ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9747524

JOINT INFERENCE OF MULTIPLE GRAPHS WITH HIDDEN VARIABLES FROM STATIONARY GRAPH SIGNALS

Samuel Rey[⋆]*, Andrei Buciulea*[⋆]*, Madeline Navarro*[†]*, Santiago Segarra*[†]*, and Antonio G. Marques*[⋆]

[⋆]Dept. of Signal Theory and Communications, King Juan Carlos University, Madrid, Spain
[†]Dept. of Electrical and Computer Engineering, Rice University, Houston, USA

ABSTRACT

Learning graphs from sets of nodal observations represents a prominent problem formally known as graph topology inference. However, current approaches are limited by typically focusing on inferring single networks, and they assume that observations from all nodes are available. First, many contemporary setups involve multiple related networks, and second, it is often the case that only a subset of nodes is observed while the rest remain hidden. Motivated by these facts, we introduce a joint graph topology inference method that models the influence of the hidden variables. Under the assumptions that the observed signals are stationary on the sought graphs and the graphs are closely related, the joint estimation of multiple networks allows us to exploit such relationships to improve the quality of the learned graphs. Moreover, we confront the challenging problem of modeling the influence of the hidden nodes to minimize their detrimental effect. To obtain an amenable approach, we take advantage of the particular structure of the setup at hand and leverage the similarity between the different graphs, which affects both the observed and the hidden nodes. To test the proposed method, numerical simulations over synthetic and real-world graphs are provided.

***Index Terms*—** Network topology inference, graph learning, graph stationarity, hidden nodes, multi-layer graphs

1. INTRODUCTION

Graphs have been successfully exploited to capture the irregular (non-Euclidean) structure commonly inherent to contemporary data for several years now. Increasingly often, several disciplines such as statistics, machine learning, or signal processing (SP), among others, rely on graphs to capture the underlying irregular domain for solving a range of applications on, e.g., communications, genetics, and brain networks [1–4]. However, despite the growing popularity of graph-related methods, in many situations the graph is unknown and we only have access to a set of nodal observations. Then, under the core assumption that the properties of the nodal observations and the topology of the sought graph are closely related, it is possible to learn the network based on the observed signals. This constitutes a prominent problem commonly known as *graph topology inference* [5, 6]. Notable approaches include correlation networks [1, Ch. 7.3.1], partial correlations and (Gaussian) Markov random fields [1, 7, 8], sparse structural equation models [9], graph-SP-based approaches [10–13], as well as their non-linear generalizations [14].

Work supported by the Spanish Fed. Grants FPU17-04520, EST21/00420, and SPGraph PID2019-105032GB-I00; URJC grants F730 and PREDOC20-003; and by the US NSF under award CCF-2008555.

The previous works share the following characteristics: (i) they focus on identifying a single network; and (ii) they assume that observations (measurements) from all the nodes are available. It is relevant to address the first item because many contemporary setups involve *multiple related networks*, each of them with a subset of available signals. This is the case, for example, in multi-hop communication networks in dynamic environments, in social networks where the same set of users may present different types of interactions, or in brain analytics where observations from different patients are available and the goal is to estimate their brain functional networks. When there exist several closely related networks, we can boost the performance of network topology inference by approaching the problem in a joint fashion that allows us to capture the relationship between the different graphs [15–19]. Regarding the second point, assuming that observations from the whole graph are available may not always be realistic. In fact, in many relevant settings the *observed* signals may correspond only to a subset of the nodes from the original graph while the rest of them remain *hidden*. If these hidden nodes are not properly accounted for, they can drastically hinder the performance of the network topology inference methods. Therefore, the presence of hidden variables entails a challenge for most of the existing algorithms, and they require important adjustments. Some works that are starting to deal with this relevant topic include Gaussian graphical model selection [20, 21], inference of linear Bayesian networks [22], nonlinear regression [23], and our previous work based on graph stationarity [24, 25], to name a few.

Based on the previous discussion, the contribution of this paper is to propose a topology inference method that simultaneously performs joint estimation of *multiple* graphs and accounts for the presence of hidden variables. To that end, we rely on results from graph SP (GSP), an area in SP that emerged as a way to generalize tools originally conceived to process signals with regular supports and extend them to signals defined in heterogeneous domains represented by a graph [3, 4, 26]. We assume that the observed signals constitute a random process that is *stationary* on the given graph. As done in our previous work [25], to formalize the relationship between the unknown network and the nodal observations, the graph stationarity assumption must be modified to model the influence of the hidden nodes. Then, to fully benefit from the joint inference formulation, a critical aspect is to capture the similarity among graphs not only accounting for the observed nodes but also for the hidden ones. This is achieved by carefully exploiting the structure inherent to the presence of latent variables with a regularization inspired by group Lasso [27]. The proposed method is evaluated using synthetic and real-world graphs and compared with other related approaches.

The remainder of the paper is organized as follows. Section 2 introduces some GSP concepts leveraged during the paper, Section 3 formally introduces the problem at hand and presents the proposed method, and Section 4 offers a numerical evaluation. Finally, Sec-

tion 5 provides some closing remarks.

2. PRELIMINARIES

We now introduce basic GSP concepts that will help in explaining the relation between the unknown graph and the observed signals while modeling the influence of the hidden variables.

Graph signal processing. Let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ be an undirected graph where $\mathcal{N}$ is the set of nodes with cardinality $|\mathcal{N}| = N$, and $\mathcal{E}$ is the set of edges such that $(i, j) \in \mathcal{E}$ only if nodes i and j are connected. For a given $\mathcal{G}$, the adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ is a (sparse) matrix with non-zero elements A_{ij} if and only if $(i, j) \in \mathcal{E}$. Related to $\mathbf{A}$ is the graph-shift operator (GSO), a square matrix that captures the topology of the graph. The GSO is defined as a matrix $\mathbf{S} \in \mathbb{R}^{N\times N}$ whose entry S_{ij} can be non-zero only if $i = j$ or $(i, j) \in \mathcal{E}$. Typical choices for $\mathbf{S}$ are the adjacency matrix $\mathbf{A}$ and the graph Laplacian $\mathbf{L}$, which is defined as $\mathbf{L} := \text{diag}(\mathbf{A1}) - \mathbf{A}$ [3, 28]. We assume that $\mathbf{S}$ is diagonalizable so it can be represented as $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$, where $\mathbf{V}$ is an $N \times N$ orthogonal matrix collecting the eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix collecting the eigenvalues of $\mathbf{S}$. Then, signals defined on the nodes of $\mathcal{G}$ are called *graph signals*, which are functions $f : \mathcal{N} \mapsto \mathbb{R}$, equivalently represented as vectors $\mathbf{x} = [x_1, ..., x_N]^T \in \mathbb{R}^N$, where x_i denotes the signal value at node i. A key assumption of GSP is that since graph signals are defined on top of the graph, their statistical properties are closely related to the topology of $\mathcal{G}$. A fundamental tool when dealing with graph signals are *graph filters*, linear graph-signal operators that account for the topology of the graph and can be defined as polynomials of the GSO $\mathbf{S}$, i.e., $\mathbf{H} = \sum_{l=0}^{L-1} h_l \mathbf{S}^l$, where $\mathbf{h} = [h_0, ..., h_L]^T$ is the vector collecting the filter coefficients. When applied to an input graph signal $\mathbf{x}$, the output of the graph filter is $\mathbf{y} = \mathbf{H}\mathbf{x} = \sum_{l=0}^{L-1} h_l(\mathbf{S}^l\mathbf{x})$, where $\mathbf{S}^l\mathbf{x}$ can be viewed as the diffusion of $\mathbf{x}$ across an l-hop neighborhood, and h_l are the coefficients of the linear combination [29].

Graph stationarity. A *random* graph signal $\mathbf{x}$ with zero mean and covariance $\mathbf{C} = \mathbb{E}[\mathbf{x}\mathbf{x}^T]$ is said to be stationary in the underlying graph $\mathcal{G}$ if its covariance matrix $\mathbf{C}$ is diagonalized by $\mathbf{V}$, the eigenvectors of the GSO $\mathbf{S}$ [28]. Equivalently[1], a *random* graph process is defined to be stationary in $\mathcal{G}$ if it can be represented as the output of a graph filter $\mathbf{H}$, which is a matrix polynomial in $\mathbf{S}$, to a white input. Specifically, under the stationarity assumption the process $\mathbf{x}$ can be written as $\mathbf{x} = \mathbf{H}\mathbf{w}$, where $\mathbf{w}$ is a random vector of white noise. When particularized to discrete time-varying signals, the two aforementioned definitions boil down to the classical definition of stationarity in time [30]. Also note that graph stationarity implies that the covariance of $\mathbf{x}$ and the GSO commute, so we have that

$$\mathbf{C}\mathbf{S} = \mathbf{S}\mathbf{C}. \tag{1}$$

3. JOINT INFERENCE IN THE PRESENCE OF HIDDEN VARIABLES

To formally introduce the problem of joint graph topology inference in the presence of hidden variables, let us assume that K undirected graphs $\{\mathcal{G}^{(k)}\}_{k=1}^K$ are defined over the same set of nodes $\mathcal{N}$, and denote as $\mathbf{X}^{(k)} = [\mathbf{x}_1^{(k)}, ..., \mathbf{x}_{M_k}^{(k)}] \in \mathbb{R}^{N\times M_k}$ the collection of (zero-mean) M_k signals defined on top of each unknown graph $\mathcal{G}^{(k)}$. Furthermore, consider that for each graph only a subset of nodes $\mathcal{O} \subset \mathcal{N}$ with cardinality $O < N$ is observed, while the remaining H nodes in the subset $\mathcal{H} = \mathcal{N} \setminus \mathcal{O}$ stay hidden. Without loss of generality, let the signals associated with the observed nodes be collected in the first O rows of $\mathbf{X}^{(k)}$ and denote them as $\mathbf{X}_{\mathcal{O}}^{(k)} \in \mathbb{R}^{O\times M_k}$. Then, it can be seen that, for each graph, the unknown GSO $\mathbf{S}^{(k)}$ and the sample covariance $\hat{\mathbf{C}}^{(k)} = \frac{1}{M_k}\mathbf{X}^{(k)}(\mathbf{X}^{(k)})^T$ are symmetric matrices with the following block structure

$$\mathbf{S}^{(k)} = \begin{bmatrix} \mathbf{S}_{\mathcal{O}}^{(k)} & \mathbf{S}_{\mathcal{O}\mathcal{H}}^{(k)} \\ (\mathbf{S}_{\mathcal{O}\mathcal{H}}^{(k)})^T & \mathbf{S}_{\mathcal{H}}^{(k)} \end{bmatrix}, \ \hat{\mathbf{C}}^{(k)} = \begin{bmatrix} \hat{\mathbf{C}}_{\mathcal{O}}^{(k)} & \hat{\mathbf{C}}_{\mathcal{O}\mathcal{H}}^{(k)} \\ (\hat{\mathbf{C}}_{\mathcal{O}\mathcal{H}}^{(k)})^T & \hat{\mathbf{C}}_{\mathcal{H}}^{(k)} \end{bmatrix}. \tag{2}$$

The $O \times O$ matrices $\mathbf{S}_{\mathcal{O}}^{(k)}$ denote the block of the GSOs capturing the connections between the observed nodes while the rest of the submatrices involve edges connected to hidden nodes. Similarly, $\hat{\mathbf{C}}_{\mathcal{O}}^{(k)}$ denotes the sample covariance of the observed signals $\mathbf{X}_{\mathcal{O}}^{(k)}$.

With these considerations in place, the problem of joint topology inference in the presence of hidden variables is introduced next.

Problem 1 *Given the $O \times M_k$ matrices $\{\mathbf{X}_{\mathcal{O}}^{(k)}\}_{k=1}^K$ collecting the signal values at the observed nodes for each graph $\mathcal{G}^{(k)}$, find the sparsest matrices $\{\mathbf{S}_{\mathcal{O}}^{(k)}\}_{k=1}^K$ encoding the structure of the K graphs under the assumptions that:*
(AS1) The number of hidden nodes is much smaller than the number of observed nodes, i.e., $H \ll O$;
(AS2) The signals $\mathbf{X}^{(k)}$ are realizations of a random process that is stationary in $\mathbf{S}^{(k)}$; and
(AS3) The distance between the K graphs is small according to a particular metric $d(\mathbf{S}^{(k)}, \mathbf{S}^{(k')})$.

Accounting for the hidden variables implies modeling their influence over the observed nodes without any additional observation, thus rendering the inference problem a challenge. To ensure the tractability of the problem, (AS1) ensures that most of the nodes are observed while (AS2) establishes a relation between the graph signals and the whole unknown graph, including the hidden nodes. Then, (AS3) guarantees that the K graphs are related so we can benefit from inferring them in a joint setting.

In the following section, we exploit the aforementioned assumptions and the block structure resulting from the presence of hidden variables to approach Problem 1 by solving a convex optimization problem.

3.1. Modeling hidden variables in the joint inference problem

Fundamental to approach Problem 1 is modeling the impact of the hidden nodes in the stationarity assumption (AS2), which implies that the matrices $\mathbf{C}^{(k)}$ and $\mathbf{S}^{(k)}$ are simultaneously diagonalized by the eigenvectors of the kth graph $\mathbf{V}^{(k)}$. To that end, similar to the previous work in [25], we avoid the challenging task of obtaining the submatrix of eigenvectors corresponding to the observed nodes by leveraging the commutativity of the matrices $\mathbf{C}^{(k)}$ and $\mathbf{S}^{(k)}$, and the block structure introduced in (2). More specifically, upon *lifting* the $O\times O$ matrices $\mathbf{P}^{(k)} := \mathbf{C}_{\mathcal{O}\mathcal{H}}^{(k)}(\mathbf{S}_{\mathcal{O}\mathcal{H}}^{(k)})^T$, if we focus on the upper left block at both sides of the equality in (1) we have that the graph stationarity in the presence of hidden nodes is represented by

$$\mathbf{C}_{\mathcal{O}}^{(k)}\mathbf{S}_{\mathcal{O}}^{(k)} + \mathbf{P}^{(k)} = \mathbf{S}_{\mathcal{O}}^{(k)}\mathbf{C}_{\mathcal{O}}^{(k)} + (\mathbf{P}^{(k)})^T. \tag{3}$$

Furthermore, the matrices $\mathbf{P}^{(k)}$ are the product of two matrices of sizes $O \times H$ and $H \times O$. Then, due to (AS1) we have that $H \ll O$, and hence we know that the rank of $\mathbf{P}^{(k)}$ is upper bounded by H.

[1] A small technical condition must hold for these two statements to be equivalent; see [30].

With the previous considerations in place, we approach the sparse joint topology inference problem in the presence of hidden nodes by means of the following non-convex optimization problem

$$\begin{aligned}\min_{\{\mathbf{S}_{\mathcal{O}}^{(k)},\mathbf{P}^{(k)}\}_{k=1}^{K}} \quad & \sum_k \alpha_k \|\mathbf{S}_{\mathcal{O}}^{(k)}\|_0 + \sum_{k<k'} \beta_{k,k'} d_S(\mathbf{S}_{\mathcal{O}}^{(k)}, \mathbf{S}_{\mathcal{O}}^{(k')}) \\ & + \sum_{k<k'} \eta_{k,k'} d_P(\mathbf{P}^{(k)}, \mathbf{P}^{(k')}) \\ \text{s. t.} \quad & \operatorname{rank}(\mathbf{P}^{(k)}) \leq H, \\ & \|\hat{\mathbf{C}}_{\mathcal{O}}^{(k)}\mathbf{S}_{\mathcal{O}}^{(k)} + \mathbf{P}^{(k)} - \mathbf{S}_{\mathcal{O}}^{(k)}\hat{\mathbf{C}}_{\mathcal{O}}^{(k)} - (\mathbf{P}^{(k)})^T\|_F^2 \leq \epsilon, \\ & \mathbf{S}_{\mathcal{O}}^{(k)} \in \mathcal{S}.\end{aligned} \tag{4}$$

The first and second constraints capture assumptions (AS1) and (AS2), with ϵ being a small positive number capturing the fidelity of the sample covariance. The set $\mathcal{S}$ encodes the properties of the desired GSOs. In this paper we will focus on the case where the GSO is given by the adjacency matrix of the underlying undirected graph with non-negative weights and no self-loops. Thus, from now onwards we set the feasibility set $\mathcal{S}$ to be given by

$$\mathcal{S} = \mathcal{S}_{\mathrm{A}} := \{\mathbf{S} \,|\, S_{ij} \geq 0,\ \mathbf{S} = \mathbf{S}^T,\ S_{ii} = 0,\ \textstyle\sum_j S_{j1} = 1\},$$

where the last condition fixes the scale of the admissible graphs by setting the weighted degree of the first node to 1, which rules out the trivial solution $\mathbf{S} = \mathbf{0}$. Other GSOs such as the normalized Laplacian can be accommodated via minor adaptations to $\mathcal{S}$; see [12].

Similar to standard joint inference approaches [17], the objective function of (4) captures the similarity of the K graphs with the function $d_S(\cdot,\cdot)$. Nevertheless, when accounting for the presence of hidden variables, assumption (AS3) is also reflected in the unobserved blocks of the GSOs. This important observation, captured by the function $d_P(\cdot,\cdot)$, allows us to incorporate additional structure reducing the degrees of freedom and rendering the problem more manageable. More specifically, note that the matrix $\mathbf{P}^{(k)}$ is given by the product of $\mathbf{C}_{\mathcal{O}\mathcal{H}}^{(k)}$ and $(\mathbf{S}_{\mathcal{O}\mathcal{H}}^{(k)})^T$ with the latter being a submatrix of a sparse GSO, so it can be seen that the matrices $\mathbf{P}^{(k)}$ present a column-sparse structure. Furthermore, since the K graphs are similar, the submatrices $\mathbf{S}_{\mathcal{O}\mathcal{H}}^{(k)}$ are also similar, which implies that the matrices $\mathbf{P}^{(k)}$ present a similar column-sparsity pattern. In other words, the columns with non-zero entries are likely to be placed in the same positions for the different matrices $\mathbf{P}^{(k)}$. By designing a distance function $d_P(\cdot,\cdot)$ that exploits this additional structure we improve the estimation of the matrices $\mathbf{P}^{(k)}$, resulting in a better estimation of the matrices $\mathbf{S}_{\mathcal{O}}^{(k)}$.

The non-convexity of (4), which arises from the presence of the rank constraint and the ℓ_0 norm, renders the optimization problem computationally hard to solve, leading us to implement some convex relaxations that are detailed next.

3.2. Convex relaxations for the joint topology inference

The rank constraints are commonly avoided by augmenting the objective function with a nuclear norm penalty, which promotes low-rank solutions by seeking matrices with sparse singular values. However, this penalty does not preserve the characteristic column sparsity of the matrices $\mathbf{P}^{(k)}$. To circumvent this issue, in contrast to [25], we employ the group Lasso regularization [27] and rely on the fact that, in this particular setting, we can promote low rankness by reducing the number of non-zero columns while still achieving a reliable estimate. Then, we replace the ℓ_0 norm by a reweighted ℓ_1 minimization [31], an iterative algorithm rooted on a logarithmic penalty that: i) converges to a stationary point [32]; and ii) usually outperforms the widely used ℓ_1 norm.

By leveraging the aforementioned relaxations we address the joint topology inference problem in the presence of hidden variables by solving an iterative method. Under this approach, for each iteration, we solve the following convex problem

$$\begin{aligned}\min_{\{\mathbf{S}_{\mathcal{O}}^{(k)},\mathbf{P}^{(k)}\}_{k=1}^{K}} \quad & \sum_k \alpha_k \operatorname{vec}(\mathbf{W}^{(k)})^T \operatorname{vec}(\mathbf{S}_{\mathcal{O}}^{(k)}) \\ & + \sum_{k<k'} \beta_{k,k'} \|\mathbf{S}_{\mathcal{O}}^{(k)} - \mathbf{S}_{\mathcal{O}}^{(k')}\|_1 \\ & + \sum_k \gamma_k \|\mathbf{P}^{(k)}\|_{2,1} + \sum_{k<k'} \eta_{k,k'} \left\| \begin{bmatrix} \mathbf{P}^{(k)} \\ \mathbf{P}^{(k')} \end{bmatrix} \right\|_{2,1} \\ & + \sum_k \mu_k \|\hat{\mathbf{C}}_{\mathcal{O}}^{(k)}\mathbf{S}_{\mathcal{O}}^{(k)} + \mathbf{P}^{(k)} - \mathbf{S}_{\mathcal{O}}^{(k)}\hat{\mathbf{C}}_{\mathcal{O}}^{(k)} - (\mathbf{P}^{(k)})^T\|_F^2 \\ \text{s. t.} \quad & \mathbf{S}_{\mathcal{O}}^{(k)} \in \mathcal{S}.\end{aligned} \tag{5}$$

To compute the weight matrices $\mathbf{W}^{(k)}$, let $t = 1...T$ denote the iteration index (omitted in the expression above to alleviate the notation), and compute the kth weight matrix for the tth iteration as $W_{ij}^{(k,t)} = (S_{\mathcal{O}_{ij}}^{(k,t-1)} + \delta)^{-1}$ with $S_{\mathcal{O}_{ij}}^{(k,t-1)}$ being the solution obtained during the $t-1$th iteration and δ a small positive constant. Hence, for each iteration t we first compute the weight matrices $\mathbf{W}^{(k,t)}$ and, then, employ those to estimate the matrices $\mathbf{S}_{\mathcal{O}}^{(k,t)}$ and $\mathbf{P}^{(k,t)}$. Coming back to the formulation in (5), note that the distance $d_S(\cdot,\cdot)$ is set to the ℓ_1 norm to promote similar edges on the K graphs. The norm $\|\cdot\|_{2,1}$ represents the group Lasso penalty by first computing the ℓ_2 norm of the columns of the input matrix and then the ℓ_1 norm of the resulting vector. To capture the similar column-sparsity pattern of the matrices $\mathbf{P}^{(k)}$ resulting from the similarity of the K graphs, we design the function $d_P(\cdot,\cdot)$ relying on the group Lasso penalty. More specifically, we concatenate each pair of matrices $\mathbf{P}^{(k)}$ and $\mathbf{P}^{(k')}$ to create a tall matrix and then promote column sparsity on the tall matrix with the $\ell_{2,1}$ norm. Note that a column of all zeros in the tall matrix implies that the same column in $\mathbf{P}^{(k)}$ and $\mathbf{P}^{(k')}$ will only contain zeros, thus promoting the desired structure. Finally, it is worth noting that we moved the commutativity constraint to the objective function. Due to the iterative nature of the proposed method, the estimation of the observed GSO during the first iteration might be far from the true GSO, and hence, a more restrictive constraint as the one employed in (4) might be misleading. Augmenting the objective function with the commutativity penalty is more amenable to an iterative approach.

4. NUMERICAL EXPERIMENTS

We now introduce the simulations performed over synthetic and real-world graphs to evaluate the proposed method. When graphs are randomly generated, they are sampled from an Erdős-Rényi (ER) model with $N = 20$ nodes and edge probability $p = 0.2$. The code for the following experiments is available on GitHub[2] and the interested reader is referred there for specific implementation details.

Test case 1. In the first experiment, we evaluate the influence of the hidden variables and its detrimental effect on the topology inference task when the true covariance matrix is known. The results

[2] https://github.com/reysam93/hidden_joint_inference/tree/ICASSP2022

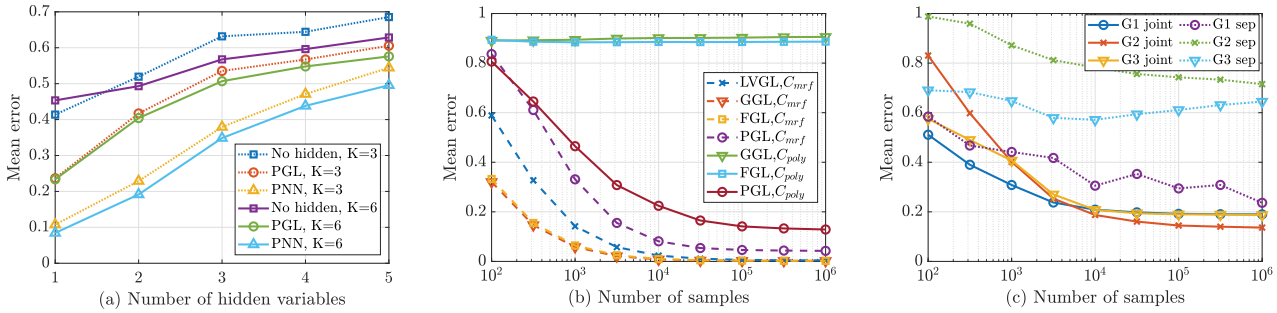

Fig. 1: Numerical validation of the proposed algorithm. a) Mean error of 100 realizations as the number of hidden variables increases for different models and values of K. b) Mean error of the recovered graphs for several algorithms as the number of samples increases. c) Mean error of the recovered graphs for joint and separate approaches as the number of samples increases. The first two experiments use Erdős-Rényi graphs with $N = 20$ and $p = 0.2$ and the third one employs real-world graphs.

are depicted in Figure 1a, where we report the error of the recovered graphs, computed as $\sum_{k=1}^{K} \|\mathbf{S}_{\mathcal{O}}^{(k)} - \hat{\mathbf{S}}_{\mathcal{O}}^{(k)}\|_F^2 / K \|\mathbf{S}_{\mathcal{O}}^{(k)}\|_F^2$, for several models as the number of hidden variables increases on the x-axis. The error is averaged over 64 realizations with ER graphs. The considered models are: (i) "PGL", which stands for the method introduced in (5); (ii) "PNN", which denotes the reweighed algorithm proposed in [24] augmented with the joint penalty $d_S(\cdot,\cdot)$ to perform the joint inference; and (iii) "No hidden", which is a joint inference method unaware of the presence of hidden variables similar to the work in [17]. In addition, for each model we let K take the values in $\{3, 6\}$. Looking at the results, we can observe that "PGL" and "PNN", which take into account the presence of hidden variables, outperform the method "No hidden", showcasing the benefit of a robust formulation. Also, the method proposed in (5) outperforms "PNN", the other alternative accounting for hidden variables. This reflects the advantage of employing the group Lasso regularization and incorporating the graph similarity through the careful design of the function $d_P(\cdot,\cdot)$. Lastly, it is worth noting that the performance improves for higher values of K, achieving better results when more related graphs are available.

Test case 2. Next, we evaluate the influence of the number of observed signals and compare the performance of the proposed approach with other related alternatives. In this experiment, only a single hidden node is considered. To that end, in Figure 1b we show the mean normalized error of the recovered graphs on the y-axis as the number of samples increases on the x-axis. The error is computed as in the previous experiment and the mean is considered over 30 realizations of $K = 3$ ER graphs with 10 realizations of random covariance matrices for each, resulting in a total of 300 realizations. We compare the proposed model ("PGL") with latent variable graphical Lasso ("LVGL") [20], and with group and fusion graphical Lasso ("GGL" and "FGL"), both from [16]. For each model, signals are generated using two different types of covariance matrices: (i) $\mathbf{C}_{mrf} = (\sigma\mathbf{I} + \phi\mathbf{S})^{-1}$ where ϕ is a positive random number and σ is a positive number so that $\mathbf{C}_{MRF}^{-1}$ is positive semi-definite; and $\mathbf{C}_{poly} = \mathbf{H}^2$ where the matrix $\mathbf{H}$ is a graph filter with random coefficients $\mathbf{h}$. By looking at the Figure 1b, it can be observed that, when $\mathbf{C}_{mrf}$ is employed, the graphical Lasso models slightly outperform the proposed approach. This is expected since they are tailored for this specific type of covariance matrices. However, we can also see that the performance of the proposed algorithm is close to that of the alternatives, illustrating the benefits of considering both the joint optimization and the presence of hidden variables. On the other hand, when we focus on the covariance matrices $\mathbf{C}_{poly}$, it is evident that the proposed method "PGL" clearly outperforms the alternatives, demonstrating that the proposed method is based on more general assumptions. Note that the results for "LVGL" for the polynomial covariance are not included since the error was too high.

Test case 3. Finally, we test the proposed algorithm and the impact of performing the topology inference in a joint fashion using real-world graphs. We employ three graphs defined on a common set of 32 nodes. Nodes represent students from the University of Ljubljana and the different networks encode different types of interactions among the students[3]. The error is computed as before and one hidden variable is considered. The results, illustrated in Figure 1c, show the error of the recovered graphs as the number of samples increases. The displayed error is the mean of 30 realizations of random stationary graph signals and only one hidden variable is considered. Also, for each of the three graphs we include the performance of both the joint and the separate estimation. It can be observed that the recovery of the three graphs improves when a joint approach is followed, showcasing the benefits of exploiting the existing relationship between the different networks. Furthermore, this experiment confirms that the developed method is also suitable for real applications.

5. CONCLUSIONS

In this paper, we presented a new method for solving the challenging problem of joint graph topology inference in the presence of hidden nodes. To tackle this ill-posed inference problem, we assume that (i) the number of hidden nodes H is much smaller than the number of observed nodes O; (ii) the observed signals are realizations from a random process stationary in $\mathbf{S}^{(k)}$; and (iii) the K graphs are closely related. Furthermore, we exploit the inherent block structure of the matrices $\mathbf{C}^{(k)}$ and $\mathbf{S}^{(k)}$ to solve the joint topology inference problem by solving an optimization framework. A reweighted ℓ_1 norm to promote sparse solutions is employed, and the stationarity assumption is adapted to the presence of hidden nodes by defining the (unknown) low-rank lifting matrices $\mathbf{P}^{(k)}$. Instead of relying in the nuclear norm, low-rank matrices $\mathbf{P}^{(k)}$ are achieved by promoting column-sparsity with the group Lasso penalty. Moreover, the similarity of the K graphs is leveraged in two ways. First, for each pair of graphs, we look for matrices $\mathbf{S}_{\mathcal{O}}^{(k)}$ with a similar edge pattern by minimizing the ℓ_1 penalty, and second, we look for matrices $\mathbf{P}^{(k)}$ with a similar column sparsity pattern. The proposed method is evaluated using synthetic and real world graphs, and a comparison with other baseline methods based on graph stationarity and on graphical Lasso is provided.

[3]The original data can be found at `http://vladowiki.fmf.uni-lj.si/doku.php?id=pajek:data:pajek:students`

6. REFERENCES

[1] E. D. Kolaczyk, *Statistical Analysis of Network Data: Methods and Models*, Springer, New York, NY, 2009.

[2] O. Sporns, *Discovering the Human Connectome*, MIT Press, Boston, MA, 2012.

[3] 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.

[4] A. Ortega, P. Frossard, J. Kovacevic, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proc. IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[5] 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, 2019.

[6] S. Sardellitti, S. Barbarossa, and P. Di Lorenzo, "Graph topology inference based on sparsifying transform learning," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1712–1727, 2019.

[7] N. Meinshausen and P. Buhlmann, "High-dimensional graphs and variable selection with the lasso," *Ann. Statist.*, vol. 34, pp. 1436–1462, 2006.

[8] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[9] B. Baingana, G. Mateos, and G. B. Giannakis, "Proximal-gradient algorithms for tracking cascades over social networks," *IEEE J. Sel. Topics Signal Process.*, vol. 8, pp. 563–575, Aug. 2014.

[10] J. Mei and J.M.F. Moura, "Signal processing on graphs: Estimating the structure of a graph," in *IEEE Intl. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 2015, pp. 5495–5499.

[11] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under laplacian and structural constraints," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 825–841, 2017.

[12] S. Segarra, A. G. Marques, G. Mateos, and A. Ribeiro, "Network topology inference from spectral templates," *IEEE Trans. Signal Info. Process. Networks*, vol. 3, no. 3, pp. 467–483, Sep. 2017.

[13] S. Segarra, A. G. Marques, M. Goyal, and S. Rey, "Network topology inference from input-output diffusion pairs," in *IEEE Wrkshp. Statistical Signal Process. (SSP)*. IEEE, 2018, pp. 508–512.

[14] G. V. Karanikolas, G. B. Giannakis, K. Slavakis, and R. M. Leahy, "Multi-kernel based nonlinear models for connectivity identification of brain networks," in *IEEE Intl. Conf. Acoust., Speech and Signal Process. (ICASSP)*, Shanghai, China, Mar. 20-25, 2016.

[15] Y. Murase, J. Török, H. H. Jo, K. Kaski, and J. Kertész, "Multilayer weighted social network model," *Physical Review E*, vol. 90, no. 5, pp. 052810, 2014.

[16] P. Danaher, P. Wang, and D. M. Witten, "The joint graphical lasso for inverse covariance estimation across multiple classes," *J. of the Roy. Statistical Soc.: Ser. B (Statistical Methodology)*, vol. 76, no. 2, pp. 373–397, 2014.

[17] M. Navarro, Y. Wang, A. G. Marques, C. Uhler, and S. Segarra, "Joint inference of multiple graphs from matrix polynomials," *arXiv preprint arXiv:2010.08120*, 2020.

[18] J. Arroyo, A. Athreya, Joshua Cape, Guodong C., Carey E. P., and J. T. Vogelstein, "Inference for multiple heterogeneous networks with a common invariant subspace," *J. Mach. Learn. Res.*, vol. 22, no. 142, pp. 1–49, 2021.

[19] Y. Wang, S. Segarra, and C. Uhler, "High-dimensional joint estimation of multiple directed Gaussian graphical models," *Electron. J. Statist.*, vol. 14, no. 1, pp. 2439–2483, 2020.

[20] V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky, "Latent variable graphical model selection via convex optimization," *Ann. of Statist.*, vol. 40, no. 4, pp. 1935–1967, 2012.

[21] A. Chang, T. Yao, and G. I. Allen, "Graphical models and dynamic latent factors for modeling functional brain connectivity," in *2019 IEEE Data Science Wrksp. (DSW)*. IEEE, 2019, pp. 57–63.

[22] A. Anandkumar, D. Hsu, A. Javanmard, and S. Kakade, "Learning linear bayesian networks with latent variables," in *Intl. Conf. Machine Learning (ICML)*, 2013, pp. 249–257.

[23] J. Mei and J. M. F. Moura, "Silvar: Single index latent variable models," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2790–2803, 2018.

[24] A. Buciulea, S. Rey, C. Cabrera, and A. G Marques, "Network reconstruction from graph-stationary signals with hidden variables," in *Asilomar Conf. Signals, Systems, and Computers*. IEEE, 2019, pp. 56–60.

[25] A. Buciulea, S. Rey, and A. G Marques, "Learning graphs from smooth and graph-stationary signals with hidden variables," *arXiv preprint arXiv:2111.05588*, 2021.

[26] A. Sandryhaila and J.M.F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.

[27] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *J. of Comput. and Graphical Statist.*, vol. 22, no. 2, pp. 231–245, 2013.

[28] P. Djuric and C. Richard, *Cooperative and Graph Signal Processing: Principles and Applications*, Academic Press, 2018.

[29] S. Segarra, A. G. Marques, and A. Ribeiro, "Optimal graph-filter design and applications to distributed linear network operators," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4117–4131, 2017.

[30] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5911–5926, 2017.

[31] E. J Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. of Fourier Anal. and Appl.*, vol. 14, no. 5-6, pp. 877–905, 2008.

[32] M. Hong, M. Razaviyayn, Z. Luo, and J. Pang, "A unified algorithmic framework for block-structured optimization involving big data: With applications in machine learning and signal processing," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 57–77, 2015.