  $\mu$  at the  $m$ th step can be obtained by an exact line search [46] given as  $\min_{\mu} f_X(\mathbf{X}^m + \mu\Delta\mathbf{X}^m)$ . By setting the derivative of  $f_X$  with respect to  $\mu$  equal to zero, we obtain

$$\text{tr}[(\Delta\mathbf{X}^m)^T \nabla f_X(\mathbf{X}^m + \mu\Delta\mathbf{X}^m)] = 0,$$

with the gradient of  $f_X$  calculated as

$$\begin{aligned} \nabla f_X = & 2\mathcal{D}(\mathbf{X} - \mathbf{Y}) - 2\mathbf{R}\mathcal{D}(\mathbf{X} - \mathbf{Y})\mathbf{B}^T + \rho(\mathbf{X} - \mathbf{P}^k) + \mathbf{Q}^k \\ & + 2\alpha(\mathbf{L}\mathcal{D}(\mathbf{X}) - \mathbf{R}\mathbf{L}\mathbf{X}\mathbf{B}^T + \mathbf{L}\mathbf{X}\mathbf{B}\mathbf{B}^T). \end{aligned} \quad (31)$$

Therefore, we can determine optimal stepsize  $\mu$  and update the searching direction by introducing the Fletcher-Reeves parameter given by  $\theta = \|\nabla f_X(\mathbf{X}^{m+1})\|_F^2 / \|\nabla f_X(\mathbf{X}^m)\|_F^2$ . The corresponding iterative optimization is detailed in *Algorithm 2*.

Similar to the subproblem of (25), by adding a constant term  $\frac{1}{2}\text{tr}\left(\frac{(\mathbf{Q}^k)^T \mathbf{Q}^k}{\rho^2}\right)$ , the subproblem of (26) is equivalent to the following optimization problem

$$\mathbf{P}^{k+1} = \arg \min_{\mathbf{P} \in \mathbb{R}^{N \times M}} \frac{1}{2} \left\| \mathbf{P} - \mathbf{X}^{k+1} - \frac{\mathbf{Q}^k}{\rho} \right\|_F^2 + \frac{\gamma}{\rho} \|\mathbf{P}\|_*, \quad (32)$$

which has closed-form solution

$$\mathbf{P}^{k+1} = \Gamma_{\gamma/\rho} \left( \mathbf{X}^{k+1} + \frac{\mathbf{Q}^k}{\rho} \right), \quad (33)$$

where  $\Gamma$  is singular value thresholding operator [47] that is the proximity operator associated with the nuclear norm. For each  $\tau \geq 0$ , the  $\Gamma$  is defined as follows:

$$\Gamma_{\tau}(\mathbf{X}) = \mathbf{U}\Theta_{\tau}(\mathbf{\Sigma})\mathbf{V}^T, \quad (34)$$

where  $\mathbf{U}, \mathbf{V}$  and  $\mathbf{\Sigma}$  are obtained from the singular value decomposition (SVD) of  $\mathbf{X}$ , that is,  $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ , with  $\sigma_i$  denoting the  $i$ th singular value and

$$\Theta_{\tau}(\sigma_i) = \text{sign}(\sigma_i) \max(|\sigma_i| - \tau, 0). \quad (35)$$

The operator (35) applies a soft-thresholding rule to the singular values of  $\mathbf{X}$ , effectively shrinking these towards zero.

The overall graph learning framework is presented in *Algorithm 1*. It should be noted that the optimization problem in (P1) is not jointly convex in  $\mathbf{L}$  and  $\mathbf{X}$ , the solution therefore corresponds to a local optimum rather than a global optimum. Besides, our empirical results suggest that after only eight iterations or less, the objective  $Q_1(\mathbf{L}, \mathbf{X})$  does not change more than the predefined threshold.

### C. Complexity analysis

We next provide a brief complexity analysis of the proposed GL-LRSS. For problem ( $\mathcal{S}_L$ ), the computation is dominated by the update of  $\mathbf{L}$  in (22), and the update is in turn dominated by  $\mathcal{D}(\mathbf{X})\mathcal{D}(\mathbf{X})^T$ , where the matrix-matrix product costs  $\mathcal{O}(N^2M + M^2N + N^3)$ . For problem ( $\mathcal{S}_X$ ), there are two main steps having the highest computation burden. For the first step of updating  $\mathbf{X}^k$ , we utilize the conjugate gradient method instead of the calculation in (29). In *Algorithm 2*, the computation is dominated by the gradient calculation according to (31), which is mainly determined by the matrix-matrix product, that is,  $\mathbf{R}\mathbf{L}\mathbf{X}\mathbf{B}^T$ , which costs  $\mathcal{O}(N^2M + M^2N + N^3)$  flops. When updating  $\mathbf{P}^k$  in the second step of (26), the computation of  $\Gamma$  dominates the computation consumption. The SVD of  $\mathbf{X}$  takes the computational cost of  $\mathcal{O}(\min(M^2N, N^2M))$  [48]. The last step of updating  $\mathbf{\Xi}$  and  $\mathbf{Q}$  involves the product of scalars and matrices with a cost of  $\mathcal{O}(MN)$ . Overall, the GL-LRSS is dominated by the updates of  $\mathbf{X}$  in (25) and  $\mathbf{L}$  in (22).

## V. EXPERIMENTS

We verified the effectiveness and performance of the proposed method on a variety of datasets: 1) two synthetic datasets under different graph structures, 2) meshes representing a dancing man [49], 3) a daily temperature dataset of China obtained from the National Oceanic and Atmospheric Administration [50], and 4) a daily evaporation dataset of California from the Department of Water Resources [51]. Moreover, we compare the proposed GL-LRSS with several state-of-the-art methods, including GMS [23], GL-Logdet [26], GL-Sigrep [14], SpecTemp [32], LGE [12], PCAG [21] and RPCAG [22]. GMS, GL-Logdet and SpecTemp are graph learning methods that only infer the graph structure from observations, whereas PCAG and RPCAG estimate low-rank components under a  $k$ -nearest-neighbor graph. In contrast, GL-LRSS, GL-Sigrep, and LGE simultaneously estimate the graph and low-rank components. For real-world data, we evaluated two types of  $R$  matrices for the proposed method. Specifically, we either set  $R$  to identity matrix  $\mathbf{I}$  or considered prior information of  $R$ .

We provided visual and quantitative results of the edges from the learned graph and the ground-truth graph. We conducted 20 independent Monte Carlo simulations to test the average performance of the proposed and baseline methods. To measure the estimation performance, we used the low-rank component estimation error (LCE):  $\|\hat{\mathbf{X}} - \mathbf{X}_0\|_F / \|\mathbf{X}_0\|_F$  and graph structure estimation error (GSE):  $\|\hat{\mathbf{L}} - \mathbf{L}_0\|_F / \|\mathbf{L}_0\|_F$ . Additionally, to measure the recovery performance of the edge position in the ground-truth graphs, we obtained the *Precision*, *Recall*, *F-measure* and *Normalized Mutual Information (NMI)*

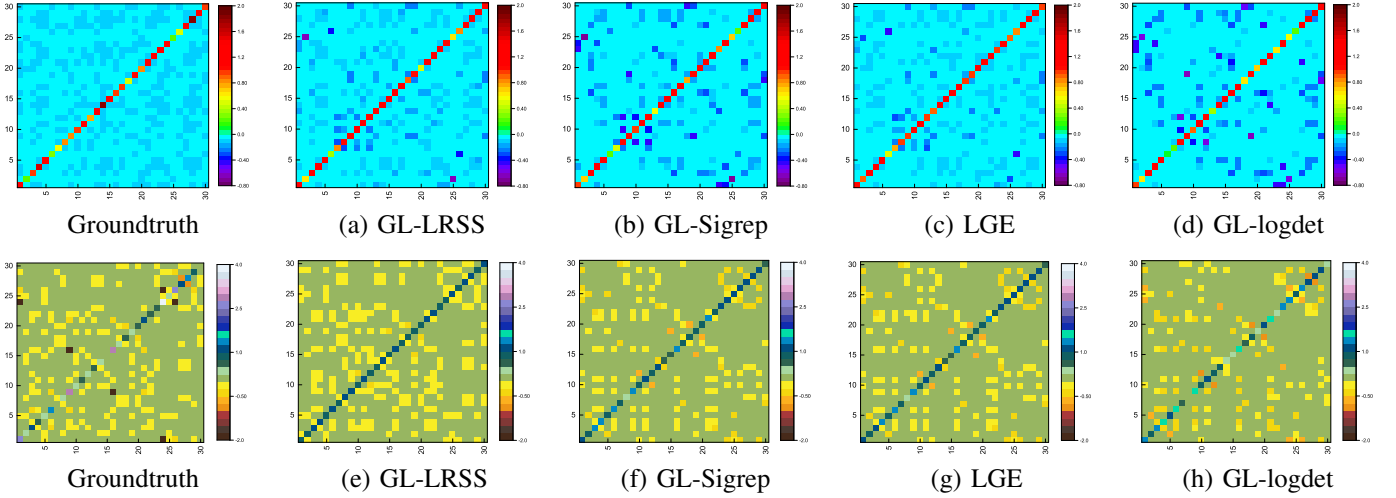


Fig. 1. Visual comparison between the learned graph Laplacian matrices and the ground-truth Laplacian. The columns from the left to the right are the ground-truth Laplacian, the Laplacians recovered by GL-LRSS, GL-Sigrep, LGE and GL-logdet. The rows from the top to the bottom are the learning results for the random geometric graph  $\mathcal{G}_{RGG}$  and grid graph  $\mathcal{G}_{grid}$ , respectively.

TABLE II  
GRAPH LEARNING PERFORMANCE FROM DIFFERENT TYPES OF TIME-VARYING GRAPH SIGNAL IN THE PROPOSED AND BASELINE METHODS.

	Random geometric graph $\mathcal{G}_{RGG}$						Grid graph $\mathcal{G}_{grid}$					
	GL-LRSS	GL-Sigrep	LGE	GL-logdet	PCAG	RPCAG	GL-LRSS	GL-Sigrep	LGE	GL-logdet	PCAG	RPCAG
F-measure	<b>0.8201</b>	0.7087	0.7196	0.6861	-	-	<b>0.7832</b>	0.6913	0.7029	0.6764	-	-
Precision	<b>0.8709</b>	0.7834	0.6469	0.8565	-	-	<b>0.7633</b>	0.6547	0.6593	0.7517	-	-
Recall	<b>0.7984</b>	0.6561	0.8212	0.5793	-	-	<b>0.8117</b>	0.7554	0.7575	0.6456	-	-
NMI	<b>0.5096</b>	0.2330	0.2761	0.2138	-	-	<b>0.4198</b>	0.3282	0.3339	0.3033	-	-
GSE	<b>0.3315</b>	0.3814	0.3445	0.5375	-	-	<b>0.7068</b>	0.7229	0.7234	0.9664	-	-
LCE	<b>0.0545</b>	0.2446	0.1424	-	0.4220	0.2432	<b>0.0665</b>	0.2465	0.1452	-	0.2223	0.1221

[52]. These four measures take values between 0 and 1, where values close to 1 indicate higher learning performance. For a fair comparison, we used a grid search to set the regularization parameters that maximize the performance for each method.

#### A. Experiments on synthetic data

We evaluated the GL-LRSS performance on synthetic datasets, which were created considering a 30-vertex undirected graph and two different graph connectivity models: grid graph  $\mathcal{G}_{grid}$  and random geometric graph  $\mathcal{G}_{RGG}$ . For the grid graph, each vertex with a random coordinate was connected to its five nearest neighbors, and the edge weight between two vertices was inversely proportional to their distance. For the random geometric graph, the vertex coordinates were uniformly random in a unit square, and each edge weight was determined from a Gaussian function,  $W(i, j) = \exp\left(-\frac{d(i, j)^2}{2\sigma^2}\right)$ , where  $\sigma = 0.5$ , considering threshold weights below 0.7. After graph construction, we computed the graph Laplacian matrix and normalized its trace to 30.

Given a ground-truth graph, we generated  $30 \times 100$  time-varying graph signals based on the proposed model in Eqs. (4) and (5). Unless otherwise stated, state transition  $R$  was set as the identity matrix. The case of a general diagonal  $R$  was also considered, as reported below. We selected the eigenvectors corresponding to the smallest  $r = 3$  eigenvalues as the basis

vectors, that is, the columns of  $\mathbf{U}$ . Zero-mean Gaussian noise having a standard deviation of 0.5 was set as the perturbation. Initial signal  $\mathbf{x}_1$  and weighted difference signal  $\mathbf{x}_t - R\mathbf{x}_{t-1}$  were smooth graph signals residing on the subspace corresponding to the three smallest eigenvalues of the graph Laplacian  $\mathbf{L}$ . Hence, the time-varying graph signals were approximately low-rank and presented spatiotemporal smoothness.

We applied GL-LRSS, GL-Sigrep, LGE, and GL-Logdet to learn the graph Laplacian matrices given only observation  $\mathbf{Y}$ . We used GL-LRSS, GL-Sigrep, LGE, PCAG, and RPCAG to estimate the low-rank components and obtained their average performance across 20 random instances of two graphs with the associated graph signals.

1) *Performance comparison:* A visual comparison of the evaluated methods is shown in Fig. 1, which depicts the ground-truth graph Laplacian and the Laplacian matrices learned by GL-LRSS, GL-Sigrep, LGE, and GL-Logdet from left to right. The first and second rows show the results under graph models  $\mathcal{G}_{RGG}$  and  $\mathcal{G}_{grid}$ , respectively. In both cases, the graph Laplacian provided by GL-LRSS is visually more consistent with the ground truth than the Laplacians obtained from the baseline methods. For further performance analysis, we obtained the quantitative results listed in Table II. Compared with the other four graph learning methods, the *F-measure* increases with the decreasing score of the *LCE* for



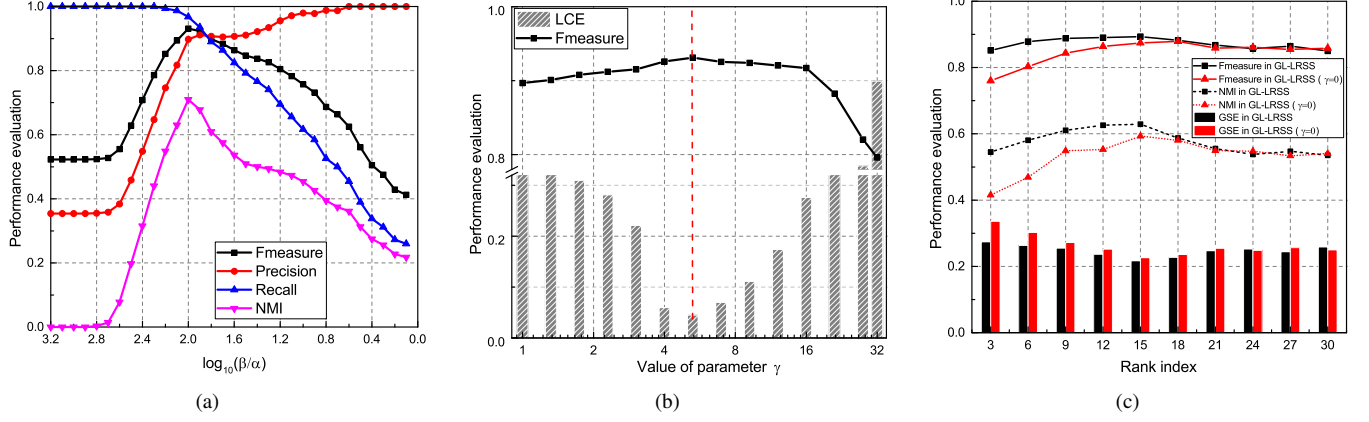


Fig. 2. For a random instance of  $\mathcal{G}_{RGG}$ , (a) performance of the GL-LRSS under different ratios of  $\beta$  to  $\alpha$ , with  $\gamma = 5.278$ , (b) performance of the GL-LRSS under different value of  $\gamma$ , where  $\alpha$  and  $\beta$  are chosen to maximize the *F-measure* for each  $\gamma$  and (c) the performance comparison of the proposed GL-LRSS and the GL-LRSS ( $\gamma = 0$ ) without nuclear norm under the different rank index.

the proposed GL-LRSS. Hence, better low-rank component estimation improves the graph estimation accuracy. For the five low-rank component estimation methods applied to  $\mathcal{G}_{grid}$ , the *LCE* decreases with the increasing *F-measure* scores. Specifically, the performance of PCAG and RPCAG on  $\mathcal{G}_{grid}$  is better than that on  $\mathcal{G}_{RGG}$  because the predefined graph is closer to the ground truth in  $\mathcal{G}_{grid}$ . These results suggest that a better graph inference improves low-rank component estimation. Thus, because the two estimation steps are alternately optimized, the proposed method outperforms GL-logdet, PCAG, and RPCAG.

The proposed GL-LRSS outperforms the baseline methods in graph inference and low-rank component estimation. For  $\mathcal{G}_{RGG}$ , GL-LRSS achieves the highest *F-measure* of 0.8201 and an *NMI* of 0.5096, as well as the lowest *GSE* of 0.3315 and an *LCE* of 0.0545. The improvement of GL-LRSS, compared with GL-Sigrep, is caused by the exploitation of long-term correlations, that is, the low-rank components. The improvement of GL-LRSS over LGE is due to the proper modeling of short-term correlations in Eq. (5), which demonstrate the benefits of applying spatiotemporal smoothness during graph learning. For  $\mathcal{G}_{grid}$ , the superior GL-LRSS performance is less obvious, possibly due to the low-rank assumption that limits the graph information encoded in the low-rank component, which varies depending on graph types.

2) *Effect of regularization parameters:* To better understand the behavior of the proposed GL-LRSS under different regularization parameter settings, we chose different powers of 2 ranging from 0 to 5 with variations of 0.4 to set  $\gamma$ , and different powers of 10 ranging from 0 to -2 with variations of 0.1 for  $\alpha$  and from 2 to 0 with variations of 0.1 for  $\beta$ . For the same  $\mathcal{G}_{RGG}$ , Fig. 2(a) shows the learning performance for fixed  $\gamma$  and varying ratios of  $\beta$  to  $\alpha$ . Because the learned graph approaches the ground truth, the *recall-precision* curve gradually interacts, leading to an *F-measure* peak. Thus, an appropriate ratio of  $\beta$  to  $\alpha$  can maximize the graph learning performance of the proposed GL-LRSS. A similar trend can be observed in the *NMI* curve.

To investigate the effect of parameter  $\gamma$ , we fixed  $\alpha$  and

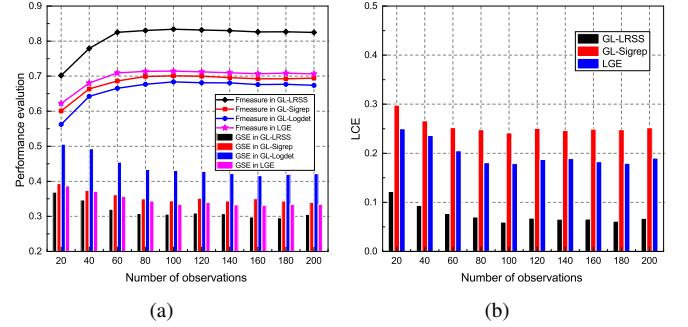


Fig. 3. (a) Graph learning performance of the baseline and proposed methods under different number of signals, and (b) low-rank component estimation performance of the baseline and proposed methods under different number of signals, for a random instance of  $\mathcal{G}_{RGG}$ .

$\beta$  to their best values (Fig. 2(a)) while varying  $\gamma$ . The GL-LRSS performance according to  $\gamma$  is shown in Fig. 2(b). The *F-measure* initially increases with increasing  $\gamma$ , possibly due to the action of the unclear norm in (P1) on the low-rank component estimation. After the *F-measure* reaches its peak of 0.93 and the *LCE* reaches its minimum, the performance decreases, because the influence of the unclear norm weakens. Hence, an appropriate value of  $\gamma$  improves the low-rank component estimation and the overall graph inference.

To verify the effectiveness of the nuclear norm  $\|\mathbf{X}\|_*$ , we generated graph signals for a random instance of  $\mathcal{G}_{RGG}$  under varying  $r$ . Then, we inferred the graph by solving (P1) with  $\gamma > 0$  and  $\gamma = 0$ . The GL-LRSS performance with and without ( $\gamma = 0$ ) the nuclear norm according to the rank index is shown in Fig. 2(c). For  $\gamma = 0$ , the GL-LRSS performance is not affected by the nuclear norm regularization term. For the *F-measure*, GL-LRSS with  $\|\mathbf{X}\|_*$  outperforms that without  $\|\mathbf{X}\|_*$  for the low rank index. The superior GL-LRSS performance with  $\|\mathbf{X}\|_*$  is less obvious as the rank index increases. This is possibly due to the introduction of the nuclear norm, which is more effective at a lower rank index, whereas its influence declines as the index approaches 30. Similar results were obtained for *NMI* and *GSE*. These

TABLE III  
THE GL-LRSS PERFORMANCE IN TWO CASES OF GENERAL DIAGONAL  $R$ .

	F-measure	Precision	Recall	NMI	GSE	LCE
$R_{\text{known}}$	0.7657	0.8203	0.7346	0.3817	0.3632	0.1390
$R_{\text{unknown}}$	0.6707	0.7873	0.5874	0.2833	0.6249	0.2870

results verify the correctness of the optimization problem (P1).

3) *Effect of number of observations*: For one random instance of a random geometric graph, we investigated the influence of the number of signals, from 20 to 200 in increments of 20. The graph learning performance is shown in Fig. 3(a) in terms of *F-measure* and *GSE*. We also report the performance of GSP-based methods as baselines for Laplacian recovery. The performance of all methods initially increases with the availability of an increasing number of signals for graph learning, until convergence after approximately 80 signals. Moreover, among all the evaluated methods, the proposed GL-LRSS attains the highest *F-measure* of approximately 0.82 and the lowest *GSE* of approximately 0.28, indicating its higher graph learning performance. The errors of the low-rank components recovered by GL-LRSS, GL-Sigrep, and LGE are shown in Fig. 3(b). The *LCE* trend is similar to that of the *F-measure*. Figs. 3(a) and 3(b) verify that GL-LRSS outperforms the other methods in terms of both graph learning and low-rank component estimation, possibly because our formulation utilizes long- and short-term correlations in spatiotemporal signals to facilitate learning.

4) *Effect of general diagonal matrix  $R$* : To examine the GL-LRSS performance when  $R$  is a general diagonal matrix, we generated  $R$  from a normal distribution  $\mathcal{N}(0.5, 0.25^2)$  and guaranteed that every entry in  $R$  was less than 1. We considered two cases of known and unknown  $R$ . For the unknown  $R$ , we assumed an incorrect  $R$  ( $R = \mathbf{I}_N$ ). For a random instance of  $\mathcal{G}_{\text{RGG}}$ , the results of the evaluated method for the two cases are listed in Table III. The GL-LRSS performance with unknown  $R$  is much worse than that for the known matrix, possibly due to the mismatching  $R$ . From Tables II and III for known  $R$ , the GL-LRSS performance for a general diagonal  $R$  is not as good as that for  $R$  being the identity matrix. Regarding the *LCE*, the advantage of GL-LRSS for  $R$  being the identity matrix is obvious, possibly because  $R = \mathbf{I}_N$  is the best case for low-rank component recovery during graph learning.

### B. Graph learning from dancer mesh dataset

We also evaluated the proposed GL-LRSS on real-world data. We first considered the dancer mesh dataset containing 143 frames that represent different dancing postures. At each frame, we considered the distance of 300 mesh vertices from each coordinate to the centroid as the observed signals, thus obtaining 143 time-varying graph signals with dimension of 300. During the whole sequence, the graph between the mesh vertices is unknown but is assumed to be fixed. GL-LRSS aimed to determine the intrinsic graph by capturing the body connectivity between the mesh vertices in terms of distances during the dancing sequence.

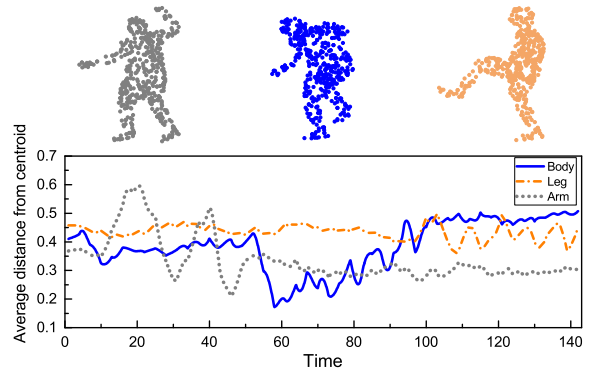


Fig. 4. Clustering of the dancer mesh: the plot (below) shows for each line the average distance between the points in different part of body and the centroid. We observe that each frame belongs to different phase of the dance, named "Arm", "Leg", "Body". The classification of the motion depends on the main fluctuation of the lines, that is, the part of body mainly involved in the dance.

TABLE IV  
COMPARISON OF THE MOTION CLASSIFICATION PERFORMANCE BETWEEN DIFFERENT METHODS IN DANCER MESH DATA.

	GL-LRSS	GL-Sigrep	LGE	PCAG	RPCAG	K-means on original data
RI	0.8385	0.7271	0.7835	0.7340	0.7455	0.6698
Purity	0.8671	0.7203	0.8015	0.7343	0.7343	0.5874
NMI	0.6422	0.5040	0.6095	0.5412	0.5651	0.4519

As depicted in Fig. 4, we obtained the ground-truth clusters of frames labeled by three dancing postures (i.e., moving arms, stretching legs, and bending body). For performance evaluation, we performed *k*-means clustering on the recovered low-rank component and compared the classification results. According to our experimental results on synthetic data, the effectiveness of the low-rank component estimation depends on the quality of the learned graph. Therefore, clustering performance on the low-rank component reflects the graph learning performance. We used *Purity*, *NMI*, and *RI* [53] to make a quantitative evaluation of clustering results.

We compared the clustering performance of the proposed GL-LRSS with two GSP-based methods (i.e., PCAG and RPCAG) both having a predefined five-nearest-neighbor graph. We also applied *k*-means clustering to the original data for reference. The classification results are listed in Table IV. The proposed GL-LRSS achieves the highest *RI* score of 0.8385, greater than the scores of 0.7271, 0.7835, 0.7340, and 0.7455 obtained by GL-Sigrep, LGE, PCAG, and RPCAG, respectively. Similar results were obtained in terms of *Purity* and *NMI*. As expected, the performance of *k*-means clustering on the original dataset is the worst, possibly due to its susceptibility to noise. These results demonstrate that the proposed GL-LRSS provides superior performance compared with the comparison methods on the dancer mesh dataset.

### C. Graph learning from temperature dataset

The daily average temperature data is collected from 60 observation sites in China [50] over 150 days starting from January 1, 2017 and have a size of  $60 \times 150$ . We aimed at learn-

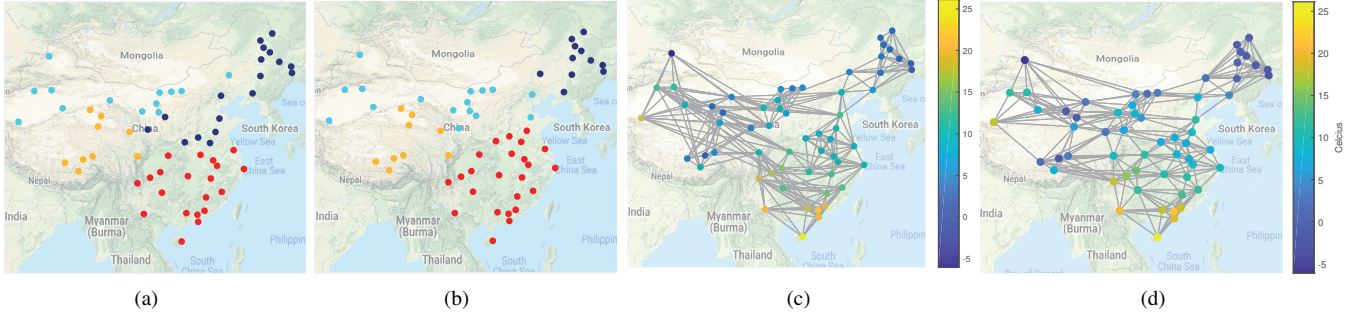


Fig. 5. (a) The locations of 60 measuring stations in China. Different colors represent the ground-truth 4 clusters that correspond to 4 geographical regions. (b) The clustering results utilizing the graph Laplacian obtained by the GL-LRSS( $R_I$ ). (c) Graph structure learned by the GL-LRSS( $R_I$ ), which achieves the best  $RI$  score in clustering performance. (d) Graph structure established by eight nearest neighbors according to the physical location of measuring stations. The color code in (c) and (d) represents the realistic temperature in Celcius scale on the 20th day.

TABLE V  
THE PERFORMANCE OF GRAPH LEARNING METHODS IN RECOVERING GROUND-TRUTH CLUSTERS OF TEMPERATURE MEASURING STATIONS.

	RI	Purity	NMI
KNN	0.7567	0.6667	0.4855
GMS	0.7667	0.5833	0.5037
GL-logdet	0.7411	0.6667	0.4701
SpecTemp	0.7832	0.5833	0.5201
GL-Sigrep	0.79	0.7167	0.5397
LGE	0.7833	0.75	0.5236
GL-LRSS ( $R_I$ )	0.8633	0.85	0.7203
GL-LRSS ( $R_{prior}$ )	0.8656	0.8333	0.7352

ing a graph structure for uncovering the inherent relationship between these observation sites in terms of daily temperature variations. In the experiment, we did not have an available ground-truth graph. Additionally, a  $k$ -nearest-neighbor graph was inappropriate. However, four climate zones of China (i.e., northern, southern, northwest, Qinghai-Tibet) could be regarded as a ground-truth clustering of the observation sites, which are differentiated by colors in Fig. 5(a). For performance evaluation, we compared the clustering results of the proposed and baseline methods by applying spectral clustering [54], which utilizes the learned graph Laplacian to divide the observation sites into four disjoint clusters. The clustering performance can reflect the quality of a graph.

Fig. 5(b) and 5(c) show the four-cluster partitions and graph topology obtained from the proposed GL-LRSS( $R_I$ ). The clusters are clearly distinguishable and close to the ground truth in Fig. 5(a). For comparison, we also show the natural choice of the graph constructed using eight nearest neighbors<sup>1</sup> in Fig. 5(d). The resulting graph seems inaccurate because it only considers the physical distance regardless of other influencing factors (e.g., altitude). Consequently, observation sites that are geometrically close may be geographically distant. Table V lists the best  $RI$ ,  $Purity$ , and  $NMI$  values achieved by the evaluated graph learning algorithms. Compared with the baseline methods, the proposed GL-LRSS achieves the highest scores for all three evaluation measures. Moreover,

by properly using the prior information of  $R^2$ , GL-LRSS ( $R_{prior}$ ) outperforms GL-LRSS( $R_I$ ). Hence, the proposed GL-LRSS outperforms the baseline methods for learning the graph topology on the temperature data from China.

#### D. Graph learning from evapotranspiration dataset

We now move onto the final real-world dataset, California daily evapotranspiration (ETo) dataset, published by the California Department of Water Resources [51]. It is collected from 62 active observation sites over 150 days starting from February 1, 2018, and it contains data with a size of  $62 \times 150$ . We aimed at inferring a graph that captures the similarity in the evapotranspiration evolution of different observation stations. In this experiment, because we did not have an obvious ground-truth graph topology, we used an ETo Zone Map [55] as reference, which divides the 62 observation sites into one of four zones. The ground-truth clusters are shown in Fig. 6(a). Similar to the previous example, we applied spectral clustering to the learned graphs and compared the resulting clusters with the ground truth.

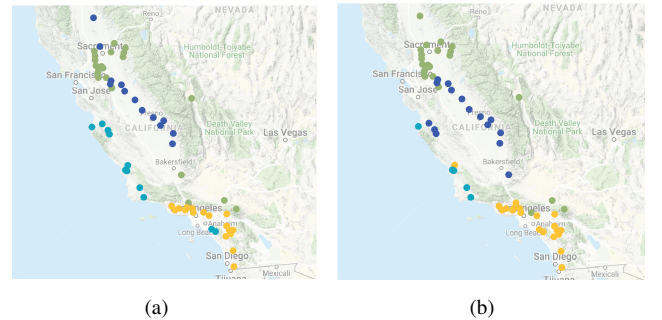


Fig. 6. (a) The ground-truth clusters of 62 observation sites in California. The colors from green, blue, cyan-blue to yellow represent ETo zone 14, zone 12, zone 6 and zone 9, respectively. (b) The resulting clusters obtained by the proposed GL-LRSS( $R_I$ ).

Fig. 6(b) shows the clustering results of the proposed GL-LRSS( $R_I$ ). The clusters obtained from the proposed GL-LRSS are visually similar to the ground-truth clusters. The corresponding quantitative results are listed in Table VI. Compared

<sup>1</sup>For the KNN baseline, we choose  $k = 8$  for both the temperature and ETo datasets, which leads to the best performance (i.e.,  $RI$  score).

<sup>2</sup>The parameter  $c_i$  in matrix  $R$  can be viewed as autocorrelation coefficient of data in the  $i$ th observation site. Here, we obtain  $c_i$  in advance through the autocorrelation function (ACF) (i.e., function [acf,lags]=autocorr(x)).

TABLE VI  
THE PERFORMANCE OF GRAPH LEARNING METHODS IN RECOVERING  
GROUND-TRUTH CLUSTERS OF ETO MEASURING STATIONS.

	RI	Purity	NMI
KNN	0.7644	0.6613	0.4805
GMS	0.7685	0.6774	0.5113
GL-logdet	0.7612	0.6290	0.4613
SpecTemp	0.7653	0.6451	0.4799
GL-Sigrep	0.8065	0.7419	0.5865
LGE	0.8153	0.7903	0.5945
GL-LRSS ( $R_I$ )	0.8496	0.8225	0.6544
GL-LRSS ( $R_{prior}$ )	0.8486	0.8064	0.6462

with the GSP-based methods (i.e., GL-Sigrep, GL-Logdet, SpecTemp, and LGE) and the other baseline methods, the proposed GL-LRSS( $R_I$ ) achieves the highest scores of 0.8496 for *RI*, 0.8255 for *Purity*, and 0.6544 for *NMI*. Nevertheless, the superior performance of GL-LRSS( $R_{prior}$ ) is not obvious, possibly because the correlation coefficients are inaccurate for the ETO data. Therefore, the proposed method exhibits higher performance than the comparison methods on the ETO dataset.

## VI. DISCUSSION

In this section, we first clarify the difference of the proposed method with methods for vector autoregressive models. Then, we discuss the applicability of the proposed method for a nondiagonal case of  $R$  in our model.

### A. Comparison with methods for vector autoregressive models

The model given by Eqs. (4) and (5) is analogous to a noisy version of a first-order vector autoregressive model. The differences between these models are twofold.

First, the driving noise in vector autoregression is assumed to be Gaussian white noise, whereas  $\mathbf{v}_t$  in (5) follows a GMRF encoding graph structure in the covariance matrix. Benefitting from the vector autoregressive model with driving noise having a covariance following the GMRF, the proposed model in (5) can thus characterize the spatial and temporal structures of the spatiotemporal signal through  $\mathbf{v}_t$  and  $R$ , respectively.

Second, the transition matrices in the vector autoregressive model can be viewed as weighted graphs that show Granger causal connections between nodes, but they are usually unknown. Thus, vector autoregression methods (e.g., those proposed in [36] and [43]) aim to recover multiple transition matrices from observations. In contrast, transition matrix  $R$  in (5) is known. As mentioned, because the graph structure is encoded in the covariance of  $\mathbf{v}_t$ , our graph learning problem can be regarded as inverse covariance matrix estimation.

### B. Applicability of our method for a nondiagonal $R$

In this work, we assume all measuring nodes to be independent for simplicity, that is, considering the dependencies of data in its own measuring node. Thus, we use a diagonal transition matrix  $R$  in the proposed model. In some cases, there exists dependencies between measuring nodes and hence the transition matrix is nondiagonal. However, under certain conditions of nondiagonal  $R$ , we can also derive the same

optimization problem in (15) by Proposition 1 below; thus, the proposed method can be also applied in those conditions.

**Proposition 1.** *If state transition matrix  $R$  in (5) is real and symmetric but not necessarily diagonal, the eigendecomposition of  $R$  is denoted as  $R = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$ , and the model given by (3) and (5) can be transformed into an equivalent model by multiplying  $\mathbf{Q}^T$  on both sides of the equation. The solution to (15) still represents the MAP estimate of the process variable in the equivalent model.*

*Proof:* As matrix  $R$  is real and symmetric, the eigendecomposition of  $R$  is denoted as  $R = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$ . We reformulate the model in (3) and (5) by multiplying matrix  $\mathbf{Q}^T$  as

$$\tilde{\mathbf{y}}_t = \tilde{\mathbf{x}}_t + \tilde{\mathbf{n}}_t, \quad (36)$$

$$\tilde{\mathbf{x}}_t = \mathbf{\Lambda}\tilde{\mathbf{x}}_{t-1} + \tilde{\mathbf{v}}_t, \quad (37)$$

where  $\tilde{\mathbf{y}}_t = \mathbf{Q}^T \mathbf{y}_t$ ,  $\tilde{\mathbf{x}}_t = \mathbf{Q}^T \mathbf{x}_t$ ,  $\tilde{\mathbf{n}}_t = \mathbf{Q}^T \mathbf{n}_t$  and  $\tilde{\mathbf{v}}_t = \mathbf{Q}^T \mathbf{v}_t$ . Based on the definition of  $\mathbf{n}_t$  and  $\mathbf{v}_t$ , we have that  $\tilde{\mathbf{n}}_t \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I}_N)$  and  $\tilde{\mathbf{v}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q})$ .

Note that model (5) with a non-diagonal transition matrix can be transformed into an equivalent model (37) with a diagonal transition matrix.

Given the weighted difference signal  $\tilde{\mathbf{d}}_t = \tilde{\mathbf{y}}_t - \mathbf{\Lambda}\tilde{\mathbf{y}}_{t-1} = \tilde{\mathbf{v}}_t + \tilde{\mathbf{n}}_t - \mathbf{\Lambda}\tilde{\mathbf{n}}_{t-1}$  and the multivariate Gaussian distribution on  $\tilde{\mathbf{v}}_t$ , we can compute the MAP estimate of  $\tilde{\mathbf{v}}_t$  as follows:

$$\begin{aligned} \tilde{\mathbf{v}}_{t,MAP}(\tilde{\mathbf{d}}_t) &:= \arg \max_{\tilde{\mathbf{v}}_t} p(\tilde{\mathbf{v}}_t | \tilde{\mathbf{d}}_t) = \arg \max_{\tilde{\mathbf{v}}_t} p(\tilde{\mathbf{d}}_t | \tilde{\mathbf{v}}_t) p(\tilde{\mathbf{v}}_t) \\ &= \arg \min_{\tilde{\mathbf{v}}_t} -\log p_E(\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t) - \log p_V(\tilde{\mathbf{v}}_t) \\ &= \arg \min_{\tilde{\mathbf{v}}_t} (\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t)^T \mathbf{W}^{-1} (\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t) + \alpha \tilde{\mathbf{v}}_t^T \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q} \tilde{\mathbf{v}}_t, \end{aligned}$$

where  $\mathbf{W} = \mathbf{I}_N + \mathbf{\Lambda}\mathbf{\Lambda}^T$ . By leveraging the inequality in (14), we obtain a relaxed optimization problem:

$$\min_{\tilde{\mathbf{v}}_t} \|\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t\|_2^2 + \alpha \tilde{\mathbf{v}}_t^T \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q} \tilde{\mathbf{v}}_t. \quad (38)$$

Specifically, the first and second terms in (38) can be rewritten as  $\mathbf{Q}^T (\mathbf{d}_t - \mathbf{v}_t)$  and  $\mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t$ , respectively. Using the inequality  $\|\mathbf{Q}\|_2^2 \|\mathbf{Q}^T (\mathbf{d}_t - \mathbf{v}_t)\|_2^2 \geq \|\mathbf{d}_t - \mathbf{v}_t\|_2^2$ , we further simplify the optimization problem (38) as

$$\min_{\mathbf{v}_t} \|\mathbf{d}_t - \mathbf{v}_t\|_2^2 + \alpha \mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t. \quad (39)$$

Therefore, the proof is complete.  $\blacksquare$

## VII. CONCLUSION

We study the problem of graph learning for spatiotemporal signals with long- and short-term correlations. By leveraging the spatiotemporal smoothness that reflects the temporal and graph structural information, as well as the low-rank property of the spatiotemporal signal, we formulate graph learning as a joint problem of estimating low-rank components and the graph topology. A new graph learning method, GL-LRSS, is proposed by applying alternating minimization and the ADMM to solve the formulated problem. These two optimization strategies improve each other, fostering better graph learning. Experiments on synthetic datasets verify a substantial performance improvement of the proposed GL-LRSS over state-of-the-art graph learning and low-rank component estimation methods. In addition, experiments on three real-world



datasets demonstrate that the proposed GL-LRSS outperforms the baseline methods in practice. In our future work, we plan to study a more general signal model with an arbitrary transition matrix and explore effective graph learning approaches.

## APPENDIX A

### DERIVATION OF THE CLOSED-FORM SOLUTION IN (29)

Being prepared for the following analysis, we first introduce the property of the vec-operator as follows:

$$\text{tr}(\mathbf{A}^T \mathbf{B}) = \text{vec}(\mathbf{A})^T \text{vec}(\mathbf{B}). \quad (40)$$

Then the second term in (28) can be transformed as

$$\begin{aligned} \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) &= \text{vec}(\mathbf{X} - \mathbf{R} \mathbf{X} \mathbf{B})^T \text{vec}[\mathbf{L}(\mathbf{X} - \mathbf{R} \mathbf{X} \mathbf{B})] \\ &= [\text{vec}(\mathbf{X})^T - \text{vec}(\mathbf{X})^T (\mathbf{B} \otimes \mathbf{R})] \cdot \\ &\quad [(\mathbf{I}_M \otimes \mathbf{L}) \text{vec}(\mathbf{X}) - (\mathbf{B}^T \otimes \mathbf{L} \mathbf{R}) \text{vec}(\mathbf{X})] \\ &= \text{vec}(\mathbf{X})^T \mathbf{T}_d (\mathbf{I}_M \otimes \mathbf{L}) \mathbf{T}_d^T \text{vec}(\mathbf{X}). \end{aligned}$$

Similarly, the first term in (28) is given by

$$\|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 = \text{tr}(\mathcal{D}(\mathbf{X} - \mathbf{Y})^T \mathcal{D}(\mathbf{X} - \mathbf{Y})) = \text{vec}(\mathbf{X} - \mathbf{Y})^T \mathbf{T}_d \mathbf{T}_d^T \text{vec}(\mathbf{X} - \mathbf{Y}),$$

and thus the objective function in problem (28) can be equivalently written as

$$\begin{aligned} \tilde{f}_X(\mathbf{v}) &= (\mathbf{v}^T - \text{vec}(\mathbf{Y})^T) \mathbf{T}_d \mathbf{T}_d^T (\mathbf{v} - \text{vec}(\mathbf{Y})) + \alpha \mathbf{v}^T \mathbf{G} \mathbf{v} \\ &\quad + \frac{\rho}{2} [\mathbf{v}^T - \text{vec}(\mathbf{P})^T + \text{vec}(\mathbf{Q})^T / \rho] [\mathbf{v} - \text{vec}(\mathbf{P}) + \text{vec}(\mathbf{Q}) / \rho], \end{aligned}$$

where  $\mathbf{G} = \mathbf{T}_d (\mathbf{I}_M \otimes \mathbf{L}) \mathbf{T}_d^T \in \mathbb{R}^{NM \times NM}$  and  $\mathbf{v} = \text{vec}(\mathbf{X})$ . The gradient of  $\tilde{f}_X(\mathbf{v})$  can be deduced as

$$\begin{aligned} \nabla \tilde{f}_X(\mathbf{v}) &= 2 \mathbf{T}_d \mathbf{T}_d^T \mathbf{v} - 2 \mathbf{T}_d \mathbf{T}_d^T \text{vec}(\mathbf{Y}) + 2 \alpha \mathbf{G} \mathbf{v} \\ &\quad + \text{vec}(\mathbf{Q}) + \rho \mathbf{v} - \rho \text{vec}(\mathbf{P}). \end{aligned} \quad (41)$$

The proof is accomplished by setting (41) to zero.

## REFERENCES

- [1] A. R. McIntosh, W. K. Chau, A. B. Protzner, "Spatiotemporal analysis of event-related fMRI data using partial least squares," *Neuroimage*, vol. 23, no. 2, pp. 764-775, 2004.
- [2] I. Kompatsiaris, and M. Strytzis, "Spatiotemporal Segmentation and Tracking of Objects for Visualization of Videoconference Image Sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 8, pp. 1388-1402, 2000.
- [3] H. Pham, C. Shahabi, Y. Liu, "EBM: an entropy-based model to infer social strength from spatiotemporal data," in *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pp. 265-276, 2013.
- [4] N. Eckert, E. Parent, R. Kies, H. Baya, "A spatio-temporal modelling framework for assessing the fluctuations of avalanche occurrence resulting from climate change: application to 60 years of data in the northern French Alps," *Climatic Change*, vol. 101, no. 3, pp. 515-553, 2010.
- [5] A. Sandryhalia, and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80-90, 2014.
- [6] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83-98, May. 2013.
- [7] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *Proc. 37th IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 3921-3924.
- [8] D. Romero, M. Ma, and G. B. Giannakis, "Kernel-based reconstruction of graph signals," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 2547-2560, May. 2017.
- [9] A. Sandryhalia, and J. M. F. Moura, "Discrete signal processing on graphs: Graph filters," in *Proc. 38th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2013, pp. 6163-6166.
- [10] J. Cheng, Q. Ye, H. Jiang, D. Wang, and C. Wang, "STCDG: An efficient data gathering algorithm based on matrix completion for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, pp. 850-861, 2013.
- [11] 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.
- [12] L. Rui, H. Nejati, S. H. Safavi, and N. M. Cheung, "Simultaneous low rank component and graph estimation for high-dimensional graph signals: Application to brain imaging," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 4134-4138.
- [13] Y. Liu, L. Yang, K. You, W. Guo and W. Wang, "Graph learning based on spatiotemporal smoothness for time-varying graph signal," *IEEE ACCESS*, vol. 7, pp. 62372 - 62386, 2019.
- [14] 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.
- [15] V. Kalofolias, "How to learn a graph from smooth signals," in *Proc. 19th Int. Conf. Artif. Intell. Statist.*, pp. 920-929, May. 2016.
- [16] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Fast robust PCA on graphs," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 740-756, Jun. 2016.
- [17] 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.
- [18] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Process. Mag.*, vol. 36, no. 3, pp. 16-43, May. 2019.
- [19] M. T. Bahadori, Q. R. Yu, and Y. Liu, "Fast multivariate spatio-temporal analysis via low rank tensor learning," in *Advances in neural information processing systems*, 2014, pp. 3491-3499.
- [20] 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, 2014.
- [21] B. Jiang, C. Ding, and J. Tang, "Graph-laplacian PCA: Closed-form solution and robustness," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2013, pp. 3492-3498.
- [22] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Robust principal component analysis on graphs," in *Proceedings of International Conference on Computer Vision*, Santiago, Chile, 2015, pp. 2812-2820.
- [23] A. Jung, "Learning the Conditional Independence Structure of Stationary Time Series: A Multitask Learning Approach," in *IEEE Transactions on Signal Processing*, vol. 63, no. 21, pp. 5677-5690, Nov. 1, 2015.
- [24] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432-441, 2008.
- [25] R. Mazumder and T. Hastie, "Exact covariance thresholding into connected components for large-scale graphical lasso," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 781-794, 2012.
- [26] B. M. Lake and J. B. Tenenbaum, "Discovering structure by learning sparse graph," in *Proc. 33rd Annual Cognitive Science conf.*, 2010, pp. 778-783.
- [27] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under structural and laplacian constraints," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 825-841, Sept. 2017.
- [28] S. P. Chepuri, S. Liu, G. Leus, and A. O. Hero, "Learning sparse graphs under smoothness prior," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 6508-6512.
- [29] M. G. Rabbat, "Inferring sparse graphs from smooth signals with theoretical guarantees," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 6533-6537.
- [30] V. Kalofolias, A. Loukas, D. Thanou, and P. Frossard, "Learning time varying graphs," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 2826-2830.
- [31] Yamada, Koki, Yuichi Tanaka, and Antonio Ortega. "Time-varying Graph Learning Based on Sparseness of Temporal Variation," in *Proc. 44th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2019, pp. 5411-5415.
- [32] S. Segarra, A. G. Marques, G. Mateos, and P. Vandergheynst, "Network topology inference from spectral templates," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 467-483, Sept. 2017.

- [33] B. Pasdeloup, V. Gripon, G. Mercier, D. Pastor, and M. G. Rabbat, "Characterization and inference of graph diffusion processes from observations of stationary signals," *IEEE Trans. Signal Inf. Process. Netw.*, vol. PP, no. 99, pp. 1-16, 2017.
- [34] R. Shafipour, S. Segarra, A. G. Marques, and G. Mateos, "Identifying undirected network structure via semidefinite relaxation," in *Proc. 43th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2018, pp. 4049-4053.
- [35] D. Thanou, X. Dong, D. Kressner, and P. Frossard, "Learning heat diffusion graphs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 484-499, Sept. 2017.
- [36] 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.
- [37] B. Baingana and G. B. Giannakis, "Tracking switched dynamic network topologies from information cascades," *IEEE Trans. Signal Process.*, vol. 65, no. 4, pp. 985-997, Feb. 2017.
- [38] Y. Shen, B. Baingana, and G. B. Giannakis, "Topology inference of directed graphs using nonlinear structural vector autoregressive models," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 6513-6517.
- [39] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis," *Journal of the ACM*, vol. 58, no. 3, pp. 1-37, May, 2011.
- [40] N. Cressie and C. K. Wikle, *Statistics for spatio-temporal data*. John Wiley & Sons, 2011.
- [41] S. Li, K. Li, and Y. Fu, "Temporal subspace clustering for human motion segmentation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 4453-4461, 2015.
- [42] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817-829, 2018.
- [43] 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, 2011.
- [44] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1-122, 2011.
- [45] J. R. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," Pittsburgh, PA, USA, Tech. Rep., 1994.
- [46] S. Boyd and L. Vandenberghe, *Convex optimization*. New York, NY, USA: Cambridge University Press, 2004.
- [47] J. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. on Optimization*, vol. 20, no. 4, pp. 1956-1982, 2010.
- [48] G. H. Golub and C. F. Van Loan, *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, USA, third edition, 1996.
- [49] Dynamic mesh for a dancing man, [Online]. Available: <https://epfl-lts2.github.io/rrp-html/start.html>.
- [50] National Centers for Environmental Information, 2018, [Online]. Available: <https://www.ncdc.noaa.gov/cdo-web/datasets>.
- [51] California Irrigation Management Information System, 2018, [Online]. Available: <https://cimis.water.ca.gov>.
- [52] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge University Press, 2008.
- [53] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *J. Amer. Statist. Assoc.*, vol. 66, no. 336, pp. 846-850, 1971.
- [54] U. Von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395-416, 2007.
- [55] Evapotranspiration Zones in California, 2018, [Online]. Available: <https://www.cimis.water.ca.gov/App-Themes/images/etozonemap.jpg>.

# Graph Learning for Spatiotemporal Signal with Long Short-Term Characterization

Yueliang Liu, *Student Member, IEEE*, Wenbin Guo, *Member, IEEE*, Kangyong You, Lei Zhao, Tao Peng, Wenbo Wang, *Senior Member, IEEE*

**Abstract**—Mining natural associations from high-dimensional spatiotemporal signals have received significant attention in various fields including biology, climatology and financial analysis, etcetera. Due to the widespread correlation in diverse applications, ideas that taking full advantage of correlated property to find meaningful insights of spatiotemporal signals have begun to emerge. In this paper, we study the problem of uncovering graphs that better reveal the relations behind data, with the help of long and short term correlated property in spatiotemporal signals. A spatiotemporal signal model considering both spatial and temporal relationship is firstly presented. Particularly, a low-rank representation together with a Gaussian Markov process is adopted to describe the signals' time-correlated behavior. Next, we cast the graph learning problem as a joint low-rank component estimation and graph Laplacian inference problem. A Low-Rank and Spatiotemporal Smoothness-based graph learning method (GL-LRSS) is proposed, which novelly introduces spatiotemporal smooth prior to the field of time-vertex signal analysis. Through jointly exploiting the low-rank property of long-time observations and the smoothness of short-time observations, the overall performance is effectively improved. Experiments on both synthetic and real-world datasets demonstrate the significant improvement on learning accuracy of the proposed GL-LRSS over current state-of-the-art low-rank estimation and graph learning methods.

**Index Terms**—Graph learning, spatiotemporal signal, graph signal, low rank, spatiotemporal smoothness.

## I. INTRODUCTION

IN a variety of modern applications, from finance and sociology to transportation and sensor network, many problems in signal processing, machine learning and statistics involve the analysis of spatiotemporal signals. Much of these signals take the form of long time series measured over a certain spatial range. Examples include biomedical imaging data [1], video sequences [2], social interactions among individuals [3], and environmental sensing [4]. Due to the complex spatial and temporal correlation, together with the space-time interactions, analyzing spatiotemporal signals is a challenging problem.

This work was supported by the National Natural Science Foundation of China (61271181, 61571054), the Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory Foundation and BUPT Excellent Ph.D. Students Foundation (Grant CX2018101) (*Corresponding author: Wenbin Guo.*)

Y. Liu and W. Guo are with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory, Shijiazhuang 050000, China (e-mail: {liuyueliang, gwb}@bupt.edu.cn).

K. You, L. Zhao, T. Peng and W. Wang are with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, China (email: {ykyyiwang, leizhao, wbwang}@bupt.edu.cn).

Graph is a useful tool for data analysis, as it shows a flexible description of data living on an irregular domain. In recent years, graph signal processing (GSP) [5] offers an engineering paradigm for processing spatiotemporal signals on graphs, referred to as time-varying graph signals, based on spectral graph theory [6]. For analyzing and learning purposes, it is often meaningful to represent the data through graph, and utilize graph Laplacian matrix which is equivalent to the graph topology to deal with numerous problems including graph signal compression [7], graph signal reconstruction [8], and graph filtering [9], [10], etcetera.

Nevertheless, though graph-based methods have been successful for many tasks, so far the graph structure is not always available and its natural choice (e.g., geographical K-nearest-neighbors) may not well capture the intrinsic relationship among data. The demands for graph learning that aims to spot trends or forecast future behavior from data analysis is raising. Therefore, how to extract the underlying relationship from observed spatiotemporal signals is important. In our previous work [11], we study the graph learning problem for time-varying graph signals where the temporal dynamic is particularly described through a proposed space-time signal model. As such, we successfully propose an efficient graph learning method by regularizing spatiotemporal smoothness of the graph signal. However, in many cases, the collected spatiotemporal data is approximately low-rank over a long-term horizon and has short-term stability. It is essential to consider these properties in signal representation, while many studies ignore the time-correlation of signals, for example by treating the successive signal independently or processing in the entire dimensional space [12], [13]. Even though the temporal relationship in our previous work is modeled as a first-order Markov process, it lacks the long-term characterization of spatiotemporal signals. Thus, this paper focuses on an enhanced graph learning method by making full use of long short-term correlated properties in spatiotemporal signals.

### A. Related works

Nowadays several works have summarized the approaches for the issue of low-rank component recovery [21] and graph learning [14], [15]. But these issues have not been jointly studied yet. For low-rank component recovery, many researches approximate spatiotemporal signals as low-rank matrices [16]–[18], which assumes the matrix collecting the time sequences to be approximately low-rank and achieves satisfying results. Recently, GSP-based approaches are proposed to recover approximated low-rank components by using spectral graph

regularization [19]–[21]. They incorporate graph smoothness on the low-rank matrices and improve both clustering and recovery performance. It is worth mentioning that, in all these works, the graph is predefined based on the geometric distance, which may not be accurate enough for further analysis.

For graph learning, the early studies learn the graphical model by per-node neighborhood selection [22]. To be stable under noise, the work in [23]–[25] propose graphical Lasso methods to estimate an inverse covariance or precision matrix. Nowadays, the fast-growing field of GSP provides a new way to solve graph learning problems. The basic idea of these methods is to identify Gaussian Markov Random Field (GMRF) models with precision matrix denoted by graph Laplacian matrix or its variants. By leveraging the smooth property of graph signals, smoothness-based methods have been adopted for graph inference. Dong *et al.* [12] firstly propose a valid combinatorial graph Laplacian (CGL) learning method under a smooth graph representation. Following this work, Kalofolias [13] reformulates the problem in terms of the adjacency matrix and proposes a computationally efficient algorithm. To generalize the restriction of the precision matrix being CGL, Egilmez *et al.* [26] identify a GMRF model whose precision matrix can be multiple types of graph Laplacian. Alternative smoothness-based approaches [11], [27], [28] also show effectiveness, with the methodological implementation provided in the former two based on temporal dynamics and edge selection, respectively. A theoretical analysis of reconstruction error is provided in [28]. The above methods learn a graph from smooth graph signals, while a few works make extra assumptions on graph dynamic for time-varying graph learning. For instance, the work in [29] learns a dynamic graph under the hypothesis that graph changes smoothly over time, and the method of Koki *et al.* [30] is based on the sparseness of graph variation.

There is another family of approaches to tackle the graph learning problem by incorporating physical insights on graph signal. In this case, the observations are modeled as the results of a physical process on the graph, for example, diffusion-based [31]–[34] and causality-based [35]–[37] methods. To be specific, Segarra *et al.* [31] and Padeloup *et al.* [32] identify a graph from stationary observations that are assumed to be generated by a diffusion process. To generalize the work, Shafipour *et al.* [33] explore the graph learning strategy that can be applied to non-stationary graph signals. Thanou *et al.* [34] propose a graph learning framework where the graph signals are the outcomes of heat diffusion processes. In addition, causality-based methods focus on estimating asymmetric adjacency matrix corresponding to a directed graph. In [35], Mei *et al.* consider a causal graph process to characterize the time series and apply it to temperature analysis. Under a structural equation model, authors in [36] propose a recursive least-square estimator to track both signal state and graph topology. Similarly, Shen *et al.* [37] describe the nonlinear dependency of signals via a structural vector autoregressive model and develop an efficient estimator to infer a sparse graph. Notice that graphs can be extracted from the aforementioned graph learning methods, but none of these works consider the global correlation of observations, i.e., the low-rank property.

The works of [12] and [38] are most related to our work. In signal representation, different from the model [12] for time-independent signals, we propose a general model for the spatiotemporal signal. Specifically, we fully exploit its long short-term correlation property to describe multiple types of time-correlated behaviors. Besides, benefit from such representation, the proposed model can be regarded as analogous to the Kalman filter, which is shown to deal with prediction tasks in our previous work [11]. Furthermore, in the optimization problem, the study in [38] and this work have the same purpose that jointly estimates the graph and low-rank component. By contrast to [38] that directly combines these two estimation problems for the final optimization, we derive the optimization problem based on the Bayesian statistical method and hence introduce a new regularization term to promote graph learning. As will be discussed, the superiority and utility of the proposed graph learning method is verified in our experiments.

## B. Contributions

In this paper, in order to learn a graph with high quality, a graph learning method is proposed, which takes low-rank property and local smoothness of spatiotemporal signals into consideration. Therein, leverage on the procedure of low-rank component estimation, the quality of the learned graph is well improved. In turn, the low-rank component is better estimated with the help of a refined graph. The main contributions of this paper are summarized as follows,

- 1) Taking advantage of both the long-term and short-term correlation properties, a graph-based model is proposed for the spatiotemporal signal. In particular, we integrate the low-rank representation in a global sense and temporal evolution in a local sense for signal description. Benefit from such description, spatiotemporal smoothness is introduced as a new prior to facilitate graph learning.
- 2) Under the signal model, the graph learning problem is formulated as a joint graph refinement and low-rank component estimation problem, which is then solved by the proposed low-rank and spatiotemporal smoothness-based graph learning method (GL-LRSS) as an application of ADMM and alternating minimization schemes. With special consideration of the different evolution patterns of signal among observation sites, the learning framework is well adapted to the real data analysis.
- 3) We perform numerous experiments on both synthetic and real-world datasets. Visual and quantitative comparison are provided in synthetic data. Besides, several classification tasks on real-world datasets demonstrate the superior performance of the proposed GL-LRSS over the state-of-the-art low-rank estimation and graph learning methods.

The remainder of this paper is organized as follows. In Section II, an overview of the notation and the preliminaries in graph signal processing are reviewed. In Section III, a low-rank graph-based model is proposed and the corresponding spatiotemporal smoothness prior is introduced. In Section IV, we formulate the graph learning problem as a joint low rank and graph topology estimation problem, and propose GL-LRSS to alternatively solve the optimization problem. The performance of GL-LRSS is presented and compared with



TABLE I  
LIST OF SYMBOLS AND THEIR MEANING

Symbols	Meaning
$\mathcal{G} \mid \mathbf{L} \mid \mathcal{L}^N$	weighted graph   graph Laplacian matrix   set of CGLs
$\mathcal{V} \mid \mathcal{E}$	vertex set   edge set
$N \mid M$	number of vertices   number of time instants
$\mathbf{I} \mid \mathbf{W} \mid \mathbf{D}$	identity matrix   adjacency matrix   degree matrix
$\mathbf{U} \mid \mathbf{\Lambda}$	eigenvector matrix   eigenvalue matrix of $\mathbf{L}$
$\mathbf{0} \mid \mathbf{1}$	column vector of zeros   column vector of ones
$\mathbf{X}^{-1} \mid \mathbf{X}^\dagger$	inverse of $\mathbf{X}$   pseudo-inverse of $\mathbf{X}$
$\mathbf{X}^T \mid \mathbf{x}^T$	transpose of $\mathbf{X}$   transpose of $\mathbf{x}$
$(\mathbf{X})_{ij}$	entry of $\mathbf{X}$ at $i$ -th row and $j$ -th column
$\mathbf{x}_i$	$i$ -th entry of $\mathbf{x}$
$\geq (\leq)$	element-wise greater (less) than or equal to operator
$\mathbf{X} \succeq \mathbf{0}$	$\mathbf{X}$ is a positive semidefinite matrix
$\text{tr} \mid \text{vec}$	trace operator   vectorization operator
$\otimes \mid \langle \cdot, \cdot \rangle$	Kronecker product operator   inner product operator
$\text{diag}(\mathbf{x})$	diagonal matrix formed by elements of $\mathbf{x}$
$p(\mathbf{x})$	probability density function of random vector $\mathbf{x}$
$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \Sigma)$	zero-mean multivariate Gaussian with covariance $\Sigma$
$\ \mathbf{X}\ _*$	nuclear norm of $\mathbf{X}$
$\ \mathbf{x}\ _1 \mid \ \mathbf{X}\ _1$	sum of absolute values of all elements ( $l_1$ -norm)
$\ \mathbf{x}\ _2^2 \mid \ \mathbf{X}\ _F^2$	sum of squared values of elements

1 baseline methods on both synthetic and real-world datasets  
2 in Section V. Section VI concludes the whole paper.

## 3 II. NOTATION AND PRELIMINARIES

### 4 A. Notations

5 For the convenience of reading, we firstly introduce some  
6 notations. Throughout the paper, the lowercase normal letters  
7 (e.g.,  $\alpha$  and  $\beta$ ), the lowercase boldface letters (e.g.,  $\mathbf{x}$  and  
8  $\mathbf{u}$ ) and the uppercase boldface letters (e.g.,  $\mathbf{X}$  and  $\mathbf{L}$ ) denote  
9 scalars, vectors, and matrices, respectively. Unless other stated,  
10 calligraphic capital letters (e.g.,  $\mathcal{E}$  and  $\mathcal{L}$ ) represent sets. **The**  
11 **rest of the notations are listed in Table I.**

### 12 B. Graph Laplacian

13 In this paper, we focus on an undirected, weighted graph  
14 with nonnegative edge weight and no self-loops. Let  $\mathcal{G} =$   
15  $(\mathcal{V}, \mathcal{E}, \mathbf{W})$  be an  $N$ -vertex weighted graph where  $\mathcal{V} =$   
16  $(v_1, \dots, v_N)$  is the vertex set and  $\mathcal{E}$  is the edge set of the graph.  
17 The adjacency matrix  $\mathbf{W}$  is an  $N \times N$  symmetric matrix such  
18 that  $(\mathbf{W})_{ij} = (\mathbf{W})_{ji}$ . The CGL of  $\mathcal{G}$  is defined as  $\mathbf{L} = \mathbf{D} - \mathbf{W}$ ,  
19 where the diagonal matrix  $\mathbf{D}$  denotes the degree matrix with  
20 its  $i$ th diagonal entry indicating the degree of vertex  $i$  (i.e.,  
21  $\text{diag}(\mathbf{D})_i = \sum_{j=1}^N W_{ij}$ ). In general, the set of CGL matrices  
22 can be written as

$$\mathcal{L}^N = \left\{ \mathbf{L} \mid \mathbf{L} \succeq \mathbf{0}, (\mathbf{L})_{ij} = (\mathbf{L})_{ji} \leq 0, i \neq j, \text{ and } \mathbf{L} \cdot \mathbf{1} = \mathbf{0} \right\}. \quad (1)$$

23 As shown in (1), the CGL is a real symmetric positive  
24 semidefinite matrix, so its eigenvalues are all non-negative.  
25 Provided that the eigendecomposition of CGL is  $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ ,  
26 where  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)$  and  $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N]$   
27 are the matrix of eigenvalue and eigenvector, respectively. The  
28 graph frequency spectrum is defined by the ascending array  
29 of eigenvalue  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$ , referred to as graph  
30 frequency, and the orthogonal eigenvectors  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N$  are  
31 the harmonics associated with graph frequencies. In addition,  
32 the CGL of a connected graph always has a zero value

of eigenvalue (i.e.,  $\lambda_1 = 0$  with multiplicity one) which  
corresponds to the eigenvector  $\mathbf{u}_1 = 1/\sqrt{N} \cdot \mathbf{1}$ .

### 35 C. Smooth Graph Signals

36 For a graph signal  $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$ , where  $x_i$  is  
37 attached to vertex  $v_i$ , its frequency component is defined by  
38 the graph Fourier transform (GFT), denoted as  $\hat{\mathbf{x}} = \mathbf{U}^T \mathbf{x}$ .  
39 Here, the frequency components corresponding to a higher  
40 eigenvalue indicate the larger variations between the signals of  
41 vertices, while the ones corresponding to small eigenvalue are  
42 relatively smooth. **Actually, many application datasets show**  
43 **that signals residing on a graph change smoothly between**  
44 **connected vertices.** Such smoothness property suggests how  
45 frequently a graph signal varies with respect to the underlying  
46 graph. To quantify the smoothness of signal  $\mathbf{x}$ , a typical metric  
47 can be written by graph Laplacian quadratic form [6] as

$$S(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{(i,j) \in \mathcal{E}} (\mathbf{W})_{ij} [x_j - x_i]^2, \quad (2)$$

48 where  $\mathcal{I} = \{(i, j) \mid (v_i, v_j) \in \mathcal{E}\}$  is the set of index pairs of  
49 vertices. As shown in Eq. (2), it measures the total variation  
50 of connected vertices associated with the edge set  $\mathcal{E}$ . **From**  
51 **the view of vertex domain, the smaller value of (2), the better**  
52 **smoothness of the signals on the graph structure.**

### 53 D. Correlated property of Spatiotemporal Signals

54 **Spatiotemporal signals can be viewed as time series attached**  
55 **to a graph of the observation sites. The common characteristics**  
56 **of them are massive redundancy and strong correlation. As**  
57 **pointed out in [40], global and local consistency principles**  
58 **have been identified for data description, which leads to**  
59 **two types of correlation properties. Next, we review these**  
60 **correlation properties in spatiotemporal signals as follows.**

61 **Long-term correlation:** According to the global consistency,  
62 there exist high correlations within spatiotemporal signals  
63 under a fixed spatial structure of observation sites [16], [18].  
64 Such correlation defined in a global sense can be interpreted  
65 as spatiotemporal signals lying in a low-dimensional subspace  
66 or being approximately low-rank [17], [21].

67 **Short-term correlation:** Following the local consistency  
68 principle, spatiotemporal signals are locally correlated [11],  
69 [40] as well. Concretely speaking, the observations of a certain  
70 site are correlated in neighboring time instants, and hence the  
71 temporal sequences vary smoothly over time. Similarly, at each  
72 time instant, observation sites nearby are observed spatially  
73 correlated with values being close to each other. These two  
74 types of short-term correlations are evaluated by temporal  
75 smoothness and spatial smoothness, respectively.

76 The past works in GSP are based on spatial and temporal  
77 smoothness. For instance, spatial smoothness is widely applied  
78 in GSP tasks including [8], [13] and [21], while quite a few  
79 works, such as [41] and [42], take advantage of temporal  
80 smoothness. Combining the two types of smoothness, our  
81 previous work [11] introduces spatiotemporal smoothness and  
82 proposes a graph learning method based on it. The spatiotem-  
83 poral smoothness is recalled in Assumption 1.

**Assumption 1** (Spatiotemporal smoothness). *The weighted time differences of spatiotemporal signals are smooth with respect to the graph structure.*

As shown above, the spatiotemporal smoothness characterizes the short-term property of time-varying graph signals.

### III. A GRAPH-BASED REPRESENTATION FOR SPATIOTEMPORAL SIGNALS

#### A. Signal representation

A spatiotemporal signal can be characterized by a matrix  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M] \in \mathbb{R}^{N \times M}$ , where  $N$  and  $M$  are the number of observation sites and the number of time instants, respectively. As illustrated in Section II-D, spatiotemporal signals are usually correlated in a global sense and local sense. Taking these correlation properties into consideration, the observed signal can be modeled as

$$\mathbf{y}_t = \mathbf{x}_t + \mathbf{n}_t, \quad (3)$$

$$\mathbf{x}_t = \mathbf{R}\mathbf{x}_{t-1} + \mathbf{v}_t, \quad (4)$$

where  $\mathbf{y}_t \in \mathbb{R}^N$  is the observation at time instant  $t$ ,  $\mathbf{x}_t \in \mathbb{R}^N$  forms the low-rank component, and  $\mathbf{n}_t \in \mathbb{R}^N$  denotes the perturbation part that is adopted as an isotropic noise model. We assume that the noise  $\mathbf{n}_t$  follows a multivariate Gaussian distribution with zero mean and covariance  $\sigma_n^2 \mathbf{I}_N$ .

To describe the short-term correlation, we impose a first-order Gaussian Markov process on variable  $\mathbf{x}_t$  in Eq. (4). The state transition matrix is expressed by a correlation matrix  $\mathbf{R} = \text{diag}(c_1, c_2, \dots, c_N)$ , where  $c_i$  is the local correlation coefficient of the  $i$ th observation site with value ranging from 0 to 1. Each  $c$  can be read as the autocorrelation coefficient that describes the correlation of data with a delayed copy (one-time lag in this model) of itself, and obtained from the training phase in advance. In particular,  $\mathbf{R}$  in our model is a general diagonal matrix, which considers the differences of data correlation in distinct observation sites. Based on our found in real data that the correlation coefficients in observation sites are concentrated around a value, we use a normal distribution  $\mathcal{N}(\mu, \sigma^2)$  to model the value of  $c_i, i = 1, \dots, N$ .

As for long-term correlation, we present an approximately low-rank representation for spatiotemporal signals. Such representation to some extent may compensate for the inaccuracy of signal description where we only consider the first-order autoregression in our model. By imposing a principle component analysis type (PCA-type) representation on the process variable  $\mathbf{v}_t$ , the signals are constrained to be low-rank, which enforces the commonalities along space and time dimensions.

**Remark 1.** *In some cases, the short-term correlation can lead to a long-term correlation. Under the normal distribution of coefficient  $c$ , the signal matrix  $\mathbf{X}$  will be low-rank when the variance  $\sigma^2 = 0$  (e.g.,  $\mathbf{R} = \mathbf{I}$ ). However, with the increasing value of  $\sigma^2$ , or in other words,  $\mathbf{R}$  is a more general diagonal matrix, the low-rank property of signal is less prominent.*

Recall that graph learning often relies on a signal representation that reflects the topology of the graph. In the following, we discuss the choice of process variable in our model.

#### B. Time-varying signals on graph

In this paper, we focus on the spatiotemporal signals which can be viewed as time-varying signals attached to a graph of observation sites. By using the graph Laplacian matrix, we model the temporal evolution of signal on a graph by the following definition of process variable  $\mathbf{v}_t$

$$\mathbf{v}_t = \mathbf{U}_{(r)} \mathbf{z}_t \text{ and } \mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \Lambda_{(r)}^\dagger). \quad (5)$$

The PCA-type representation at time instant  $t$  is a product  $\mathbf{v}_t = \mathbf{U}_{(r)} \mathbf{z}_t$ , where  $\mathbf{z}_t \in \mathbb{R}^{r \times 1}$  is the coefficient matrix and  $\mathbf{U}_{(r)} \in \mathbb{R}^{N \times r}$  is the basis vectors. Based on the fact that the eigenvector matrix of graph Laplacian can be interpreted as graph Fourier basis for graph signal representation, it provides a natural choice for basis vectors. Thus, the basis vectors are selected as the first  $r$  eigenvectors of the graph Laplacian. For coefficient matrix  $\mathbf{z}_t$ , it is assumed to follow  $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \Lambda_{(r)}^\dagger)$ , where  $\Lambda_{(r)}^\dagger$  is the Moore-Penrose pseudoinverse of eigenvalue matrix with the first  $r$  eigenvalues.

The motivation of the above definition is twofold. First, we seek a low-rank representation in terms of a small number of basis vectors where most of the variability of the data lies. The selected eigenvectors corresponding to  $r$  smallest eigenvalues can bring benefits to the smoothness property. Second, it can build an intuitive relationship between the graph structure and the graph signal. According to the definition (5), the assumption on  $\mathbf{z}_t$  together with the basis vector  $\mathbf{U}_{(r)}$  leads to a multivariate Gaussian distribution of  $\mathbf{v}_t$ , i.e.,  $\mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \tilde{\mathbf{L}}^\dagger)$ , with  $\tilde{\mathbf{L}}^\dagger = \mathbf{U}_{(r)} \Lambda_{(r)}^\dagger \mathbf{U}_{(r)}^T$ , such that the property or representation of time-varying signals can reflect the topology of the graph. To be noted, as  $\tilde{\mathbf{L}}^\dagger$  is the approximation of  $\mathbf{L}^\dagger$ , the signal only contains partial information of graph Laplacian, which actually influences the graph learning performance as will be discussed in the experiments.

Next, we show that the proposed model under a Gaussian prior on  $\mathbf{v}_t$  promotes the spatiotemporal smoothness in Assumption 1. Being prepared for the following analysis, we introduce the weighted difference operator of signal.

**Definition 1** (Weighted difference operator). *The weighted difference operator of graph signal  $\mathbf{X}$  as  $\mathcal{D}(\mathbf{X}) = \mathbf{X} - \mathbf{R}\mathbf{X}\mathbf{B}$ , where  $\mathbf{R}$  is the local correlation matrix and  $\mathbf{B}$  is the shift operator denoted as*

$$\mathbf{B} = \begin{bmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & & \ddots & \\ & & 0 & \ddots & 1 \\ & & & & 0 \end{bmatrix}_{M \times M}, \quad (6)$$

and the weighted difference signal equals to  $\mathcal{D}(\mathbf{X}) = [\mathbf{x}_1, \mathbf{x}_2 - \mathbf{R}\mathbf{x}_1, \mathbf{x}_3 - \mathbf{R}\mathbf{x}_2, \dots, \mathbf{x}_M - \mathbf{R}\mathbf{x}_{M-1}]$ .

Similar to the smoothness metric (2), the mathematical expression of the spatiotemporal smoothness is defined as

$$S(\mathcal{D}(\mathbf{X})) = \sum_{t=1}^M S(\mathbf{x}_t - \mathbf{R}\mathbf{x}_{t-1}) = \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})). \quad (7)$$

According to our model, the process variable  $\mathbf{v}_t$  reveals the core component of the graph signal. To intuitively show this,

we apply the weighted difference operator to the observed signal and the model becomes

$$\mathbf{d}_t = \mathbf{y}_t - \mathbf{R}\mathbf{y}_{t-1} = \mathbf{v}_t + \mathbf{n}_t - \mathbf{R}\mathbf{n}_{t-1}, \quad t = 1, 2, \dots, M \quad (8)$$

where we set  $\mathbf{d}_1 = \mathbf{y}_1$ . Based on the equation (8), the conditional probability of  $\mathbf{d}_t$  given  $\mathbf{v}_t$  can be written as

$$\mathbf{d}_t | \mathbf{v}_t \sim \mathcal{N}(\mathbf{v}_t, \sigma_n^2 (\mathbf{I}_N + \mathbf{R}\mathbf{R}^T)). \quad (9)$$

Then given the weighted difference signal  $\mathbf{d}_t$  and the Gaussian prior distribution of  $\mathbf{v}_t$ , we can compute a maximum a posteriori (MAP) estimate of the core component. Specifically, by applying Bayes' rule, the MAP estimate of  $\mathbf{v}_t$  is

$$\begin{aligned} \mathbf{v}_{tMAP}(\mathbf{d}_t) &:= \arg \max_{\mathbf{v}_t} p(\mathbf{v}_t | \mathbf{d}_t) = \arg \max_{\mathbf{v}_t} p(\mathbf{d}_t | \mathbf{v}_t) p(\mathbf{v}_t) \\ &= \arg \min_{\mathbf{v}_t} (-\log p_E(\mathbf{d}_t - \mathbf{v}_t) - \log p_V(\mathbf{v}_t)) \end{aligned} \quad (10)$$

$$\begin{aligned} &= \arg \min_{\mathbf{v}_t} (\mathbf{d}_t - \mathbf{v}_t)^T \mathbf{W}^{-1} (\mathbf{d}_t - \mathbf{v}_t) + \alpha \mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t \\ &\stackrel{relax}{:=} \arg \min_{\mathbf{v}_t} \|\mathbf{d}_t - \mathbf{v}_t\|_2^2 + \alpha \mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t, \end{aligned} \quad (11)$$

where  $\mathbf{W} = \mathbf{I}_N + \mathbf{R}\mathbf{R}^T$  and  $\alpha$  is a constant parameter proportional to the variance of noise  $\sigma_n^2$ . We note that the objective function (10) is hard to process when the correlation matrix  $\mathbf{R}$  is unknown. Taking advantages of the diagonal property of matrix  $\mathbf{R}$  and the following inequality

$$(\mathbf{d}_t - \mathbf{v}_t)^T \mathbf{W}^{-1} (\mathbf{d}_t - \mathbf{v}_t) \geq \lambda_{\min}(\mathbf{W}^{-1}) \|\mathbf{d}_t - \mathbf{v}_t\|_2^2, \quad (12)$$

the optimization problem (11) can be obtained through a relaxation procedure [11]. We can also derive the same problem for a nondiagonal matrix  $\mathbf{R}$  by Proposition 1.

**Proposition 1.** *If the state transition matrix  $\mathbf{R}$  in the model (4) is real, symmetric but non-diagonal, then*

1) *The analysis for a diagonal state transition matrix can be extended to a non-diagonal case, and*

2) *The optimization problems in the two cases are the same.*

*Proof:* See Appendix A. ■

It is worth mentioning that the regularization term  $\mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t$  derived in (11) is the same as  $S(\mathbf{x}_t - \mathbf{R}\mathbf{x}_{t-1})$  when the whole time instants  $M$  are considered. As a result, it shows that our proposed model favors the spatiotemporal smoothness, which can be applied to the field of time-vertex signal analysis.

#### IV. GRAPH LEARNING BASED ON LOW RANK AND SPATIOTEMPORAL SMOOTHNESS (GL-LRSS)

In many cases, the graph structure is typically unavailable, which makes the MAP estimate in (11) difficult to solve. Moreover, an accurate graph inference calls for a deep understanding on the property of spatiotemporal signals. Therefore, in the following, jointly exploiting the local smoothness and the global correlated property of spatiotemporal signals, we propose an efficient graph learning method. In Subsection A, we first formulate the graph learning problem. Then an optimization algorithm to the proposed problem, GL-LRSS, is presented in Subsection B based on ADMM and alternating minimization schemes. The complexity analysis of the proposed algorithm is provided as well.

##### A. Problem Formulation

As mentioned in Section II-D, time-varying graph signals smoothly evolve with respect to underlying graph topology, and meanwhile exhibit low-rank property based on the global consistency. Hence, given the observations of  $M$  time instants  $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_M] \in \mathbb{R}^{N \times M}$ , we focus on two objects of interest: (i) learn the graph Laplacian matrix  $\mathbf{L}$  that is equivalent to the graph structure. (ii) achieve better low-rank component estimation. Mathematically, by imposing additional constraints on graph Laplacian and low-rank component  $\mathbf{X}$ , we reformulate the graph learning problem (11) as a joint optimization problem with respect to  $\mathbf{L}$  and  $\mathbf{X}$ :

$$\begin{aligned} (\text{P1}) \quad & \min_{\mathbf{L}, \mathbf{X}} \quad Q_1(\mathbf{L}, \mathbf{X}) \\ \text{s.t.} \quad & Q_1(\mathbf{L}, \mathbf{X}) = \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) \\ & + \beta \|\mathbf{L}\|_F^2 + \gamma \|\mathbf{X}\|_*, \\ & \mathbf{L} \in \mathcal{L}^N, \quad \text{tr}(\mathbf{L}) = N, \end{aligned}$$

where  $\alpha$ ,  $\beta$  and  $\gamma$  are three positive regularization parameters corresponding to the regularization terms. In addition,  $\text{tr}(\cdot)$ ,  $\|\cdot\|_F$  and  $\|\cdot\|_*$  denote the trace, Frobenius norm and nuclear norm of a matrix, respectively. The first regularization  $\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}))$  induces the spatiotemporal smoothness encoded in Eq. (7). Together with the trace constraint that aims to avoid trivial solution, the second regularization  $\|\mathbf{L}\|_F^2$  controls the sparsity of the off-diagonal entries in  $\mathbf{L}$ , namely, the edge weights of the graph. To promote the long-term correlation property, we impose nuclear norm  $\|\mathbf{X}\|_*$  defined as the sum of the singular values of  $\mathbf{X}$ . It is the convex envelope of  $\text{rank}(\mathbf{X})$  that can be easy to solve. The last Laplacian constraint guarantees that the learned graph Laplacian is a valid CGL matrix satisfying the property in (1).

Notice that in problem (P1), the graph Laplacian and low-rank component interact with each other. It inspires us to extract a more accurate low-rank component for a better graph learning performance. Although the two regularization terms  $\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}))$  and  $\|\mathbf{X}\|_*$  promote the correlation property of spatiotemporal signals from a different perspective, they compensate each other to infer a meaningful graph. The detailed explanation is shown as follows

- Benefit from the signal representation, a new regularization term  $\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X}))$  is derived for graph learning. It encodes spatial and temporal relations of  $\mathbf{X}$  in graph Laplacian  $\mathbf{L}$  and weighted difference operator  $\mathcal{D}$  respectively, and meanwhile enforces the weighted difference signal to be smooth on graph. Let us highlight that, by contrast to the differential smoothness [18], this term contains a general correlation matrix  $\mathbf{R}$  that considers the different temporal evolution of data in distinct observation sites. As will be shown in real experiments, when a proper matrix  $\mathbf{R}$  is known a priori, the graph learning performance can be further improved.
- The use of  $\|\mathbf{X}\|_*$  induces a low-rank estimation, which enforces the commonalities within spatiotemporal signals. Although the short-term correlation in our model can lead to the low-rank property under specific conditions,

---

**Algorithm 1 : Graph learning based on low rank and spatiotemporal smoothness (GL-LRSS)**


---

**Input:** Observations  $\mathbf{Y}$ , local correlation  $\mathbf{R}$ , regularization parameters  $\alpha, \beta, \gamma$ , maximum iteration  $K$ , threshold  $\varepsilon$ .

- 1: Initialization:  $\mathbf{X}^0 = \mathbf{Y}$ ,  $k = 1$ ;
- 2: **repeat**
- 3:   1) Graph topology refinement:  
 $\mathbf{L}^{k+1} = G(\mathbf{X}^k, \mathbf{Y})$  by (16)
- 4:   2) Low-rank component estimation:  
 $\mathbf{X}^{k+1} = C(\mathbf{L}^{k+1}, \mathbf{Y})$  by (20)-(22)
- 5:   3)  $(\hat{\mathbf{L}}, \hat{\mathbf{X}}) = (\mathbf{L}^k, \mathbf{X}^k)$ ,  $k = k + 1$ ;
- 6: **until**  $k = K$  or  $|Q_1(\mathbf{L}^k, \mathbf{X}^k) - Q_1(\mathbf{L}^{k+1}, \mathbf{X}^{k+1})| < \varepsilon$

**Output:** Refined graph  $\hat{\mathbf{L}}$ , low-rank component  $\hat{\mathbf{X}}$ .

---

it may be inaccurate for signal description as only a first-order Markov model is considered. Such a drawback is compensated by the nuclear norm term that characterizes signal in a global sense, and the effectiveness of imposing such term is verified in Section V-A.

Having given the above analysis, we will give the solution to the problem. According to the fact that the better low-rank component estimation promotes the quality of the learned graph, while a good graph, in turn, facilitates an accurate low-rank component estimation. It motivates our alternating minimization framework, which iteratively refines the graph topology and estimates the low-rank component.

### B. Optimization algorithm

As the optimization problem (P1) is not jointly convex in  $\mathbf{L}$  and  $\mathbf{X}$ , GL-LRSS is therefore proposed to solve the above non-convex problem through an alternating optimization scheme. At each step, we optimize only one variable while holding all other variables constant. The iteration is shown as follows

1.  $G(\mathbf{X}, \mathbf{Y}) \triangleq \arg \min_{\mathbf{L}} Q_1(\mathbf{L}, \mathbf{X})$ , (S<sub>L</sub>)  
s.t.  $\mathbf{L} \in \mathcal{L}^N$ ,  $\text{tr}(\mathbf{L}) = N$ .
2.  $C(\mathbf{L}, \mathbf{Y}) \triangleq \arg \min_{\mathbf{X}} Q_1(\mathbf{L}, \mathbf{X})$ . (S<sub>X</sub>)

It is interesting to find that (S<sub>L</sub>) and (S<sub>X</sub>) are two subproblems with respect to the graph Laplacian  $\mathbf{L}$  and graph signal  $\mathbf{X}$ , respectively. By iteratively refining graph from low-rank representation and estimating the low-rank component with the help of the learned graph, we obtain the final solution of (P1) through alternating minimization. The detailed procedures are shown in the following derivation.

1) *Graph refinement in problem (S<sub>L</sub>):* Notice that (S<sub>L</sub>) is a strictly convex problem under convex constraints, since the Hessian matrix of the objective function  $2\beta\mathbf{I}_N$  is positive definite. To solve such a constrained convex problem, we use the alternating direction method of multipliers (ADMM) [43]. We reformulate the problem (P1) with respect to the graph Laplacian  $\mathbf{L}$  as

$$\begin{aligned} \min_{\mathbf{L}} \quad & \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) + \beta \|\mathbf{L}\|_F^2, \\ \text{s.t.} \quad & \mathbf{L} - \mathbf{Z} = 0, \\ & \mathbf{Z} \in \mathcal{L}^* \end{aligned} \quad (13)$$

where  $\mathbf{Z}$  is the auxiliary variable matrix and  $\mathcal{L}^*$  is denoted as  $\mathcal{L}^* = \{\mathbf{L} | \mathbf{L} \succeq 0, L_{ji} = L_{ij} \leq 0, i \neq j, \text{ and } \mathbf{L} \cdot \mathbf{1} = \mathbf{0}, \text{tr}(\mathbf{L}) = N\}$ . (14)

Therefore, the augmented Lagrangian of (13) is

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{L}, \mathbf{Z}, \mathbf{\Xi}) = & \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) + \beta \|\mathbf{L}\|_F^2, \\ & + \langle \mathbf{\Xi}, \mathbf{Z} - \mathbf{L} \rangle + \frac{\rho}{2} \|\mathbf{Z} - \mathbf{L}\|_F^2, \end{aligned} \quad (15)$$

where  $\mathbf{\Xi}$  is the Lagrange multiplier and  $\langle \cdot, \cdot \rangle$  is the inner product of matrices, while  $\rho > 0$  is the prescribed penalty parameter. We use the following formulas to update  $\mathbf{L}$ ,  $\mathbf{Z}$  and  $\mathbf{\Xi}$  to find a saddle point of (15)

$$\begin{aligned} \mathbf{L}^{k+1} &= \arg \min_{\mathbf{L}} \mathcal{L}_\rho(\mathbf{L}, \mathbf{Z}^k, \mathbf{\Xi}^k), \\ \mathbf{Z}^{k+1} &= \arg \min_{\mathbf{Z}} \mathcal{L}_\rho(\mathbf{L}^{k+1}, \mathbf{Z}, \mathbf{\Xi}^k), \\ \mathbf{\Xi}^{k+1} &= \mathbf{\Xi}^k + \rho(\mathbf{Z}^{k+1} - \mathbf{L}^{k+1}). \end{aligned} \quad (16)$$

By solving the linear equation where the gradient of each subproblem of (16) equals to zero, we have the following solutions

$$\mathbf{L}^{k+1} = \frac{\rho \mathbf{Z}^k + \mathbf{\Xi}^k - \alpha \mathcal{D}(\mathbf{X}) \mathcal{D}(\mathbf{X})^T}{2\beta + \rho}, \quad \mathbf{Z}^{k+1} = \prod_{\mathcal{L}^*} \left( \mathbf{L}^{k+1} - \frac{1}{\rho} \mathbf{\Xi}^k \right), \quad (17)$$

where  $\prod_{\mathcal{L}^*}$  is the Euclidean projection onto the set  $\mathcal{L}^*$ .

2) *Low-rank component Estimation in problem (S<sub>X</sub>):* As shown in (P1), the first two terms of  $\mathbf{X}$  are differentiable. But the nuclear norm term is undifferentiable, which is typically handled by the proximal operators. Due to the decomposability and converge property of ADMM, we also choose ADMM method to tackle the problem (S<sub>X</sub>). First of all, we provide an equivalent formulation of (P1) with respect to  $\mathbf{X}$

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{P}} \quad & \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) + \gamma \|\mathbf{P}\|_*, \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{P}, \end{aligned} \quad (18)$$

Notice that the objective function is split into two parts through introducing the linear equality constraint. Then the augmented Lagrangian of (18) is as follows

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{X}, \mathbf{P}, \mathbf{Q}) = & \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) \\ & + \gamma \|\mathbf{P}\|_* + \langle \mathbf{Q}, \mathbf{X} - \mathbf{P} \rangle + \frac{\rho}{2} \|\mathbf{X} - \mathbf{P}\|_F^2, \end{aligned} \quad (19)$$

where  $\mathbf{Q}$  is the Lagrange multiplier and  $\rho$  is a penalty parameter. Based on the augmented Lagrangian in (19), a final solution is obtained through the following iterative scheme

$$\mathbf{X}^{k+1} = \arg \min_{\mathbf{X}} \mathcal{L}_\rho(\mathbf{X}, \mathbf{P}^k, \mathbf{Q}^k), \quad (20)$$

$$\mathbf{P}^{k+1} = \arg \min_{\mathbf{P}} \mathcal{L}_\rho(\mathbf{X}^{k+1}, \mathbf{P}, \mathbf{Q}^k), \quad (21)$$

$$\mathbf{Q}^{k+1} = \mathbf{Q}^k + \rho(\mathbf{X}^{k+1} - \mathbf{P}^{k+1}). \quad (22)$$

According to (19), the subproblem (20) can be rewritten as

$$\begin{aligned} \mathbf{X}^{k+1} = & \arg \min_{\mathbf{X}} \|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 + \alpha \text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) \\ & + \frac{\rho}{2} \|\mathbf{X} - \mathbf{P}^k + \mathbf{Q}^k / \rho\|_F^2. \end{aligned} \quad (23)$$

As we can see, the subproblem (23) is a differentiable convex optimization problem that admits a closed-form solution. For the convenience of expression, we utilize the property of the vectorization operator, that is,  $\text{vec}(\mathbf{AXB}) =$



---

**Algorithm 2 : Method for solving subproblem (23)**


---

**Input:**  $\mathbf{Y}$ ,  $\mathbf{R}$ ,  $\mathbf{B}$ ,  $\mathbf{L}^{k+1}$ ,  $\mathbf{P}^k$ ,  $\mathbf{Q}^k$ ,  $\alpha$ ,  $\rho$ ,  $K$ , error tolerance  $\delta$ .

- 1: Initialization:  $\mathbf{X}_0 = \mathbf{0}$ ;  $\Delta\mathbf{X}_0 = -\nabla f_X(\mathbf{X}_0)$ ;
  - 2: **repeat**
  - 3:   1) Dynamic stepsize selection:
  - 4:    $\mu = -\frac{\text{tr}\{(\Delta\mathbf{X}_m)^T \nabla f_X(\mathbf{X}_m)\}}{\text{tr}\{(\Delta\mathbf{X}_m)^T [\nabla f_X(\Delta\mathbf{X}_m) + \psi]\}},$   
with  $\psi = 2\mathcal{D}(\mathbf{Y}) - 2\mathbf{R}\mathcal{D}(\mathbf{Y})\mathbf{B}^T + \rho\mathbf{P}^k - \mathbf{Q}^k$ ;
  - 5:   2) Conjugate direction update:
  - 6:    $\mathbf{X}_{m+1} = \mathbf{X}_m + \mu\Delta\mathbf{X}_m$ ;
  - 7:    $\Delta\mathbf{X}_{m+1} = -\nabla f_X(\mathbf{X}_{m+1}) + \theta\Delta\mathbf{X}_m$ ;
  - 8:    $m = m + 1$ ;
  - 9: **until** Stopping criterion satisfied
- Output:**
- Recovered
- $\mathbf{X}$
- .
- 

Similar to the subproblem (20), by adding a constant term  $\frac{1}{2}\text{tr}\left(\frac{(\mathbf{Q}^k)^T \mathbf{Q}^k}{\rho^2}\right)$ , the subproblem (21) is equivalent to the following optimization problem

$$\mathbf{P}^{k+1} = \arg \min_{\mathbf{P}} \frac{1}{2} \left\| \mathbf{P} - \mathbf{X}^{k+1} - \frac{\mathbf{Q}^k}{\rho} \right\|_F^2 + \frac{\gamma}{\rho} \|\mathbf{P}\|_*. \quad (27)$$

The above optimization has a closed-form solution

$$\mathbf{P}^{k+1} = \Gamma_{\gamma/\rho} \left( \mathbf{X}^{k+1} + \frac{\mathbf{Q}^k}{\rho} \right), \quad (28)$$

where  $\Gamma$  is the singular value thresholding operator [46] that is the proximity operator associated with the nuclear norm. For each  $\tau \geq 0$ , the  $\Gamma$  is defined as follows

$$\Gamma_\tau(\mathbf{X}) = \mathbf{U}\Theta_\tau(\boldsymbol{\Sigma})\mathbf{V}^T, \quad (29)$$

where  $\mathbf{U}$ ,  $\mathbf{V}$  and  $\boldsymbol{\Sigma}$  are obtained from the singular value decomposition (SVD) of  $\mathbf{X}$ , that is,  $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ , with  $\sigma_i$  denoting the  $i$ th singular value and

$$\Theta_\tau(\sigma_i) = \text{sign}(\sigma_i) \max(|\sigma_i| - \tau, 0). \quad (30)$$

The operator (30) applies a soft-thresholding rule to the singular values of  $\mathbf{X}$ , effectively shrinking these towards zero.

**The stopping criterion for solving subproblem ( $\mathcal{S}_L$ ) and ( $\mathcal{S}_X$ ) could be either a maximum number of iterations, or the change of target variable less than a threshold. By alternately minimizing the two subproblems, we can get the final solution within a few iterations. The detailed procedure for solving (P1) is summarized in Algorithm 1.**

**Complexity analysis:** In the following, we briefly discuss the computational complexity of our graph learning algorithm. For the problem ( $\mathcal{S}_L$ ), the computation is dominated by the update of  $\mathbf{L}$  in (17). The update procedure is dominated by  $\mathcal{D}(\mathbf{X})\mathcal{D}(\mathbf{X})^T$  where the matrix-matrix product costs  $\mathcal{O}(N^2M + M^2N + N^3)$  computational complexity. As for the problem ( $\mathcal{S}_X$ ), there are two main steps that are computation consuming. When it comes to the first step updating  $\mathbf{X}^k$ , we utilize the conjugate gradient method instead of the calculation of (24). As shown in Algorithm 2, the computation is dominated by the gradient calculation according to (26). The gradient procedure is mainly determined by the matrix-matrix product, i.e.,  $\mathbf{R}\mathbf{L}\mathbf{X}\mathbf{B}^T$ , which consumes  $\mathcal{O}(N^2M + M^2N + N^3)$  flops. When updating  $\mathbf{P}^k$  in the second step (21), the computation of  $\Gamma$  dominates the computation consumption. It takes  $\mathcal{O}(\min(M^2N, N^2M))$  for computing the SVD of matrix  $\mathbf{X}$  [47]. The last step of  $\Xi$  update and  $\mathbf{Q}$  update involves the product of scalar and matrix, and cost  $\mathcal{O}(MN)$ . Overall, we learn that the computation of proposed GL-LRSS is dominated by the  $\mathbf{X}$  update in (20) and the  $\mathbf{L}$  update in (17).

## V. EXPERIMENTS

The suitability of the proposed method for graph learning problem is illustrated on a wide variety of datasets: (a) two synthetic datasets under different graph structures, (b) dancer meshes representing a dancing man [48], (c) the daily temperature dataset of China from National Oceanic and Atmospheric Administration (NOAA) [49] and (d) the daily evaporation data of California from the California Department of Water Resources [50].

( $\mathbf{B}^T \otimes \mathbf{A}$ )  $\text{vec}(\mathbf{X})$ . Then, the optimal update of  $\mathbf{X}^{k+1}$  is denoted as

$$\text{vec}(\mathbf{X}^{k+1}) = \left( 2\mathbf{T}_d\mathbf{T}_d^T + 2\alpha\tilde{\mathbf{L}} + \rho\mathbf{I}_{MN} \right)^{-1} \left( \text{vec}(\rho\mathbf{P}^k - \mathbf{Q}^k) + \tilde{\mathbf{Y}} \right), \quad (24)$$

where  $\text{vec}(\cdot)$  is the vectorization operator that stacks the columns of a matrix into a vector, and the dimension of the transformed vector is  $MN \times 1$ . In addition, the parameters  $\tilde{\mathbf{L}}$  and  $\tilde{\mathbf{Y}}$  are respectively represented by  $\mathbf{T}_d(\mathbf{I}_M \otimes \mathbf{L})\mathbf{T}_d^T$  and  $2\mathbf{T}_d\mathbf{T}_d^T\text{vec}(\mathbf{Y})$ , with  $\otimes$  denoting the Kronecker product operator and  $\mathbf{T}_d$  denoting

$$\mathbf{T}_d = \begin{bmatrix} \mathbf{I}_N & -\mathbf{R} & & & \\ & \mathbf{I}_N & -\mathbf{R} & & \\ & & \mathbf{I}_N & \ddots & \\ & & & \ddots & -\mathbf{R} \\ & & & & \mathbf{I}_N \end{bmatrix}_{NM \times NM}. \quad (25)$$

The detailed derivation of (24) is described in Appendix B.

To be noted, the solution in (24) consists of calculating the inverse of an  $MN \times MN$  dimensional matrix. With the increasing number of vertices or time instants, this procedure can be expected to be time-consuming. The conjugate gradient method [44] can be used to deal with such a problem efficiently. For simplicity, we denote the objective function in (23) as  $f_X(\cdot)$ . In each iteration, it updates the stepsize and the searching direction. Since the  $f_X$  is differentiable, the optimal stepsize at the  $m$ th step can be decided by exact line search [45], i.e.,  $\min_{\mu} f_X(\mathbf{X}^m + \mu\Delta\mathbf{X}^m)$ , where  $\mu$  and  $\Delta\mathbf{X}^m$  denote the stepsize and the search direction at the  $m$ th iteration, respectively. Taking the derivative with respect to  $\mu$  and then setting to zero, we have

$$\text{tr} \left[ (\Delta\mathbf{X}^m)^T \nabla f_X(\mathbf{X}^m + \mu\Delta\mathbf{X}^m) \right] = 0,$$

with the gradient of  $f_X$  calculated as

$$\nabla f_X = 2\mathcal{D}(\mathbf{X} - \mathbf{Y}) - 2\mathbf{R}\mathcal{D}(\mathbf{X} - \mathbf{Y})\mathbf{B}^T + \rho(\mathbf{X} - \mathbf{P}^k) + \mathbf{Q}^k + 2\alpha(\mathbf{L}\mathcal{D}(\mathbf{X}) - \mathbf{R}\mathbf{L}\mathbf{X}\mathbf{B}^T + \mathbf{L}\mathbf{X}\mathbf{B}\mathbf{B}^T). \quad (26)$$

Therefore, we can determine the optimal stepsize  $\mu$  and update the searching direction by introducing the Fletcher-Reeves parameter given as  $\theta = \|\nabla f_X(\mathbf{X}^{m+1})\|_F^2 / \|\nabla f_X(\mathbf{X}^m)\|_F^2$ . The detailed procedure of iteration is described in Algorithm 2.

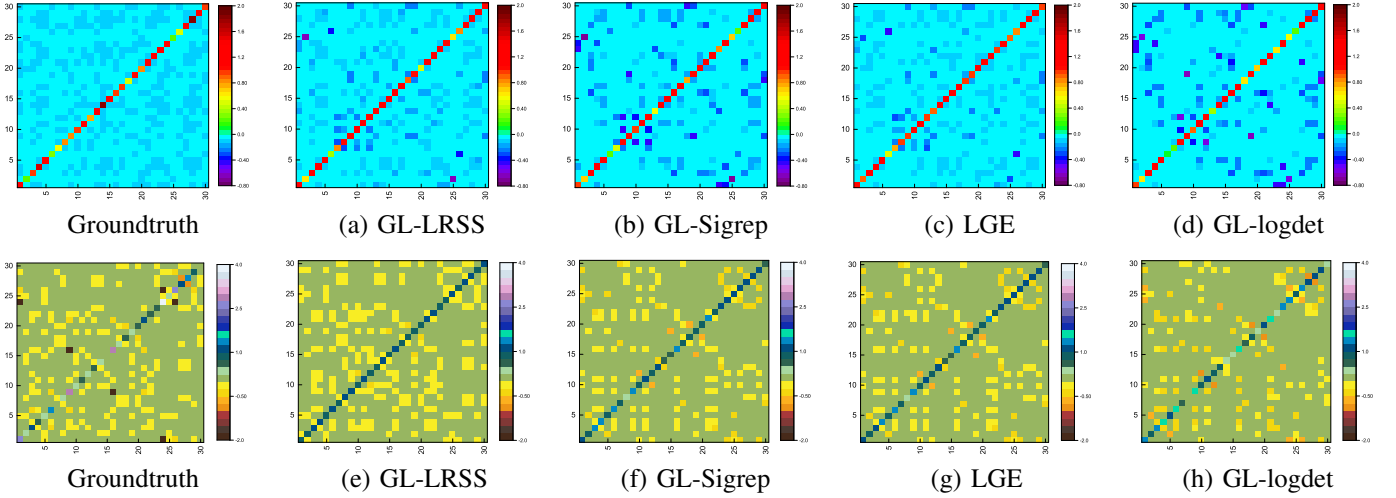


Fig. 1. Visual comparison between the learned graph Laplacian matrices and the groundtruth Laplacian. The columns from the left to the right are the groundtruth Laplacian, the Laplacians recovered by GL-LRSS, GL-Sigrep, LGE and GL-logdet. The rows from the top to the bottom are the learning results for the random geometric graph  $\mathcal{G}_{RGG}$  and grid graph  $\mathcal{G}_{grid}$ , respectively.

TABLE II  
GRAPH LEARNING PERFORMANCE FROM DIFFERENT TYPES OF TIME-VARYING GRAPH SIGNAL IN THE PROPOSED AND BASELINE METHODS.

	Random geometric graph $\mathcal{G}_{RGG}$						Grid graph $\mathcal{G}_{grid}$					
	GL-LRSS	GL-Sigrep	LGE	GL-logdet	PCAG	RPCAG	GL-LRSS	GL-Sigrep	LGE	GL-logdet	PCAG	RPCAG
F-measure	<b>0.8201</b>	0.7087	0.7196	0.6861	-	-	<b>0.7832</b>	0.6913	0.7029	0.6764	-	-
Precision	<b>0.8709</b>	0.7834	0.6469	0.8565	-	-	<b>0.7633</b>	0.6547	0.6593	0.7517	-	-
Recall	<b>0.7984</b>	0.6561	0.8212	0.5793	-	-	<b>0.8117</b>	0.7554	0.7575	0.6456	-	-
NMI	<b>0.5096</b>	0.2330	0.2761	0.2138	-	-	<b>0.4198</b>	0.3282	0.3339	0.3033	-	-
GSE	<b>0.3315</b>	0.3814	0.3445	0.5375	-	-	<b>0.7068</b>	0.7229	0.7234	0.9664	-	-
LCE	<b>0.0545</b>	0.2446	0.1424	-	0.4220	0.2432	<b>0.0665</b>	0.2465	0.1452	-	0.2223	0.1221

The proposed GL-LRSS is compared with several state-of-the-art methods, including GMS [22], GL-logdet [25], GL-Sigrep [12], SpecTemp [31], LGE [38], PCAG [19] and RPCAG [20]. Notice that GMS, GL-logdet and SpecTemp are graph learning methods that only infer the graph structure from observations, while PCAG and RPCAG are methods for only estimating low-rank components under a KNN graph. However, GL-LRSS, GL-Sigrep, and LGE simultaneously estimate the graph and low-rank component. For real-world data, we examine two types of  $\mathbf{R}$  in our method. One is GL-LRSS ( $\mathbf{R}_l$ ) with  $\mathbf{R} = \mathbf{I}$  and the other is GL-LRSS ( $\mathbf{R}_{prior}$ ) with a prior information of  $\mathbf{R}$ .

In our experiments, we provide both visual and quantitative comparison between the edges of the learned graph and the ones of the groundtruth graph. Particularly, we perform Monte-Carlo simulations to test the average performance of the proposed and baseline methods. To measure the estimation performance, we use low-rank component estimation error (LCE):  $\|\hat{\mathbf{X}} - \mathbf{X}_0\|_F / \|\mathbf{X}_0\|_F$  and graph structure estimation error (GSE):  $\|\hat{\mathbf{L}} - \mathbf{L}_0\|_F / \|\mathbf{L}_0\|_F$ . In addition, to evaluate the performance in terms of the recovery of the edge position in the groundtruth graph, we use four evaluation criteria, namely, *Precision*, *Recall*, *F-measure* and *Normalized Mutual Information (NMI)* [51]. The above four criteria take a value between 0 to 1, where the value more close to 1 implies the

better graph learning performance. Specifically, the *F-measure* is the overall criterion that takes both *Precision* and *Recall* into consideration, and it is defined as

$$F\text{-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (31)$$

where *Precision* measures the percentage of the correct edges in the learned graph and *Recall* evaluates the percentage of edges in the groundtruth graph that are presented in the learned graph. *NMI* is utilized to measure the mutual dependence between the learned edge set and the groundtruth one from an information-theoretic perspective. To make a fair comparison on graph learning methods, we select the regularization parameters through a grid search in each method, which maximizes the performance. Then we obtain the average performance over 20 independent Monte-Carlo experiments.

#### A. Experiments on synthetic data

In this subsection, we test the performance of the proposed method in synthetic datasets. We first create several synthetic datasets based on a 30-vertex undirected graph, chosen from two different graph connectivity models: the grid graph  $\mathcal{G}_{grid}$  and the random geometric graph  $\mathcal{G}_{RGG}$ . For a grid graph, each vertex with random coordinate is connected to its five nearest neighbors and the edge weight between two vertices is inversely proportional to their distance. As for

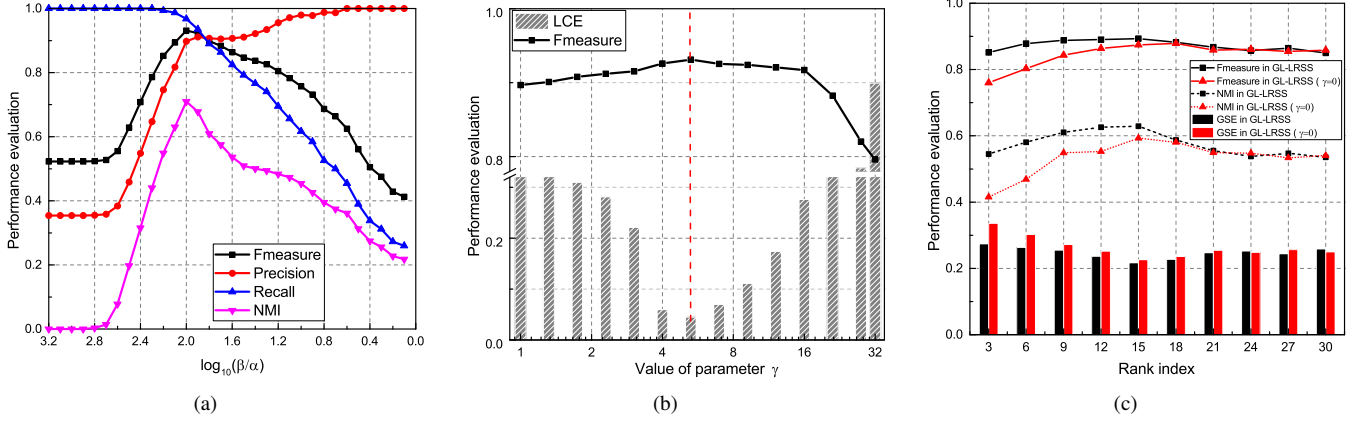


Fig. 2. For a random instances of  $\mathcal{G}_{RGG}$ , (a) performance of the GL-LRSS under different ratios of  $\beta$  to  $\alpha$ , with  $\gamma = 5.278$ , (b) performance of the GL-LRSS under different value of  $\gamma$ , where  $\alpha$  and  $\beta$  are chosen to maximize the  $F$ -measure for each  $\gamma$  and (c) the performance comparison of the proposed GL-LRSS and the GL-LRSS ( $\gamma = 0$ ) without nuclear norm under the different rank index.

the random geometric graph, we generate the coordinates of vertices uniformly at random in the unit square, determine the edge weights by a threshold Gaussian function  $W(i, j) = \exp\left(-\frac{d(i, j)^2}{2\sigma^2}\right)$ , where  $\sigma = 0.5$ , and threshold weights that are less than 0.7. After the graph construction, we compute the graph Laplacian matrix and normalize its trace to 30.

Given a specific groundtruth graph, we next generate  $30 \times 100$  time-varying graph signals  $\mathbf{Y}$  based on the proposed model in (3) and (4). Without loss of generality, the local correlation matrix  $\mathbf{R}$  is set as an identity matrix. We select eigenvectors corresponding smallest  $r = 3$  eigenvalues as the basis vectors, i.e., the columns of  $\mathbf{U}$ . As for the perturbation, the standard deviation of zero-mean Gaussian noise is set to 0.5. Notice that the initial signal  $\mathbf{x}_1$  and the weighted difference signal  $\mathbf{x}_t - \mathbf{R}\mathbf{x}_{t-1}$  are smooth graph signals residing on the subspace corresponding to the 3 smallest eigenvalues of graph Laplacian  $\mathbf{L}$ . Hence, the time-varying graph signal  $\mathbf{Y}$  is approximately low-rank and satisfies the spatiotemporal smoothness. We then apply GL-LRSS, GL-Sigrep, LGE, GL-Logdet to learn the graph Laplacian matrices given only the observation  $\mathbf{Y}$ . Meanwhile, GL-LRSS, GL-Sigrep, LGE, together with PCAG and RPCAG are utilized to estimate the low-rank component. Finally, we average the performance of the proposed and baseline methods over 20 random instances of two graphs with the associated graph signals.

1) *Performance comparison:* We first provide a visual comparison in Fig. (1), where from left to right denotes the groundtruth graph Laplacian, the Laplacian matrices learned by GL-LRSS, GL-Sigrep, LGE and GL-Logdet. The first and the second rows denote the results under the graph model  $\mathcal{G}_{RGG}$  and  $\mathcal{G}_{grid}$ , respectively. As we can see in both cases, the graph Laplacian learned by GL-LRSS is visually more consistent with the groundtruth one than the other baseline methods. For further analyzing the performance, we next show the quantitative comparison in Table II. First, on the one hand, compared with four graph learning methods, the  $F$ -measure increases with the decreasing score of  $LCE$ . It indicates that the better low-rank component estimation leads to a more accurate graph estimation. On the other hand, when

it comes to five low-rank estimation methods in  $\mathcal{G}_{grid}$ , the  $LCE$  decreases with the increasing score of  $F$ -measure. Specially, the performance of PCAG and RPCAG in  $\mathcal{G}_{grid}$  is better than that in  $\mathcal{G}_{RGG}$ , since the predefined graph is more close to the groundtruth one in  $\mathcal{G}_{grid}$ . These results suggest that a better graph inference improves the low-rank component estimation. Thus, as two estimation steps enhance each other, it is not surprising that the performance of GL-LRSS is better than that in GL-logdet, PCAG and RPCAG. Second, the proposed GL-LRSS shows superior performance compared to the others in both graph inference and low-rank component estimation. Especially, for  $\mathcal{G}_{RGG}$ , GL-LRSS achieves highest  $F$ -measure at 0.8201,  $NMI$  scores at 0.5096 and lowest  $GSE$  at 0.3315,  $LCE$  scores at 0.0545. The improvement of GL-LRSS compared to GL-Sigrep is due to the exploitation of long-term correlation, i.e., low rank. The improvement of GL-LRSS over LGE comes from the proper modeling of short-term correlation in (4), which verifies the benefits of applying spatiotemporal smoothness in graph learning procedure. When it comes to the graph  $\mathcal{G}_{grid}$ , the advantage of GL-LRSS is less obvious, possibly due to the low-rank assumption where graph information encoded in the low-rank component is limited and different under different graph types.

2) *Algorithm analysis:* To better understand the behavior of GL-LRSS under different sets of regularization parameters, we choose different powers of 2 ranging from 0 to 5, with a stepsize 0.4 for  $\gamma$ , and different powers of 10 ranging from 0 to -2, with a stepsize 0.1 for  $\alpha$  and 2 to 0, with a stepsize 0.1 for  $\beta$ . For the same  $\mathcal{G}_{RGG}$  as before, we firstly plot in Fig. 2(a) the learning performance given a selected  $\gamma$  under different ratios of  $\beta$  to  $\alpha$ . We see that in Fig. 2(a), as the learned graph approaches to the groundtruth one, the curve of  $Recall$  and  $Precision$  gradually interact, leading to a peak value of  $F$ -measure. This implies that an appropriate ratio of  $\beta$  to  $\alpha$  can maximize the graph learning performance of the proposed algorithm. A similar trend can be also observed in the curve of  $NMI$ . Secondly, to investigate the effect of the parameter  $\gamma$ , we choose the best combination of  $\alpha$  and  $\beta$  as illustrated in Fig. 2(a) for each value of  $\gamma$ .

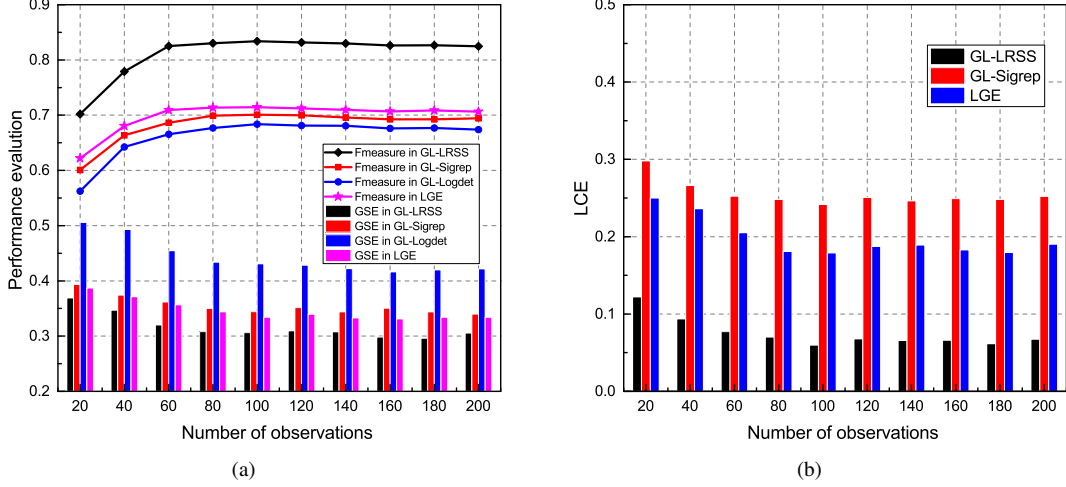


Fig. 3. (a) Graph learning performance of the baseline and proposed methods under different number of signals, and (b) low-rank component estimation performance of the baseline and proposed methods under different number of signals, for a random instances of  $\mathcal{G}_{RGG}$ .

The performance of GL-LRSS under different value of  $\gamma$  is depicted in Fig. 2(b). It is interesting to find that *F-measure* initially increases as the value of  $\gamma$  becomes larger. This can be attributed to the fact that the unclear norm in (P1) works in low-rank component estimation. After *F-measure* reaching its peak at 0.93 and meanwhile *LCE* reaching the minimum, the performance decreases as the influence of unclear norm is weakened. This implies that an appropriate  $\gamma$  enhances low-rank component estimation and thus results in a better graph inference. Next, to test the effectiveness of the term  $\|\mathbf{X}\|_*$ , we generate time-varying graph signals for a random instance of  $\mathcal{G}_{RGG}$  under the different values of  $r$ . Then we infer a graph by solving (P1) with  $\gamma > 0$  and  $\gamma = 0$ , respectively. The performance comparison of the proposed GL-LRSS and the GL-LRSS ( $\gamma = 0$ ) without nuclear norm under different rank index is shown in Fig. 2(c). In the case of  $\gamma = 0$ , the nuclear norm term does not work. As for metric *Fmeasure*, it can be observed that GL-LRSS with  $\|\mathbf{X}\|_*$  outperforms that without  $\|\mathbf{X}\|_*$  under low rank index and the advantage of GL-LRSS with  $\|\mathbf{X}\|_*$  is less obvious when the rank index increases. This possibly due to the introduction of the nuclear norm that efficiently works in the case of the lower rank index and its influence is declining as the rank index is close to 30. Similar results can be also obtained from the evaluation metric *NMI* and *GSE*. The above test verifies the correctness of the optimization model (P1).

Finally, for one random instance of random geometric graph, we investigate the influence of the number of signals varying from 20 to 200 in steps of 20. The performance of graph estimation is shown in Fig. 3(a), we plot the criteria of *F-measure* and *GSE* to evaluate the graph learning performance. We also present the performance of GSP-based methods to serve as a baseline for Laplacian recovery. As we can see, the performance of all methods initially increases as more signals are available to learn the graph, but remains stable when the number of signals is more than 80. Moreover, the proposed GL-LRSS attains highest *F-measure* around 0.82 and lowest edge recovery error *GSE* around 0.28, which shows

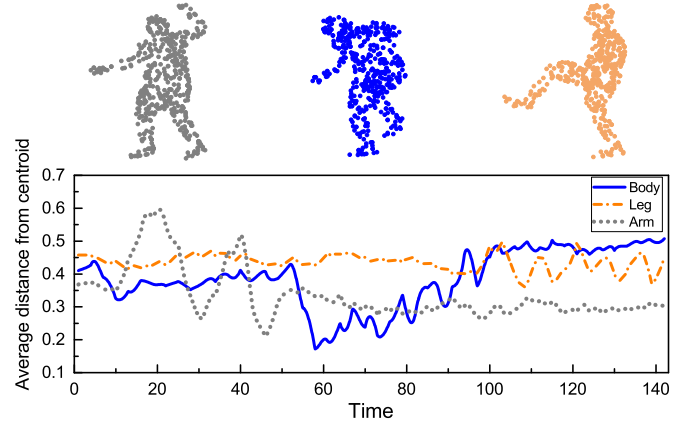


Fig. 4. Clustering of the dancer mesh: the plot (below) shows for each line the average distance between the points in different part of body and the centroid. We observe that each frame belongs to different phase of the dance, named "Arm", "Leg", "Body". The classification of the motion depends on the main fluctuation of the lines, that is, the part of body mainly involved in the dance.

better graph estimation. The error of low-rank components recovered by GL-LRSS, GL-Sigrep, and LGE are depicted in Fig. 3(b). The tendency of *LCE* is similar to that of the *F-measure* metric. Looking at Fig. 3(a) and 3(b) together, GL-LRSS outperforms the other methods in both graph and low-rank component estimation, possibly due to the fact that our formulation utilizes long and short term correlation of spatiotemporal signals to facilitate the learning performance.

#### B. Graph learning from dancer mesh dataset

We now test the proposed graph learning method on real-world data. We first consider the dancer mesh dataset describing a dance of man dancer. It collects 143 frames representing different phases of the dance. At each frame, we consider the distance of 300 mesh vertices from each coordinates to the centroid as our observed signals. This leads to 143 time-varying graph signals (i.e., one per frame), each of dimension



TABLE III

COMPARISON OF THE MOTION CLASSIFICATION PERFORMANCE BETWEEN DIFFERENT METHODS IN DANCER MESH DATA.

	GL-LRSS	GL-Sigrep	LGE	PCAG	RPCAG	K-means on original data
RI	0.8385	0.7271	0.7835	0.7340	0.7455	0.6698
Purity	0.8671	0.7203	0.8015	0.7343	0.7343	0.5874
NMI	0.6422	0.5040	0.6095	0.5412	0.5651	0.4519

TABLE IV

THE PERFORMANCE OF GRAPH LEARNING METHODS IN RECOVERING GROUNDTRUTH CLUSTERS OF TEMPERATURE MEASURING STATIONS.

	RI	Purity	NMI
KNN	0.7567	0.6667	0.4855
GMS	0.7667	0.5833	0.5037
GL-logdet	0.7411	0.6667	0.4701
SpecTemp	0.7832	0.5833	0.5201
GL-Sigrep	0.79	0.7167	0.5397
LGE	0.7833	0.75	0.5236
GL-LRSS ( $\mathbf{R}_I$ )	0.8633	0.85	0.7203
GL-LRSS ( $\mathbf{R}_{prior}$ )	0.8656	0.8333	0.7352

300. During the whole sequences, the graph between mesh vertices is unknown and assumed to be fixed. Our object is to uncover the intrinsic graph that captures the body connectivity between mesh vertices in terms of their distances in the dance.

As mentioned in Section V-A, low-rank component recovery and graph recovery benefit from each other, leading to a consistent optimal result. Even though the groundtruth graph of mesh data is unavailable, we can focus on low-rank recovery instead. As depicted in Fig. 4, according to the movement of different body parts, the frames can be labeled by three clusters indicating three phase of dance (i.e., moving arms, stretching legs and bending body). By performing k-means clustering on recovered low-rank component, the motion classification error can indirectly reflect the graph learning performance. The *Purity*, *NMI* and *RI* [52] scores are used to make a quantitative evaluation on the clustering results.

We compare the clustering performance of our GL-LRSS with GSP-based methods, PCAG and RPCAG both with a predefined five-nearest-neighbor graph. Besides, we apply k-means on original data as a baseline for clustering. The results of the dance classification are shown in Table III. As we can see, the proposed GL-LRSS achieves the highest score 0.8385 for *RI*, compared to 0.7271, 0.7835, 0.7340 and 0.7455 in GL-Sigrep, LGE, PCAG, and RPCAG, respectively. Similar results can be obtained in metrics *Purity* and *NMI*. As expected, the performance of k-means on the original data is the worst, possibly due to the fact that it is susceptible to the perturbation of noise. These results demonstrate that the proposed method provides competitive or superior performance than the comparison methods on this dancer mesh dataset.

### C. Graph learning from temperature dataset

The daily average temperature data is collected from 60 observation sites in China [49] over 150 days starting from January 1, 2017, and the size of data is  $60 \times 150$ . By applying our graph learning method, we would like to learn a graph structure to explore the inherent relationship between these observation sites in terms of the daily variations of temperature at their locations. In this example, we do not have an available groundtruth graph. Meanwhile, the natural choice of a graph based on the geometrical distance between observation sites does not seem appropriate, which will be shown in the following analysis. However, we have that the land of China can be divided into 4 zones (i.e., northern, southern, northwest and Qinghai-Tibet). This can be viewed as a groundtruth clustering of the 60 sites, which is shown by different colors in Fig. 5(a). For performance evaluation,

we apply spectral clustering [53] to the graphs learned by the proposed and baseline methods, and partition the vertex set into four disjoint clusters. We then compare these resulting clusters with the groundtruth information.

In Fig. 5(b) and 5(c), we visually show the four-cluster partition and the graph topology learned by GL-LRSS( $\mathbf{R}_I$ ). We can see that the four clusters are well distinguished, which is very close to the groundtruth one in Fig. 5(a). For comparison, we also show the natural choice of the graph constructed by 8 nearest neighbors<sup>1</sup> in Fig. 5(d). It is interesting to find that such a graph does not seem accurate enough as it only considers physical distance, regardless of other influence, e.g., altitude. The observation sites that are geometrically close may be geographically separated. It can be also verified by the results shown in Table IV where the best *RI*, *Purity* and *NMI* achieved by the graph learning algorithms are presented. Compared to the baseline methods, the GL-LRSS attains the highest score in terms of all three evaluation metrics. Besides, by properly using the prior information of  $\mathbf{R}^2$ , GL-LRSS ( $\mathbf{R}_{prior}$ ) shows better performance than GL-LRSS( $\mathbf{R}_I$ ). These results show that the proposed method outperforms the comparison methods in learning the graph topology on this temperature dataset.

### D. Graph learning from evapotranspiration dataset

We now move onto the final real-world dataset, California daily evapotranspiration (ETo) dataset, published by California Department of Water Resources [50]. It is collected from 62 active observation sites over 150 days starting from February 1, 2018, with the size of  $62 \times 150$ . By applying the proposed graph learning method, we would like to infer a graph that captures the similarity of evapotranspiration evolution between these stations. In this examples, we do not have an obvious groundtruth graph topology, however, an ETo Zone Map [54] provides another reference, which divides the 62 observation sites into one of the four zones. This leads to a groundtruth clusters shown in Fig. 6(a). Therefore, similar to the previous

<sup>1</sup>For the KNN baseline, we choose the number of neighbors  $k$  through a grid search, that leads to the best performance (i.e., *RI* score). The optimal value is  $k = 8$  for both the temperature and ETo datasets.

<sup>2</sup>The parameter  $c_i$  in matrix  $\mathbf{R}$  can be viewed as the autocorrelation coefficient of data in  $i$ th observation site. Here, we obtain the matrix  $\mathbf{R}$  in advance by using the autocorrelation function (ACF) (i.e., function `[acf,lags]=autocorr(x)`). For temperature dataset, the coefficients  $c$  at first five observation sites are shown as [0.9563, 0.9537, 0.9567, 0.9554, 0.9601]. For ETo dataset, the coefficients  $c$  at first five observation sites are shown as [0.7258, 0.7529, 0.7131, 0.6988, 0.7465].

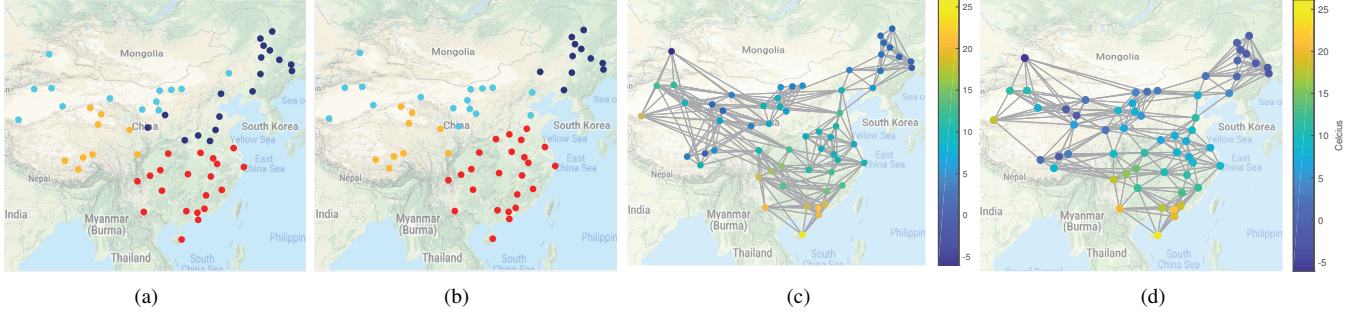


Fig. 5. (a) The locations of 60 measuring stations in China. Different colors represent the groundtruth 4 clusters that correspond to 4 geographical regions. (b) The clustering results utilizing learned graph Laplacian obtained by the GL-LRSS( $\mathbf{R}_I$ ). (c) Graph structure learned by the GL-LRSS( $\mathbf{R}_I$ ), which achieves the best  $RI$  score in clustering performance. (d) Graph structure established by 8 nearest neighbors according to the physical location of measuring stations. The color code in (c) and (d) represents the realistic temperature in Celcius scale on the 20th day.

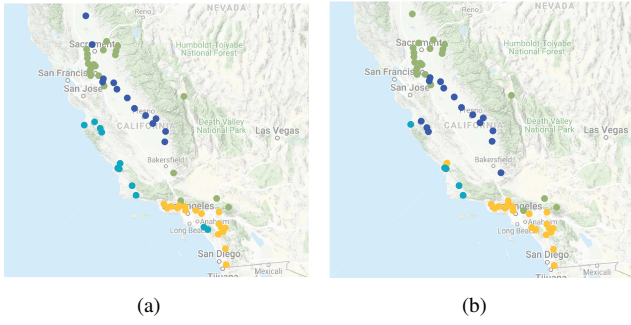


Fig. 6. (a) The groundtruth clusters of 62 observation sites in California. The colors from green, blue, cyan-blue to yellow represent ETo zone 14, zone 12, zone 6 and zone 9, respectively. (b) The resulting clusters obtained by proposed GL-LRSS( $\mathbf{R}_I$ ) method.

TABLE V  
THE PERFORMANCE OF GRAPH LEARNING METHODS IN RECOVERING GROUNDTRUTH CLUSTERS OF ETO MEASURING STATIONS.

	RI	Purity	NMI
KNN	0.7644	0.6613	0.4805
GMS	0.7685	0.6774	0.5113
GL-logdet	0.7612	0.6290	0.4613
SpecTemp	0.7653	0.6451	0.4799
GL-Sigrep	0.8065	0.7419	0.5865
LGE	0.8153	0.7903	0.5945
GL-LRSS ( $\mathbf{R}_I$ )	0.8496	0.8225	0.6544
GL-LRSS ( $\mathbf{R}_{prior}$ )	0.8486	0.8064	0.6462

schemes to solve the proposed problem. These two optimization steps facilitate from each other, leading to a better graph learning performance. Experiments on synthetic datasets verify a significant performance improvement over the state-of-the-art graph learning and low rank estimation methods. Also, experiments on three real-world datasets demonstrate that the proposed GL-LRSS outperforms these compared methods.

## APPENDIX A PROOF OF PROPOSITION 1

As the matrix  $\mathbf{R}$  is real and symmetric, the eigendecomposition of  $\mathbf{R}$  is denoted as  $\mathbf{R} = \mathbf{Q}\mathbf{A}\mathbf{Q}^T$ . Utilizing such property, we can reformulate the model in (3) and (4) by multiplying matrix  $\mathbf{Q}^T$  as

$$\tilde{\mathbf{y}}_t = \tilde{\mathbf{x}}_t + \tilde{\mathbf{n}}_t, \quad (32)$$

$$\tilde{\mathbf{x}}_t = \mathbf{\Lambda}\tilde{\mathbf{x}}_{t-1} + \tilde{\mathbf{v}}_t, \quad (33)$$

where  $\tilde{\mathbf{y}}_t = \mathbf{Q}^T \mathbf{y}_t$ ,  $\tilde{\mathbf{x}}_t = \mathbf{Q}^T \mathbf{x}_t$ ,  $\tilde{\mathbf{n}}_t = \mathbf{Q}^T \mathbf{n}_t$  and  $\tilde{\mathbf{v}}_t = \mathbf{Q}^T \mathbf{v}_t$ . Based on the definition of  $\mathbf{n}_t$  and  $\mathbf{v}_t$ , we have that  $\tilde{\mathbf{n}}_t \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I}_N)$  and  $\tilde{\mathbf{v}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q})$ .

As we can see, the model in (3) and (4) with non-diagonal matrix  $\mathbf{R}$  can be transformed into the above model with a diagonal state transition matrix  $\mathbf{\Lambda}$ .

Then given the weighted difference signal  $\tilde{\mathbf{d}}_t = \tilde{\mathbf{y}}_t - \mathbf{\Lambda}\tilde{\mathbf{y}}_{t-1} = \tilde{\mathbf{v}}_t + \tilde{\mathbf{n}}_t - \mathbf{\Lambda}\tilde{\mathbf{n}}_{t-1}$  and the multivariate Gaussian

examples, we apply the spectral clustering to the learned graph and compare the resulting clusters to the groundtruth clusters.

Fig. 6(b) shows the clustering results of the proposed GL-LRSS( $\mathbf{R}_I$ ). As depicted, the clusters obtained from the learned graph is visually very similar to the groundtruth clusters. Quantitative evaluation is further compared in Table V. Compared to the GSP-based methods (e.g., GL-Sigrep, GL-Logdet SpecTemp, LGE) and other baseline methods, the proposed GL-LRSS( $\mathbf{R}_I$ ) achieves the highest scores 0.8496 for  $RI$ , 0.8225 for  $Purity$  and 0.6544 for  $NMI$ . In addition, the advantage of GL-LRSS( $\mathbf{R}_{prior}$ ) is not obvious, possibly due to the fact that the correlation coefficients obtained are not accurate enough for the ETo data. These results show that the proposed method exhibits better performance than the comparison methods on this ETo dataset.

## VI. CONCLUSION

In this paper, we study the problem of learning graphs from spatiotemporal signals with long short-term correlation properties. By exploiting the low-rank property, as well as the spatiotemporal smoothness that accommodates both the time and graph structural information for graph learning procedure, we formulate the graph learning problem as a joint low-rank component and graph topology estimation problem. A correlation-aware graph learning method, GL-LRSS, is then proposed by applying alternating minimization and ADMM

1 distribution on  $\tilde{\mathbf{v}}_t$ , the MAP estimate of  $\tilde{\mathbf{v}}_t$  by applying Bayes'  
2 rule is expressed as

$$\begin{aligned}\tilde{\mathbf{v}}_{t,MAP}(\tilde{\mathbf{d}}_t) &:= \arg \max_{\tilde{\mathbf{v}}_t} p(\tilde{\mathbf{v}}_t | \tilde{\mathbf{d}}_t) = \arg \max_{\tilde{\mathbf{v}}_t} p(\tilde{\mathbf{d}}_t | \tilde{\mathbf{v}}_t) p(\tilde{\mathbf{v}}_t) \\ &= \arg \min_{\tilde{\mathbf{v}}_t} -\log p_E(\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t) - \log p_V(\tilde{\mathbf{v}}_t) \\ &= \arg \min_{\tilde{\mathbf{v}}_t} \left( -\log e^{-(\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t)^T \mathbf{W}^{-1} (\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t)} - \alpha \log e^{-\tilde{\mathbf{v}}_t^T \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q} \tilde{\mathbf{v}}_t} \right) \\ &= \arg \min_{\tilde{\mathbf{v}}_t} \left( \tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t \right)^T \mathbf{W}^{-1} (\tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t) + \alpha \tilde{\mathbf{v}}_t^T \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q} \tilde{\mathbf{v}}_t,\end{aligned}$$

3 where  $\mathbf{W} = \mathbf{I}_N + \Lambda \Lambda^T$ . Utilizing the inequality in (12), the  
4 above optimization problem can be relaxed as

$$\min_{\tilde{\mathbf{v}}_t} \left\| \tilde{\mathbf{d}}_t - \tilde{\mathbf{v}}_t \right\|_2^2 + \alpha \tilde{\mathbf{v}}_t^T \mathbf{Q}^T \tilde{\mathbf{L}} \mathbf{Q} \tilde{\mathbf{v}}_t. \quad (34)$$

5 According to our definition, the first term and second term can  
6 be rewritten as  $\mathbf{Q}^T (\mathbf{d}_t - \mathbf{v}_t)$  and  $\mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t$ , respectively. Ben-  
7 efit from the inequality  $\|\mathbf{Q}\|_2^2 \|\mathbf{Q}^T (\mathbf{d}_t - \mathbf{v}_t)\|_2^2 \geq \|\mathbf{d}_t - \mathbf{v}_t\|_2^2$ ,  
8 the optimization problem in (34) can be further simplified as

$$\min_{\mathbf{v}_t} \|\mathbf{d}_t - \mathbf{v}_t\|_2^2 + \alpha \mathbf{v}_t^T \tilde{\mathbf{L}} \mathbf{v}_t. \quad (35)$$

9 As shown above, the problem (35) is the same as (11).

## APPENDIX B

### DERIVATION OF THE CLOSED-FORM SOLUTION IN (24)

12 Being prepared for the following analysis, we first introduce  
13 the property of the vec-operator

$$\text{tr}(\mathbf{A}^T \mathbf{B}) = \text{vec}(\mathbf{A})^T \text{vec}(\mathbf{B}). \quad (36)$$

14 Then the second term in (23) can be transformed as

$$\begin{aligned}\text{tr}(\mathcal{D}(\mathbf{X})^T \mathbf{L} \mathcal{D}(\mathbf{X})) &= \text{vec}(\mathbf{X} - \mathbf{R} \mathbf{X} \mathbf{B})^T \text{vec}[\mathbf{L}(\mathbf{X} - \mathbf{R} \mathbf{X} \mathbf{B})] \\ &= [\text{vec}(\mathbf{X})^T - \text{vec}(\mathbf{X})^T (\mathbf{B} \otimes \mathbf{R})] \cdot \\ &\quad [(\mathbf{I}_M \otimes \mathbf{L}) \text{vec}(\mathbf{X}) - (\mathbf{B}^T \otimes \mathbf{L} \mathbf{R}) \text{vec}(\mathbf{X})] \\ &= \text{vec}(\mathbf{X})^T [(\mathbf{I}_M \otimes \mathbf{I}_N) - (\mathbf{B} \otimes \mathbf{R})] \cdot \\ &\quad [(\mathbf{I}_M \otimes \mathbf{L}) - (\mathbf{B}^T \otimes \mathbf{L} \mathbf{R})] \text{vec}(\mathbf{X}) \\ &= \text{vec}(\mathbf{X})^T \mathbf{T}_d (\mathbf{I}_M \otimes \mathbf{L}) [(\mathbf{I}_M \otimes \mathbf{I}_N) - (\mathbf{B}^T \otimes \mathbf{R})] \text{vec}(\mathbf{X}) \\ &= \text{vec}(\mathbf{X})^T \mathbf{T}_d (\mathbf{I}_M \otimes \mathbf{L}) \mathbf{T}_d^T \text{vec}(\mathbf{X}).\end{aligned}$$

15 Similarly, the first term in (23) can be denoted as

$$\begin{aligned}\|\mathcal{D}(\mathbf{X} - \mathbf{Y})\|_F^2 &= \text{tr}(\mathcal{D}(\mathbf{X} - \mathbf{Y})^T \mathcal{D}(\mathbf{X} - \mathbf{Y})) \\ &= \text{vec}(\mathbf{X} - \mathbf{Y})^T \mathbf{T}_d \mathbf{T}_d^T \text{vec}(\mathbf{X} - \mathbf{Y}),\end{aligned}$$

16 and the objective function of problem (23) can be equivalently  
17 written as

$$\begin{aligned}\tilde{f}_X(\mathbf{v}) &= (\mathbf{v}^T - \text{vec}(\mathbf{Y})^T) \mathbf{T}_d \mathbf{T}_d^T (\mathbf{v} - \text{vec}(\mathbf{Y})) + \alpha \mathbf{v}^T \mathbf{G} \mathbf{v} \\ &\quad + \frac{\rho}{2} [\mathbf{v}^T - \text{vec}(\mathbf{P})^T + \text{vec}(\mathbf{Q})^T / \rho] [\mathbf{v} - \text{vec}(\mathbf{P}) + \text{vec}(\mathbf{Q}) / \rho],\end{aligned}$$

18 where  $\mathbf{G} = \mathbf{T}_d (\mathbf{I}_M \otimes \mathbf{L}) \mathbf{T}_d^T \in \mathbb{R}^{NM \times NM}$ , and  $\mathbf{v} = \text{vec}(\mathbf{X})$ .  
19 The gradient of  $\tilde{f}_X(\mathbf{v})$  can be deduced as

$$\begin{aligned}\nabla \tilde{f}_X(\mathbf{v}) &= 2 \mathbf{T}_d \mathbf{T}_d^T \mathbf{v} - 2 \mathbf{T}_d \mathbf{T}_d^T \text{vec}(\mathbf{Y}) + 2 \alpha \mathbf{G} \mathbf{v} \\ &\quad + \text{vec}(\mathbf{Q}) + \rho \mathbf{v} - \rho \text{vec}(\mathbf{P}).\end{aligned} \quad (37)$$

20 By setting  $\nabla \tilde{f}_X(\mathbf{v})$  to zero, the unique optimal solution  
21  $\text{vec}(\mathbf{X})$  can be obtained as (24).

## REFERENCES

- [1] A. R. McIntosh, W. K. Chau, A. B. Protzner, "Spatiotemporal analysis of event-related fMRI data using partial least squares," *Neuroimage*, vol. 23, no. 2, pp. 764-775, 2004.
- [2] I. Kompatsiaris, and M. Strytzel, "Spatiotemporal Segmentation and Tracking of Objects for Visualization of Videoconference Image Sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 8, pp. 1388-1402, 2000.
- [3] H. Pham, C. Shahabi, Y. Liu, "EBM: an entropy-based model to infer social strength from spatiotemporal data," in *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pp. 265-276, 2013.
- [4] N. Eckert, E. Parent, R. Kies, H. Baya, "A spatio-temporal modelling framework for assessing the fluctuations of avalanche occurrence resulting from climate change: application to 60 years of data in the northern French Alps," *Climatic Change*, vol. 101, no. 3, pp. 515-553, 2010.
- [5] A. Sandryhalia, and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80-90, 2014.
- [6] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83-98, May. 2013.
- [7] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *Proc. 37th IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 3921-3924.
- [8] D. Romero, M. Ma, and G. B. Giannakis, "Kernel-based reconstruction of graph signals," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 2547-2560, May. 2017.
- [9] A. Sandryhalia, and J. M. F. Moura, "Discrete signal processing on graphs: Graph filters," in *Proc. 38th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2013, pp. 6163-6166.
- [10] H. E. Egilmez and A. Ortega, "Spectral anomaly detection using graph-based filtering for wireless sensor networks," in *Proc. 39th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2014, pp. 3892-3896.
- [11] Y. Liu, L. Yang, K. You, W. Guo and W. Wang, "Graph learning based on spatiotemporal smoothness for time-varying graph signal," *IEEE ACCESS*, vol. 7, pp. 62372 - 62386, 2019.
- [12] 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.
- [13] V. Kalofolias, "How to learn a graph from smooth signals," in *Proc. 19th Int. Conf. Artif. Intell. Statist.*, pp. 920-929, May. 2016.
- [14] 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.
- [15] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Process. Mag.*, vol. 36, no. 3, pp. 16-43, May 2019.
- [16] M. T. Bahadori, Q. R. Yu, and Y. Liu, "Fast multivariate spatio-temporal analysis via low rank tensor learning," in *Advances in neural information processing systems*, 2014, pp. 3491-3499.
- [17] 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, 2014.
- [18] 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.
- [19] B. Jiang, C. Ding, and J. Tang, "Graph-laplacian PCA: Closed-form solution and robustness," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2013, pp. 3492-3498.
- [20] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Robust principal component analysis on graphs," in *Proceedings of International Conference on Computer Vision*, Santiago, Chile, 2015, pp. 2812-2820.
- [21] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Fast robust PCA on graphs," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 740-756.
- [22] A. Jung, "Learning the Conditional Independence Structure of Stationary Time Series: A Multitask Learning Approach," in *IEEE Transactions on Signal Processing*, vol. 63, no. 21, pp. 5677-5690, Nov. 1, 2015.
- [23] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432-441, 2008.

- [24] R. Mazumder and T. Hastie, "Exact covariance thresholding into connected components for large-scale graphical lasso," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 781-794, 2012.
- [25] B. M. Lake and J. B. Tenenbaum, "Discovering structure by learning sparse graph," in *Proc. 33rd Annual Cognitive Science conf.*, 2010, pp. 778-783.
- [26] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under structural and laplacian constraints," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 825-841, Sept. 2017.
- [27] S. P. Chepuri, S. Liu, G. Leus, and A. O. Hero, "Learning sparse graphs under smoothness prior," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 6508-6512.
- [28] M. G. Rabbat, "Inferring sparse graphs from smooth signals with theoretical guarantees," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 6533-6537.
- [29] V. Kalofolias, A. Loukas, D. Thanou, and P. Frossard, "Learning time varying graphs," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 2826-2830.
- [30] Yamada, Koki, Yuichi Tanaka, and Antonio Ortega. "Time-varying Graph Learning Based on Sparseness of Temporal Variation." in *Proc. 44th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2019, pp. 5411-5415.
- [31] S. Segarra, A. G. Marques, G. Mateos, and P. Vandergheynst, "Network topology inference from spectral templates," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 467-483, Sept. 2017.
- [32] B. Pasdeloup, V. Gripon, G. Mercier, D. Pastor, and M. G. Rabbat, "Characterization and inference of graph diffusion processes from observations of stationary signals," *IEEE Trans. Signal Inf. Process. Netw.*, vol. PP, no. 99, pp. 1-16, 2017.
- [33] R. Shafipour, S. Segarra, A. G. Marques, and G. Mateos, "Identifying undirected network structure via semidefinite relaxation," in *Proc. 43th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2018, pp. 4049-4053.
- [34] D. Thanou, X. Dong, D. Kressner, and P. Frossard, "Learning heat diffusion graphs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 484-499, Sept. 2017.
- [35] 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.
- [36] B. Baingana and G. B. Giannakis, "Tracking switched dynamic network topologies from information cascades," *IEEE Trans. Signal Process.*, vol. 65, no. 4, pp. 985-997, Feb. 2017.
- [37] Y. Shen, B. Baingana, and G. B. Giannakis, "Topology inference of directed graphs using nonlinear structural vector autoregressive models," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 6513-6517.
- [38] L. Rui, H. Nejati, S. H. Safavi, and N. M. Cheung, "Simultaneous low rank component and graph estimation for high-dimensional graph signals: Application to brain imaging," in *Proc. 42th IEEE Int. Conf. Acoust., Speech, Signal Process.*, IEEE, 2017, pp. 4134-4138.
- [39] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis," *Journal of the ACM*, vol. 58, no. 3, pp. 1-37, May, 2011.
- [40] N. Cressie and C. K. Wile, *Statistics for spatio-temporal data*. John Wiley & Sons, 2011.
- [41] S. Li, K. Li, and Y. Fu, "Temporal subspace clustering for human motion segmentation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 4453-4461, 2015.
- [42] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817-829, 2018.
- [43] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1-122, 2011.
- [44] J. R. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," Pittsburgh, PA, USA, Tech. Rep., 1994.
- [45] S. Boyd and L. Vandenberghe, *Convex optimization*. New York, NY, USA: Cambridge University Press, 2004.
- [46] J. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. on Optimization*, vol. 20, no. 4, pp. 1956-1982, 2010.
- [47] G. H. Golub and C. F. Van Loan, *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, USA, third edition, 1996.
- [48] Dynamic mesh for a dancing man, [Online]. Available: <https://epfl-lts2.github.io/rrp-html/start.html>.
- [49] National Centers for Environmental Information, 2018, [Online]. Available: <https://www.ncdc.noaa.gov/cdo-web/datasets>.
- [50] California Irrigation Management Information System, 2018, [Online]. Available: <https://cimis.water.ca.gov>.
- [51] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge University Press, 2008.
- [52] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *J. Amer. Statist. Assoc.*, vol. 66, no. 336, pp. 846-850, 1971.
- [53] U. Von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395-416, 2007.
- [54] Evapotranspiration Zones in California, 2018, [Online]. Available: <https://www.cimis.water.ca.gov/App-Themes/images/etozonemap.jpg>.