Bayesian Estimation of Graph Signals

Ariel Kroizer, Tirza Routtenberg [ID], *Senior Member, IEEE*, and Yonina C. Eldar, *Fellow, IEEE*

Abstract—**We consider the problem of recovering random graph signals from nonlinear measurements. For this setting, closed-form Bayesian estimators are usually intractable and even numerical evaluation may be difficult to compute for large networks. In this paper, we propose a graph signal processing (GSP) framework for random graph signal recovery that utilizes information on the structure behind the data. First, we develop the GSP-linear minimum mean-squared-error (GSP-LMMSE) estimator, which minimizes the mean-squared-error (MSE) among estimators that are represented as an output of a graph filter. The GSP-LMMSE estimator is based on *diagonal* covariance matrices in the graph frequency domain, and thus, has reduced complexity compared with the LMMSE estimator. This property is especially important when using the sample-mean estimators that are based on a training dataset. We then state conditions under which the low-complexity GSP-LMMSE estimator coincides with the optimal LMMSE estimator. Next, we develop an approximate parametrization of the GSP-LMMSE estimator by graph filters. We present three implementations of the parametric GSP-LMMSE estimator for typical graph filters. These parametric graph filters are more robust to outliers and to network topology changes. In our simulations, we evaluate the performance of the proposed GSP-LMMSE estimators for the problem of state estimation in power systems, which can be interpreted as a graph signal recovery task. We show that the proposed sample-GSP estimators outperform the sample-LMMSE estimator for a limited training dataset and that the parametric GSP-LMMSE estimators are more robust to topology changes in the form of adding/removing vertices/edges.**

Index Terms—**Graph signal processing (GSP), graph filters, Bayesian estimation, linear minimum mean-squared-error (LMMSE) estimator, sample-LMMSE estimator, GSP-LMMSE estimator, graph signal recovery.**

I. INTRODUCTION

GRAPH signals arise in various applications such as the study of brain signals [1] and sensor networks [2]. The area of graph signal processing (GSP) has gained considerable interest in the last decade. GSP theory extends concepts and techniques from traditional digital signal processing (DSP) to data indexed by generic graphs, including the graph Fourier transform (GFT), graph filter design [3]–[5], and sampling and recovery of graph signals [6]–[9]. Many modern network applications involve complex models and large datasets and are characterized by nonlinear models [10], [11], for example, the brain network connectivity [12], environment monitoring [13], and power flow equations in power systems [14]–[16]. The recovery of graph signals in such networks is often intractable, especially for large networks. For example, the recovery of voltages from power measurements is an NP-hard nonconvex optimization problem [17], which is at the core of power system analysis [18]. In this case, the graph represents an electrical network, and the signals are the voltages. Thus, the development of GSP methods for the estimation of graph signals has significant practical importance, in addition to its contribution to the enrichment of theoretical statistical GSP tools.

The recovery of random graph signals can be performed by state-of-the-art Bayesian estimators, such as the minimum mean-squared-error (MMSE) and the linear MMSE (LMMSE) estimators. However, MMSE estimation is often computationally intractable and does not have a closed-form expression in nonlinear models. The LMMSE can be used when the second-order statistics are completely specified. In some cases (e.g. [19]–[21]), accurate characterization of the nonlinear model is possible by using tools such as Bussgang's theorem. However, for the general case, the distributions (e.g. the covariance matrices) of the desired graph signal and the observations are difficult to determine. Moreover, in many practical applications, the graph signal has a broad correlation function so that estimating this correlation from data with high accuracy often necessitates a larger sample size than is available [22] and requires stationarity of the signals. Low-complexity estimation algorithms have been considered as an alternative. For example, in [23], a low-rank approximation is applied to the LMMSE estimator by using the singular value decomposition (SVD) of the covariance matrices. The dual-diagonal LMMSE (DD-LMMSE) channel estimation algorithm, which is based on the diagonal of the covariance matrices, was proposed in [24]. However, these methods may lead to considerable performance loss compared with LMMSE estimation.

Graph filters have been used for many signal processing tasks, such as denoising [25], [26], classification [27], and anomaly detection [14]. The design of graph filters to obtain a desired graph frequency response for the general case has been studied and analyzed in various works [5], [28]–[30]. Model-based recovery of a graph signal from noisy measurements by graph filters for *linear* models was treated in [5], [25], [31], [32]. To derive classical graph filters, such as the Wiener filter, more restrictive assumptions on the graph signal are required [32]. Nonlinear graph filters were considered in [13], but they require higher-order statistics that are not completely specified in the general case. Graph neural network approaches were considered in [33], [34]; however, data-based methods necessitate extensive training sets, and result in nonlinear estimators, while in this paper, we focus on linear estimation with limited training data.

Fitting graph-based models to given data was considered in [35]–[37]. In [38], we proposed a two-stage method for graph signal estimation from a known nonlinear observation model, which is based on fitting a graph-based model and then implementing least-squares recovery on the approximate model. However, model-fitting approaches aim to minimize the modeling error and in general have significantly lower performance than estimators that minimize the estimation error directly. A fundamental question remains regarding how to use knowledge of the nonlinear physical model and on the graph topology to obtain a low-complexity estimator for the general nonlinear case that has optimal MSE performance.

In this paper, we consider the estimation of random graph signals from a nonlinear observation model. First, we present the sample-LMMSE estimator, in which the analytical expressions of the LMMSE estimator are replaced by their estimated values; this requires use of large training datasets. Next, we propose the GSP-LMMSE estimator, which minimizes the MSE among the subset of estimators that are represented as an output of a graph filter. We discuss the advantages of the GSP-LMMSE estimator in terms of complexity and show that 1) for models with diagonal covariance matrices in the graph frequency domain, the GSP-LMMSE and LMMSE estimators coincide; and 2) for linear models, the GSP-LMMSE estimator coincides with the graphical Wiener filter [32]. We develop the MSE-optimal parametrization of the GSP-LMMSE estimator by shift-invariant graph filters that approximate the graph frequency response of the GSP-LMMSE estimator. The parameterized estimators can be applied to any topology, making them more robust to network topology changes. We also implement three types of the parameterized GSP-LMMSE estimator based on well-known graph filters. Finally, we perform numerical simulations for the problem of state estimation in power systems. We show that in this case the sample-GSP estimators outperform the sample-LMMSE estimator for a limited training dataset and coincide with the sample-LMMSE estimator otherwise. Moreover, the proposed estimators are more robust to changes in the network topology in the form of adding/removing vertices/edges.

The rest of this paper is organized as follows. In Section II we introduce the basics of GSP and three examples of graph filters. In Section III, we formulate the estimation problem and present the MMSE and the LMMSE estimators. In Section IV, we develop the proposed GSP-LMMSE estimator and present parameterizations of the GSP-LMMSE estimator in Section V. Simulation are shown in Section VI. Finally, the paper concluded in Section VII.

We denote vectors by boldface lowercase letters and matrices by boldface uppercase letters. The operators $(\cdot)^T$, $(\cdot)^{-1}$, and $(\cdot)^\dagger$ represent the transpose, inverse, and pseudo-inverse, respectively. The notation $\circ$ denotes the Hadamard product. For a matrix $\mathbf{A}$, rank($\mathbf{A}$) is its rank. For a vector $\mathbf{a}$, diag($\mathbf{a}$) is a diagonal matrix whose ith diagonal entry is $\mathbf{a}_i$; when applied to a matrix, diag($\mathbf{A}$) is a vector collecting the diagonal elements of $\mathbf{A}$. The mth element of the vector $\mathbf{a}$ is written as a_m or $[\mathbf{a}]_m$. The (m,q)th element of the matrix $\mathbf{A}$ is written as $A_{m,q}$ or $[\mathbf{A}]_{m,q}$. The identity matrix of dimension N is written as $\mathbf{I}_N$ and the vector $\mathbf{0}_N$ is a length N vector of all zeros. The multivariate Gaussian distribution of $\mathbf{y}$ with mean vector, $\boldsymbol{\mu}$, and covariance matrix, $\boldsymbol{\Sigma}$, is denoted by $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The cross-covariance matrix of the vectors $\mathbf{a}$ and $\mathbf{b}$ is denoted by $\mathbf{C_{ab}} \triangleq \mathbb{E}[(\mathbf{a} - \mathbb{E}[\mathbf{a}])(\mathbf{b} - \mathbb{E}[\mathbf{b}])^T]$.

II. BACKGROUND: GRAPH SIGNAL PROCESSING (GSP)

We begin by reviewing GSP and general graph filters in Section II-A. Three commonly-used graph filters are then presented in Sections II-B, II-C, II-D, and will be used later in the paper.

A. GSP Background

Consider an undirected, connected, weighted graph $\mathcal{G}(\mathcal{V}, \xi, \mathbf{W})$, where $\mathcal{V}$ and ξ are sets of vertices and edges, respectively. The matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ is the non-negative weighted adjacency matrix of the graph, where $N \triangleq |\mathcal{V}|$ is the number of vertices in the graph. If there is an edge $(i,j) \in \xi$ connecting vertices i and j, the entry $\mathbf{W}_{i,j}$ represents the weight of the edge; otherwise, $\mathbf{W}_{i,j} = 0$. A common way to represent the graph topology is by the Laplacian matrix, which is defined by

$$\mathbf{L} \triangleq \text{diag}(\mathbf{W1}) - \mathbf{W}. \quad (1)$$

The Laplacian matrix, $\mathbf{L}$, is a real and positive semidefinite matrix and its eigenvalue decomposition (EVD) is given by

$$\mathbf{L} = \mathbf{V\Lambda V}^T, \quad (2)$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix consisting of the eigenvalues of $\mathbf{L}$, $0 = \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_N$, $\mathbf{V}$ is a matrix whose nth column, $\mathbf{v}_N$, is the eigenvector of $\mathbf{L}$ that is associated with λ_n, and $\mathbf{V}^T = \mathbf{V}^{-1}$. We assume, without loss of generality, that $\mathcal{G}$ is a connected graph, i.e., that $\lambda_2 \neq 0$ [39].

In this paper, a *graph signal* is an N-dimensional vector, $\mathbf{a}$, that assigns a scalar value to each vertex, i.e., each entry a_n denotes the signal value at vertix n, for $n = 1, \ldots, N$. The GFT of the graph signal $\mathbf{a}$ is defined as [4]

$$\tilde{\mathbf{a}} \triangleq \mathbf{V}^T \mathbf{a}. \quad (3)$$

Similarly, the inverse GFT (IGFT) of $\tilde{\mathbf{a}}$ is given by $\mathbf{V}\tilde{\mathbf{a}}$. Finally, a graph signal is a graph-bandlimited signal with cutoff graph frequency N_s if it satisfies [3]

$$\tilde{a}_n = 0, \ n = N_s + 1, \ldots, N. \quad (4)$$

Graph filters are useful tools for various GSP tasks. Linear and shift-invariant graph filters with respect to (w.r.t.) the graph shift operator (GSO) play essential roles in GSP. These filters generalize linear time-invariant filters used in DSP for time series, and enable performing tractable operations over graphs [3], [4]. A graph filter is a function $f(\cdot)$ applied to a GSO, where here we use the GSO given by $\mathbf{L}$, that allows an eigendecomposition as follows [3]:

$$f(\mathbf{L}) = \mathbf{V}f(\boldsymbol{\Lambda})\mathbf{V}^T, \quad (5)$$

where $f(\boldsymbol{\Lambda})$ is a diagonal matrix. That is, $f(\lambda_n)$ is the graph frequency response of the filter at graph frequency λ_n, $n = 1, \ldots, N$, and $f(\mathbf{L})$ is diagonalized by the eigenvector matrix $\mathbf{V}$ of $\mathbf{L}$. We assume that the graph filter, $f(\cdot)$, is a well-defined function on the spectrum $\{\lambda_1, \ldots, \lambda_N\}$ of $\mathbf{L}$.

Throughout this paper, we consider three commonly-used parametrizations of the graph filter function, $f(\cdot)$, that are appropriate for modeling low-pass graph filters. For clarity of representation, we will write this specific function as $h(\cdot; \boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ contains the graph filter parameters. It should be noted that under simple conditions, all filters in the form of (5) can be represented as a finite polynomial of $\mathbf{L}$ [40]. Linear graph filters can be implemented locally, e.g., with exchanges of information among neighbors [40]. However, due to the nature of the polynomial fitting problem, linear graph filters usually

have limited accuracy when used for approximating a desired graph frequency response (see, e.g. in [28]–[30]). This is further discussed in Section IV, where the benefits of the presented parametrization in this context are outlined.

B. Linear Pseudo-Inverse Graph Filter

In a linear graph filter, the filter is a polynomial of a GSO, such as the Laplacian matrix [3], [4]. In addition, the Moore-Penrose pseudo-inverse of the Laplacian matrix plays an essential role in many graph-based applications [41], [42]. Inspired by the linear graph filter, we define the linear pseudo-inverse graph filter, $f(\mathbf{L}) = h^{\text{LPI}}(\mathbf{L}^{\dagger}; \boldsymbol{\alpha}^{\text{LPI}})$, as

$$h^{\text{LPI}}(\mathbf{L}^{\dagger}; \boldsymbol{\alpha}^{\text{LPI}}) \triangleq h_0\mathbf{I}_N + h_1\mathbf{L}^{\dagger} + \cdots + h_K(\mathbf{L}^{\dagger})^K, \quad (6)$$

where K is the filter order and the filter coefficients vector is

$$\boldsymbol{\alpha}^{\text{LPI}} = \begin{bmatrix} h_0, \ldots, h_K \end{bmatrix}^T \in \mathbb{R}^{K+1}, \quad (7)$$

with $K < N$ due to the Cayley-Hamilton theorem. From (5) and (6), we conclude that the frequency response of the linear pseudo-inverse graph filter at graph frequency λ_n can be expressed as

$$h^{\text{LPI}}(\lambda_n; \boldsymbol{\alpha}^{\text{LPI}}) = \begin{cases} h_0, & n = 1 \\ \sum_{k=0}^K h_k \lambda_n^{-k}, & n = 2, \ldots, N. \end{cases} \quad (8)$$

C. Autoregressive Moving Average (ARMA) Graph Filter

Similar to temporal ARMA filters [43], an ARMA graph filter is characterized by a rational polynomial in the Laplacian matrix [5]. In this case, $f(\mathbf{L}) = h^{\text{ARMA}}(\mathbf{L}; \boldsymbol{\alpha}^{\text{ARMA}})$, where the ARMA graph filter is defined as

$$h^{\text{ARMA}}(\mathbf{L}; \boldsymbol{\alpha}^{\text{ARMA}}) \triangleq \left(\mathbf{I}_N + \sum_{r=1}^R a_r\mathbf{L}^r \right)^{-1} \sum_{q=0}^Q c_q\mathbf{L}^q \quad (9)$$

and the ARMA filter coefficient vector is

$$\boldsymbol{\alpha}^{\text{ARMA}} = [\mathbf{a}^T, \mathbf{c}^T]^T, \quad (10)$$

where $\mathbf{a} = [1, a_1, \ldots, a_R]^T$ and $\mathbf{c} = [c_0, \ldots, c_Q]^T$. It is assumed that $\mathbf{I}_N + \sum_{r=1}^R a_r\mathbf{L}^r$ is a non-singular matrix. From (5) and (9), the graph-ARMA filter frequency response at graph frequency λ_n can be expressed as

$$h^{\text{ARMA}}(\lambda_n; \boldsymbol{\alpha}^{\text{ARMA}}) = \frac{\sum_{q=0}^Q c_q\lambda_n^q}{1 + \sum_{r=1}^R a_r\lambda_n^r}, \quad (11)$$

$n = 1, \ldots, N$, where we assume that $1 + \sum_{r=1}^R a_r\lambda_n^r \neq 0$.

The linear graph filter, which is a polynomial of the Laplacian matrix, $\mathbf{L}$:

$$h^{\text{lin}}(\mathbf{L}; \boldsymbol{\alpha}^{\text{lin}}) \triangleq h_0\mathbf{I}_N + h_1\mathbf{L} + \cdots + h_K\mathbf{L}^K, \quad (12)$$

where K is the filter order and $\boldsymbol{\alpha}^{\text{lin}} = \begin{bmatrix} h_0, \ldots, h_K \end{bmatrix}^T$ is the filter coefficients vector, can be obtained as a special case of the ARMA graph filter by setting $a_r = 0, r = 1, \ldots, R$ and $Q = K$ in (9). However, the ARMA graph filter improves the accuracy of the approximation of the desired graph frequency response and requires fewer coefficients compared to linear graph filters [29]. In addition, unlike the linear pseudo-inverse graph filter, the ARMA graph filter has distributed implementations, as shown in [29], [44].

D. Low-Rank Graph Filter

In many GSP applications, the graph signals are assumed to be *bandlimited* in the graph spectrum domain [45]. In particular, under the assumption of a low-frequency graph signal, we are interested in filtering only the lowest graph frequencies and equating the output graph signal at high graph frequencies to zero. To this end, we use a low-rank graph filter.

A low-rank graph filter can be designed based on any graph filter, $f(\cdot)$, by substituting the following low-rank matrix:

$$\bar{\mathbf{L}} = \sum_{n=1}^{N_s} \lambda_n\mathbf{v}_n\mathbf{v}_n^T, \quad (13)$$

which contains the N_s smallest eigenvalues of the Laplacian matrix, $\mathbf{L}$, and their associated eigenvectors, in $f(\cdot)$. That is, we use the filter $f(\bar{\mathbf{L}})$ instead of $f(\mathbf{L})$. Note that, rank $(\bar{\mathbf{L}}) = N_s - 1 < N - 1$ and that $\bar{\mathbf{L}}$ is not a Laplacian matrix. For these filters, we use the convention that the zero power of the matrix $\bar{\mathbf{L}}$ is

$$\bar{\mathbf{L}}^0 = \sum_{n=1}^{N_s} \mathbf{v}_n\mathbf{v}_n^T. \quad (14)$$

Substituting (13) in (5), the frequency response of a low-rank graph filter at the graph frequency λ_n is

$$[f(\boldsymbol{\Lambda})]_{n,n} = \begin{cases} f(\lambda_n), & n = 1, \ldots, N_s \\ 0, & n = N_s + 1, \ldots, N. \end{cases} \quad (15)$$

The advantage of the low-rank graph filter is that it requires less filter coefficients and computation of only the N_s smallest eigenvalues and eigenvectors. As a result, the evaluation of the filter coefficients has lower computational complexity.

In this paper, we use the low-rank ARMA (LR-ARMA) graph filter,

$$f(\bar{\mathbf{L}}) = h^{\text{ARMA}}(\bar{\mathbf{L}}; \boldsymbol{\alpha}^{\text{LR}}), \quad (16)$$

where $h^{\text{ARMA}}(\cdot; \cdot)$ is defined in (9) and the reduced, low-rank Laplacian matrix is defined in (13). The filter coefficients vector in this case is given by

$$\boldsymbol{\alpha}^{\text{LR}} = [(\mathbf{a}^{\text{LR}})^T, (\mathbf{c}^{\text{LR}})^T]^T, \quad (17)$$

where $\mathbf{a}^{\text{LR}} = [1, a_1^{\text{LR}}, \ldots, a_R^{\text{LR}}]^T$ and $\mathbf{c}^{\text{LR}} = [c_0^{\text{LR}}, \ldots, c_Q^{\text{LR}}]^T$. From (5), (9), (11), and (15), the graph-LR-ARMA filter frequency response at graph frequency λ_n can be expressed as

$$h^{\text{LR-ARMA}}(\lambda_n; \boldsymbol{\alpha}^{\text{LR}}) = \begin{cases} \frac{\sum_{q=0}^{Q_{\text{LR}}} c_q^{\text{LR}}\lambda_n^q}{1 + \sum_{r=1}^{R_{\text{LR}}} a_r^{\text{LR}}\lambda_n^r}, & n = 1, \ldots, N_s \\ 0, & n = N_s + 1, \ldots, N, \end{cases} \quad (18)$$

where we assume that $1 + \sum_{r=1}^R a_r^{\text{LR}}\lambda_n^r \neq 0, \forall n = 1, \ldots, N_s$. However, it should be noted that distributed implementations of the graph-LR-ARMA filter do not exist. Moreover, the LR-ARMA filter may not be a suitable parametrization of a graph frequency response that contains nonzero (small) values at high graph frequencies.

III. MODEL AND PROBLEM FORMULATION

We now introduce the problem of estimating a random graph signal by observing its noisy nonlinear function, which is determined by the graph topology. In Section III-A, we present the measurement model. In Section III-B, the LMMSE and the sample-LMMSE estimators are derived.

A. Model and Problem Formulation

Consider the problem of recovering a random input graph signal, $\mathbf{x} \in \mathbb{R}^N$, based on the following nonlinear measurement model:

$$\mathbf{y} = \mathbf{g}(\mathbf{L}, \mathbf{x}) + \mathbf{w}, \tag{19}$$

where the measurement function, $\mathbf{g} : \mathbb{R}^{N \times N} \times \mathbb{R}^N \to \mathbb{R}^N$, and the Laplacian matrix, $\mathbf{L}$, which represents the influence of the graph topology, are assumed to be known. The noise, $\mathbf{w}$, has a known probability density function (pdf), $f_{\mathbf{w}}$, with zero mean and covariance matrix $\mathbf{C}_{\mathbf{ww}}$. We assume that the pdf of $\mathbf{x}$, $f_{\mathbf{x}}$, is known, and that $\mathbf{w}$ and $\mathbf{x}$ are independent.

This model comprises a broad family of statistical estimation problems over graphs. For example, a core problem of power system analysis is the recovery of voltages from power measurements [18], where both the voltages and the powers can be considered as graph signals [14], [46], [47] and the Laplacian matrix is the susceptance matrix. This problem is NP-hard and nonconvex [17]; It will be discussed further in Section VI. Similarly, for a water distribution system we are interested in estimating the nodal demands from measurements of the pressure heads and pipe flows using the nonlinear relation between them [48]. Another example is the problem of estimating a graph signal based on noisy observations [49]. This setting comprises a broad family of estimation problems, including group synchronization on graphs, community detection, and low-rank matrix estimation. Finally, this model can be extended to other topology matrices, such as the adjacency matrix and the Markov matrix [39].

We are interested in recovering the input graph signal, $\mathbf{x}$, from the observation vector $\mathbf{y}$ in (19), based on minimizing the MSE. Thus, we seek an estimator of $\mathbf{x}$, as follows:

$$\hat{\mathbf{x}}^{\mathrm{MMSE}} = \underset{\hat{\mathbf{x}}(\mathbf{y}, \mathbf{L}) \in \mathbb{R}^N}{\operatorname{argmin}} \, \mathbb{E}\left[(\hat{\mathbf{x}}(\mathbf{y}, \mathbf{L}) - \mathbf{x})^T (\hat{\mathbf{x}}(\mathbf{y}, \mathbf{L}) - \mathbf{x})\right]. \tag{20}$$

In the general case where the statistics of $\mathbf{x}$ and $\mathbf{y}$ are known, the MMSE estimator, $\hat{\mathbf{x}}^{\mathrm{MMSE}} = \mathbb{E}[\mathbf{x}|\mathbf{y}]$, solves (20). Computing $\hat{\mathbf{x}}^{\mathrm{MMSE}}$ requires an expression for the posterior pdf of $\mathbf{x}$ given $\mathbf{y}$, denoted by $f_{\mathbf{x}|\mathbf{y}}$. For the considered model, the pdf of $\mathbf{y}$ satisfies

$$f_{\mathbf{y}} = f_{\mathbf{g}(\mathbf{L}, \mathbf{x})} * f_{\mathbf{w}}, \tag{21}$$

where $*$ denotes the convolution operator and $f_{\mathbf{g}(\mathbf{L}, \mathbf{x})}$ is the pdf of $\mathbf{g}(\mathbf{L}, \mathbf{x})$. Since $\mathbf{g}(\mathbf{L}, \mathbf{x})$ is a nonlinear function, the pdf of the transformation of $\mathbf{x}$ given by $\mathbf{g}(\mathbf{L}, \mathbf{x})$ does not have a closed-form expression, and therefore, the pdf of $\mathbf{y}$ and the posterior pdf of $\mathbf{x}$ given $\mathbf{y}$ (i.e., $f_{\mathbf{y}}$ and $f_{\mathbf{x}|\mathbf{y}}$) do not have analytical closed-form expressions. As a result, the MMSE estimator does not have a closed-form solution. Moreover, the computational complexity of the numerical evaluation of the MMSE estimator by multidimensional integration is very high, making the evaluation impractical, especially, for large networks, i.e., large N.

B. LMMSE and Sample-LMMSE Estimators

A common sub-optimal approach is to choose to retain the MMSE criterion, but constrain the subset of estimators. The LMMSE estimator is the optimal solution to (20) over the subset of estimators that are linear functions of the measurements, $\mathbf{y}$, and is given by [50]

$$\hat{\mathbf{x}}^{(\mathrm{LMMSE})} = \mathbb{E}[\mathbf{x}] + \mathbf{C}_{\mathbf{xy}} \mathbf{C}_{\mathbf{yy}}^{-1}(\mathbf{y} - \mathbb{E}[\mathbf{y}]), \tag{22}$$

where it is assumed that $\mathbf{C}_{\mathbf{yy}}$ is a non-singular matrix. The LMMSE estimator can be used when the second-order statistics of $\mathbf{x}$ and $\mathbf{y}$ are known. However, in the considered case, the LMMSE estimator is often intractable as well, as an expression for the covariance is not generally known.

Since the pdf of $\mathbf{x}$, the statistics of the noise, the Laplacian matrix $\mathbf{L}$, and the measurement function are all known, the *sample-LMMSE* estimator can be evaluated based on a two-stage procedure ([51], p. 728 in [52]). First, the mean and covariance matrices from (22), $\mathbb{E}[\mathbf{y}]$, $\mathbf{C}_{\mathbf{xy}}$, and $\mathbf{C}_{\mathbf{yy}}$, are estimated by the sample-means and sample-covariance matrices from a training set.[1] Second, these estimators are plugged into the LMMSE estimator in (22) to obtain the sample-LMMSE estimator ([51], p. 728 in [52]).

In the first stage, P random samples of $\mathbf{x}$, $\{\mathbf{x}_p\}_{p=1}^P$, are generated, with a pdf $f_{\mathbf{x}}$. This is the training dataset. The associated output vectors under the generating model in (19) are denoted by $\{\mathbf{y}_p\}_{p=1}^P$. Then, assuming zero-mean noise in (19) and using the function $\mathbf{g}(\cdot, \cdot)$, the sample-mean observation vector from (19) is

$$\hat{\mathbf{y}} = \frac{1}{P} \sum_{p=1}^P \mathbf{g}(\mathbf{L}, \mathbf{x}_p). \tag{23}$$

Similarly, since $\mathbf{x}$ and $\mathbf{w}$ are assumed to be statistically independent, the sample cross-covariance matrix of $\mathbf{x}$ and $\mathbf{y}$, and the sample covariance matrix of $\mathbf{y}$ are computed by

$$\hat{\mathbf{C}}_{\mathbf{xy}} = \frac{1}{P} \sum_{p=1}^P (\mathbf{x}_p - \mathbb{E}[\mathbf{x}])(\mathbf{y}_p - \hat{\mathbf{y}})^T \tag{24}$$

and

$$\hat{\mathbf{C}}_{\mathbf{yy}} = \frac{1}{P} \sum_{p=1}^P (\mathbf{y}_p - \hat{\mathbf{y}})(\mathbf{y}_p - \hat{\mathbf{y}})^T + \mathbf{C}_{\mathbf{ww}}, \tag{25}$$

respectively, where $\mathbf{C}_{\mathbf{ww}}$ is assumed to be known. In the second stage, the sample-LMMSE estimator is obtained by plugging the sample-mean and sample covariance matrices from (23), (24), and (25) into (22), which results in

$$\hat{\mathbf{x}}^{(\mathrm{sample\text{-}LMMSE})} = \mathbb{E}[\mathbf{x}] + \hat{\mathbf{C}}_{\mathbf{xy}} \hat{\mathbf{C}}_{\mathbf{yy}}^{-1}(\mathbf{y} - \hat{\mathbf{y}}). \tag{26}$$

The sample-LMMSE estimator requires the computation of the inverse sample covariance matrix of $\mathbf{y}$, $\hat{\mathbf{C}}_{\mathbf{yy}}$, from (25). Therefore, a drawback of this method is that it requires an extensive dataset for stable estimation of the inverse sample covariance matrix. However, the dataset is usually limited in practical applications since the function $\mathbf{g}(\mathbf{x}, \mathbf{L})$ may change over time due to changes in the topology. Moreover, numerical evaluation of the LMMSE may discard information about the relationship between the graph signal and its underlying graph structure. Thus, it is less robust to changes in the graph topology, and there is no straightforward methodology to update the estimator to new topology, which may happen when vertices or edges are added or removed from the network. Finally, the sample-LMMSE estimator ignores the GSP properties and does not exploit additional information on the graph signal, such as smoothness or graph-bandlimitness, that can improve estimation performance. On the other hand, existing GSP-based approaches have been developed for simple *linear* models (see, e.g. [5], [25], [32]).

In order to reduce the computational complexity of the sample-LMMSE estimator, the sample-version of the diagonal-LMMSE estimator can be used [24]. This estimator minimizes

---

[1] In our model, $\mathbb{E}[\mathbf{x}]$ is known, and thus, is not replaced by its sample mean.

the MSE among linear estimators where the estimation matrix that multiplies $\mathbf{y}$ is restricted to be diagonal. By plugging in the sample mean and sample covariance values from (23)–(25) in the diagonal-LMMSE estimator (see, e.g. (18) in [24]), the sample-diagonal-LMMSE estimator is given by

$$\hat{\mathbf{x}}^{(\text{sample-D-LMMSE})} = \mathbb{E}[\mathbf{x}]$$
$$+ \text{diag}(\text{diag}(\hat{\mathbf{C}}_{\mathbf{xy}}))(\text{diag}(\text{diag}(\hat{\mathbf{C}}_{\mathbf{yy}})))^{-1}(\mathbf{y} - \hat{\mathbf{y}}). \quad (27)$$

The use of diagonal matrices in (27) results in lower computational complexity than that of the sample-LMMSE estimator. However, the performance of (27) is significantly poorer than the other methods presented in this paper (as shown in [24] and was confirmed by our simulations). This motivates us to seek improved solutions that also involve diagonal matrices and, at the same time, use the given graph.

IV. THE GSP-LMMSE ESTIMATOR

In this section, we develop the GSP-LMMSE and the sample-GSP-LMMSE estimators in Section IV-A and discuss their properties in Section IV-B. Conditions under which the proposed GSP-LMMSE estimator is also the LMMSE estimator are developed in Section IV-C.

A. GSP-LMMSE and Sample-GSP-LMMSE Estimators

In this subsection we develop a graph-based linear estimator. We consider estimators of the form

$$\hat{\mathbf{x}} = f(\mathbf{L})\mathbf{y} + \mathbf{b} = \mathbf{V}f(\mathbf{\Lambda})\mathbf{V}^T\mathbf{y} + \mathbf{b}, \quad (28)$$

where $\mathbf{b}$ is a constant vector, $\mathbf{V}$ and $\mathbf{\Lambda}$ are the eigenvector-eigenvalue matrices of the Laplacian matrix, $\mathbf{L}$, as defined in (2), and $f(\cdot)$ is the graph filter defined in (5). The GSP-LMMSE estimator is an estimator which minimizes the MSE in (20) over the subset of GSP estimators in the form of (28). By substituting (28) in (20), the general GSP-LMMSE estimator is defined by

$$\hat{\mathbf{x}}^{(\text{GSP-LMMSE})} = \mathbf{V}\hat{f}(\mathbf{\Lambda})\mathbf{V}^T\mathbf{y} + \hat{\mathbf{b}}, \quad (29)$$

where

$$\{\hat{f}(\mathbf{\Lambda}), \hat{\mathbf{b}}\} = \underset{f(\mathbf{\Lambda}) \in \mathcal{D}_N, \mathbf{b} \in \mathbb{R}^N}{\text{argmin}} \mathbb{E}[||\mathbf{V}f(\mathbf{\Lambda})\mathbf{V}^T\mathbf{y} + \mathbf{b} - \mathbf{x}||^2] \quad (30)$$

and $\mathcal{D}_N$ is the set of diagonal matrices of size $N \times N$. Since $\mathbf{V}$ is a unitary matrix, i.e., $\mathbf{V}\mathbf{V}^T = \mathbf{I}_N$, the minimization problem (30) can be rewritten in the graph frequency domain as follows:

$$\{\hat{f}(\mathbf{\Lambda}), \hat{\mathbf{b}}\} = \underset{f(\mathbf{\Lambda}) \in \mathcal{D}_N, \tilde{\mathbf{b}} \in \mathbb{R}^N}{\text{argmin}} \mathbb{E}\left[||f(\mathbf{\Lambda})\tilde{\mathbf{y}} + \tilde{\mathbf{b}} - \tilde{\mathbf{x}}||^2\right], \quad (31)$$

where $\tilde{\mathbf{y}}$, $\tilde{\mathbf{x}}$, and $\tilde{\mathbf{b}}$ are the GFT of $\mathbf{y}$, $\mathbf{x}$, and $\mathbf{b}$, respectively, as defined in (3).

We first consider the solution for $\tilde{\mathbf{b}}$:

$$\hat{\tilde{\mathbf{b}}} = \underset{\tilde{\mathbf{b}} \in \mathbb{R}^N}{\text{argmin}} \mathbb{E}\left[||f(\mathbf{\Lambda})\tilde{\mathbf{y}} + \tilde{\mathbf{b}} - \tilde{\mathbf{x}}||^2\right]. \quad (32)$$

Since (32) is convex, the optimal solution is obtained by equating the derivative w.r.t. $\tilde{\mathbf{b}}$ to zero, which results in

$$\hat{\tilde{\mathbf{b}}} = \mathbb{E}[\tilde{\mathbf{x}}] - f(\mathbf{\Lambda})\mathbb{E}[\tilde{\mathbf{y}}]. \quad (33)$$

This implies

$$\hat{\mathbf{b}} = \mathbf{V}\hat{\tilde{\mathbf{b}}} = \mathbb{E}[\mathbf{x}] - \mathbf{V}f(\mathbf{\Lambda})\mathbf{V}^T\mathbb{E}[\mathbf{y}]. \quad (34)$$

Algorithm 1: Sample-GSP-LMMSE Estimator.

Input:
- The function $\mathbf{g}(\mathbf{L}, \mathbf{x})$.
- The Laplacian matrix $\mathbf{L}$.
- The distribution of $\mathbf{x}$, $f_{\mathbf{x}}$, and its mean, $\mathbb{E}[\mathbf{x}]$.
- The distribution of $\mathbf{w}$.

Algorithm Steps:
1) Generate P random samples of $\mathbf{x}$, with a pdf $f_{\mathbf{x}}$.
2) Evaluate the sample vectors: $\hat{\mathbf{y}}$, $\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$, and $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ from (23), (39), and (40), respectively.

Output: sample-GSP-LMMSE estimator
$$\hat{\mathbf{x}}^{(\text{sGSP-LMMSE})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}\text{diag}(\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}\mathbf{V}^T(\mathbf{y} - \hat{\mathbf{y}}).$$

Substituting (33) into (31) we obtain that the graph frequency response of the GSP-LMMSE estimator is given by

$$\hat{f}(\mathbf{\Lambda}) = \underset{f(\mathbf{\Lambda}) \in \mathcal{D}_N}{\text{argmin}} \mathbb{E}[||f(\mathbf{\Lambda})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]) - (\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}])||^2]. \quad (35)$$

The solution to (35) is given by

$$\hat{f}(\mathbf{\Lambda}) = \text{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}, \quad (36)$$

where

$$\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} \triangleq \text{diag}(\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}), \quad \mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} \triangleq \text{diag}(\text{diag}(\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}})). \quad (37)$$

We assume that $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is non-singular. The GSP-LMMSE estimator is then given by

$$\hat{\mathbf{x}}^{(\text{GSP-LMMSE})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}\text{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}\mathbf{V}^T(\mathbf{y} - \mathbb{E}[\mathbf{y}]). \quad (38)$$

The GSP-LMMSE estimator from (38) is based on the *diagonal* of the covariance matrices of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ that are defined in (37), which has advantages from a computational point of view, compared with the LMMSE estimator in (22) that uses the full covariance matrices of $\mathbf{x}$ and $\mathbf{y}$. The GSP-LMMSE estimator is in fact the diagonal-LMMSE estimator [24] in the *graph frequency domain*. This estimator minimizes the MSE among the linear estimators of $\tilde{\mathbf{x}}$, where the estimation matrix that multiplies $\tilde{\mathbf{y}}$ is restricted to be a diagonal matrix.

Similar to the sample-LMMSE estimator in Section III-B, we can use the sample-mean version of the GSP-LMMSE estimator when the statistics are not completely specified. The idea is to use P random samples of $\mathbf{x}$, $\mathbf{x}_1, \ldots, \mathbf{x}_p$, that are generated from the model in (19) to compute the sample-mean vector from (23) and the diagonal sample covariance matrices from (37) by

$$\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} = \frac{1}{P}\sum_{p=1}^{P}(\tilde{\mathbf{x}}_p - \mathbb{E}[\tilde{\mathbf{x}}]) \circ \left(\mathbf{V}^T\mathbf{g}(\mathbf{L}, \mathbf{x}_p) - \mathbf{V}^T\hat{\mathbf{y}}\right) \quad (39)$$

and

$$[\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}]_{i,i} = \frac{1}{P}\sum_{p=1}^{P}\left([\mathbf{V}^T\mathbf{g}(\mathbf{L}, \mathbf{x}_p)]_i - [\mathbf{V}^T\bar{\mathbf{y}}]_i\right)^2 + [\mathbf{C}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}}]_{i,i}, \quad (40)$$

respectively, where $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a diagonal matrix, $\tilde{\mathbf{w}}$ is the GFT of $\mathbf{w}$, and $\hat{\mathbf{y}}$ is defined in (23). By substituting (23), (39), and (40) in (38), we obtain the sample-GSP-LMMSE estimator:

$$\hat{\mathbf{x}}^{(\text{sGSP-LMMSE})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}\text{diag}(\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}\mathbf{V}^T(\mathbf{y} - \hat{\mathbf{y}}), \quad (41)$$

where we assume that $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a non-singular matrix. This approach is summarized in Algorithm 1.

B. Advantages and Discussion

The GSP representation affords insight into the frequency contents of *nonlinear* graph signals, such as smoothness and graph-bandlimitness, which can be incorporated into the GSP estimators. In addition, a main advantage of the proposed sample-GSP-LMMSE estimator is that it requires the estimation of only two N-length vectors that contain the diagonal of the covariance and of the cross-covariance matrices, as described in (39) and (40). This is in contrast with the sample-LMMSE estimator from (22) that requires estimating two $N \times N$ matrices, $\hat{\mathbf{C}}_{\mathbf{xy}}$ and $\hat{\mathbf{C}}_{\mathbf{yy}}$ from (24) and (25), respectively. This advantage improves the sample-GSP-LMMSE estimator performance compared to the sample-LMMSE estimator with limited datasets used for the non-parametric estimation of the different sample-mean values. Moreover, the estimation of the inverse sample diagonal covariance matrix, $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ from (40), is more robust to limited data (i.e., smaller P) than the estimation of the inverse sample covariance matrix, $\hat{\mathbf{C}}_{\mathbf{yy}}$ from (25). Since $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a diagonal matrix, it is a non-singular matrix with probability 1 for any $P \geq 2$ (under the assumption that $\mathbf{x}$ is a continuous random variable), while the non-diagonal matrix, $\hat{\mathbf{C}}_{\mathbf{yy}}$, requires P to be much larger to be a non-singular matrix. As a result, in settings where the sample size (P) is comparable to the observation dimension (N), the sample-LMMSE estimator exhibits severe performance degradation [53], [54]. This is because the sample covariance matrix is not well-conditioned in the small sample size regime, and inverting it amplifies the estimation error [55]. In the asymptotic regime where the observation dimension, N, is fixed and $P \to \infty$, the sample mean and sample covariance matrices are consistent estimators. Thus, in the asymptotic case, the sample-LMMSE and the sample-GSP-LMMSE estimators converge to the LMMSE and the GSP-LMMSE estimators.

In terms of computational complexity, the sample-LMMSE estimator from (26) requires: 1) forming the sample mean and the sample covariance matrices $\hat{\mathbf{C}}_{\mathbf{xy}}$ and $\hat{\mathbf{C}}_{\mathbf{yy}}$ from (23), (24), and (25), respectively, with full matrix multiplications and an additional cost of $\mathcal{O}(PN^2)$; 2) computing the inverse of the $N \times N$ matrix $\hat{\mathbf{C}}_{\mathbf{yy}}$, which has a complexity of $\mathcal{O}(N^3)$; and 3) performing full matrix multiplications, with a computational complexity of $\mathcal{O}(N^3)$. The sample-GSP-LMMSE estimator from (41) requires: 1) forming the sample mean in (23) and the *diagonal* sample covariance matrices $\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ and $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ from (39) and (40), respectively, with a cost of $\mathcal{O}(PN)$, where the data is generated in the graph frequency domain; 2) computing the inverse of the diagonal matrix $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$, which has a complexity of $\mathcal{O}(N)$; and 3) performing multiplications of diagonal matrices with a cost of $\mathcal{O}(N)$. The estimator in the vertex domain in (41) requires matrix multiplication by $\mathbf{V}$ and $\mathbf{V}^T$ with a cost of $\mathcal{O}(N^3)$.

The use of the graph frequency domain in the GSP-LMMSE estimator requires the computation of the EVD of the Laplacian matrix, which is of order $\mathcal{O}(N^3)$, and can be computed offline. If the EVD can be assumed to be known then this task may be avoided. Recent works propose low-complexity methods to reduce the complexity of this task (see, e.g. [56]). In addition, the computational complexity of the GSP-LMMSE estimator can be reduced even further by using the properties of the Laplacian matrix, which tends to be sparse, and therefore, matrix operations may require fewer computations.

C. Linear Optimality Conditions

In this subsection, we address the question of under which situations the proposed GSP-LMMSE estimator coincides with the LMMSE estimator and with the graphical Wiener filter.

The following theorem states sufficient and necessary conditions for the GSP-LMMSE and LMMSE estimators to coincide.

Theorem 1: The GSP-LMMSE estimator coincides with the LMMSE estimator if

$$\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1} = \mathrm{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}, \tag{42}$$

where $\tilde{\mathbf{y}}$ and $\tilde{\mathbf{x}}$ are the GFT of $\mathbf{y}$ and $\mathbf{x}$, and where $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ and $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are non-singular matrices.

Proof: By comparing the r.h.s. of (22) and the r.h.s. of (38), it can be verified that the GSP-LMMSE estimator coincides with the LMMSE estimator if

$$\mathbf{C}_{\mathbf{xy}}\mathbf{C}_{\mathbf{yy}}^{-1} = \mathbf{V}\mathrm{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}\mathbf{V}^T. \tag{43}$$

By right and left multiplication of (43) by $\mathbf{V}^T$ and $\mathbf{V}$, respectively, and using $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}_N$, we obtain

$$\mathbf{V}^T\mathbf{C}_{\mathbf{xy}}\mathbf{V}\mathbf{V}^T\mathbf{C}_{\mathbf{yy}}^{-1}\mathbf{V} = \mathrm{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}. \tag{44}$$

Using the GFT definition in (3), the condition in (44) can be rewritten as (42). ∎

The intuition behind the result in Theorem 1 is that if (42) is satisfied, then the LMMSE estimator is also the diagonal LMMSE estimator in the graph frequency domain, i.e., the GSP-LMMSE estimator. By using the definitions in (37), it can be verified that a sufficient (but not necessary) condition for (42) to hold is that $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ and $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are diagonal matrices. In this case, $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} = \mathrm{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})$ and $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$. In the following Theorems we present two special cases for which $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ and $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are diagonal matrices, and thus, (42) holds.

Theorem 2: The GSP-LMMSE estimator coincides with the LMMSE estimator if the following conditions hold:

C.1) The nonlinear measurements function, $\mathbf{g}(\mathbf{L}, \mathbf{x})$, is separable in the graph frequency domain ("orthogonal frequencies"). That is, it satisfies

$$[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_n = [\tilde{\mathbf{g}}(\mathbf{L}, \tilde{x}_n\mathbf{v}_n)]_n, \, n = 1, \ldots, N, \tag{45}$$

where $\tilde{x}_n$ is the nth element of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{g}} = \mathbf{V}^T\mathbf{g}(\mathbf{L}, \mathbf{x})$.

C.2) The elements of the input graph signal, $\mathbf{x}$, are statistically independent in the graph frequency domain.

C.3) The noise vector, $\mathbf{w}$, is uncorrelated in the graph frequency domain, i.e., $\mathbf{C}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}}$ is a diagonal matrix.

Proof: The proof is given in Appendix A. ∎

Theorem 3: The GSP-LMMSE estimator coincides with the LMMSE estimator if Condition C.3 holds and in addition

C.4) The measurement function, $\mathbf{g}(\mathbf{L}, \mathbf{x})$, is the output of a linear graph filter as defined in (5), i.e.,

$$\mathbf{g}(\mathbf{L}, \mathbf{x}) = \mathbf{V}f(\mathbf{\Lambda})\mathbf{V}^T\mathbf{x}. \tag{46}$$

C.5) The covariance matrix of the input graph signal, $\mathbf{C}_{\mathbf{xx}}$, is diagonalizable by the eigenvector matrix of the Laplacian, $\mathbf{V}$, i.e., $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$ is a diagonal matrix.

Proof: The proof is given in Appendix B. ∎

The special case in Theorem 3 fits the model behind the development of the graphical Wiener filter [32]: First, Conditions C.3 and C.4 imply the linear model assumed in developing the graphical Wiener filter in Section V-A in [32]. Second, under Condition C.5, the signal $\mathbf{x} - \mathbb{E}[\mathbf{x}]$ is a Graph Wide-Sense Stationary (GWSS) signal (see Definition 3 and Theorem 1 in [32]), which is the requirement for the graphical Wiener filter. Therefore, under the conditions of Theorem 3 the

GSP-LMMSE estimator coincides with the graphical Wiener filter (Eq. (13) in [32]), which is also the LMMSE estimator in this case. Therefore, the proposed GSP-LMMSE estimator can be interpreted as the graphical Wiener filter, but without assuming a linear model or GWSS signals.

Conditions C.2 and C.5 are common assumptions in the GSP framework [57], [58]. However, it should be noted that even if the elements of the input graph signal, $\mathbf{x}$, are statistically independent, they are not necessarily statistically independent in the graph frequency domain, as required in Condition C.2, nor even uncorrelated. In addition, Condition C.1 is satisfied if the function $\mathbf{g}(\mathbf{L}, \mathbf{x})$ is diagonalized by the eigenvector matrix $\mathbf{V}$ of $\mathbf{L}$. Condition C.4, in which the measurements are obtained as a synthetic output of a graph linear filter as described in (46), is a sufficient condition for Condition C.1. Finally, since Condition C.1 is less restrictive then Condition C.4, Condition C.2 (independency) is more restrictive than Condition C.5 (decorrelation).

V. GSP ESTIMATORS BY PARAMETRIC GRAPH FILTERS

In the previous section, we presented a general approach to finding the optimal GSP-LMMSE estimator. However, the GSP-LMMSE estimator is a function of the specific graph structure with fixed dimensions, and thus: 1) it is not optimal when the topology is changed; 2) it is not adaptive to changes in the number of vertices, N, since when vertices are added or removed, the dimension of the Laplacian matrix and of the graph signal are changed and the GSP-LMMSE in (38) is not a valid estimator; and 3) there is no straightforward way to incorporate other graph frequency constraints. Thus, the optimal graph frequency response needs to be redeveloped for any small change in the topology.

In this section, we formulate the problem of designing a graph filter that is MSE-optimal with a specific parametric representation. Then, we develop three implementations of this design by using the three specific graph filters from Section II: a linear pseudo-inverse graph filter (Section V-B), an ARMA graph filter (Section V-C), and a low-rank ARMA graph filter (Section V-D). This universal design of graph filters by fitting the frequency response over a continuous range of graph frequencies based on the graph frequency response is adaptive to topology changes, e.g. in cases when the number of vertices and/or edges is changed. In addition, the parametric representation provides a straightforward way to integrate GSP properties. Finally, since in practice the desired graph frequency response is often approximated by its sample-mean version based on a training dataset, parametrizations can reduce outlier errors and noise effects.

The choice between the different graph filters used in this section can be done according to the trade-off between approximation accuracy, convergence rate, and computational complexity, as detailed in recent literature [5], [29]. In particular, the parametrization by the linear pseudo-inverse graph filter requires, in general, a higher filter order than the ARMA graph filter, i.e., $K > Q, R$, in order to obtain a good approximation, which may lead to stability problems in large networks. However, finding the ARMA graph filter parameters is based on a nonconvex optimization, while the parameters of the linear pseudo-inverse graph filter have a closed-form solution. Using the ARMA filter is known to improve the approximation accuracy and reduce the number of required filter coefficients compared with those of the linear ("finite impulse response") filter [29]. The LR-ARMA GSP estimator uses only part of the

EVD of $\mathbf{L}$ and requires a lower filter order compared with the ARMA graph filter. However, its performance may be significantly degraded compared to other filters for signals that are not low-frequency graph signals.

A. General Design of the Graph Frequency Response

The graph frequency response of the sample-GSP-LMMSE estimator from (36) is defined only at graph frequencies λ_n, $n = 1, \ldots, N$. In this subsection, our goal is to develop a graph-based linear estimator of the form of (28) that minimizes (20), but where $f(\boldsymbol{\Lambda})$ is restricted to specific parametrizations as a linear and shift-invariant graph filter. To this end, the MSE-optimal parameter vector, $\boldsymbol{\alpha}$, for any graph-filter parametrizations, $h(\boldsymbol{\Lambda}; \boldsymbol{\alpha})$, is found by solving (35) after substituting the specific parametrizations $f(\boldsymbol{\Lambda}) = h(\boldsymbol{\Lambda}; \boldsymbol{\alpha})$:

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha} \in \Omega_{\boldsymbol{\alpha}}}{\arg\min} \, \mathbb{E}\left[||h(\boldsymbol{\Lambda}; \boldsymbol{\alpha})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]) - (\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}])||^2 \right], \quad (47)$$

where $\Omega_{\boldsymbol{\alpha}}$ is the relevant parameter space, which is defined by the specific choice of graph filter. This parametric representation of the graph filters interpolates the graph frequency response of the GSP-LMMSE estimator to any graph frequency. Therefore, when the topology is changed, e.g. by adding/removing vertices, we can substitute the new eigenvalue matrix, $\boldsymbol{\Lambda}$, in the graph filter to obtain the approximation, without generating new training data.

The following theorem states the relation between the frequency response of the GSP-LMMSE estimator from (36) and the optimal parameter vector of a specific parametrization.

Theorem 4: The problem in (47) is equivalent to the following problem:

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha} \in \Omega_{\boldsymbol{\alpha}}}{\arg\min} \, ||\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{\frac{1}{2}}(\operatorname{diag}(h(\boldsymbol{\Lambda}; \boldsymbol{\alpha})) - \operatorname{diag}(\hat{f}(\boldsymbol{\Lambda})))||^2, \quad (48)$$

where $\hat{f}(\boldsymbol{\Lambda})$ and $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are given in (36) and (37), respectively.

Proof: The proof is given in Appendix C. ∎

From Theorem 4, the minimization of the MSE for a specific linear and shift-invariant graph filter in (47) is equivalent to the minimization of the WLS distance between the desired graph frequency response of the GSP-LMMSE estimator and the graph frequency responses of the chosen graph filter. Using the solution of (48), the h-GSP-LMMSE estimator with a general parameterized filter $h(\cdot)$ is given by

$$\hat{\mathbf{x}}^{(\text{h-GSP})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}h(\boldsymbol{\Lambda}; \hat{\boldsymbol{\alpha}})\mathbf{V}^T(\mathbf{y} - \hat{\mathbf{y}}). \quad (49)$$

Similar to the implementation of the sample-LMMSE and the sample-GSP-LMMSE estimators, using the sample-mean version of $\hat{f}(\boldsymbol{\Lambda})$, obtained by substituting (39) and (40) in (36), we obtain that the sample-mean version of the optimal graph filter parameters from (48):

$$\hat{\boldsymbol{\alpha}}^{\text{sample}} = \underset{\boldsymbol{\alpha} \in \Omega_{\boldsymbol{\alpha}}}{\arg\min} \, ||\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{\frac{1}{2}} \left(\operatorname{diag}(h(\boldsymbol{\Lambda}; \boldsymbol{\alpha})) - \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} \right) ||^2. \quad (50)$$

Then, by substituting $h(\boldsymbol{\Lambda}; \hat{\boldsymbol{\alpha}}) = h(\boldsymbol{\Lambda}; \hat{\boldsymbol{\alpha}}^{\text{sample}})$ in (49), the sample-mean version of the h-GSP-LMMSE estimator with a general parameterized filter, $h(\cdot)$, is given by

$$\hat{\mathbf{x}}^{(\text{sh-GSP})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}h(\boldsymbol{\Lambda}; \hat{\boldsymbol{\alpha}}^{\text{sample}})\mathbf{V}^T(\mathbf{y} - \hat{\mathbf{y}}). \quad (51)$$

The sample-h-GSP estimator for a general graph filter is summarized in Algorithm 2.

In the following subsections, we present three different GSP estimators for different choices of typical graph filters $h(\cdot; \boldsymbol{\alpha})$

Algorithm 2: Sample-h-GSP Estimator.

Input:
- The function $\mathbf{g}(\mathbf{L}, \mathbf{x})$.
- The Laplacian matrix $\mathbf{L}$.
- The distribution of $\mathbf{x}$, $f_{\mathbf{x}}$, and its mean, $\mathbb{E}[\mathbf{x}]$.
- The distribution of $\mathbf{w}$.
- Parameterized filter $h(\cdot)$.

Algorithm Steps:
1) Generate P random samples of $\mathbf{x}$, with a pdf $f_{\mathbf{x}}$.
2) Evaluate the sample vectors: $\hat{\mathbf{y}}$, $\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$, and $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ from (23), (39), and (40), respectively.
3) Compute the optimal graph filter coefficient vector, $\hat{\boldsymbol{\alpha}}^{\text{sample}}$ by solving (50).

Output: sample-h-GSP estimator
$\hat{\mathbf{x}}^{(\text{sh-GSP})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}h(\boldsymbol{\Lambda}; \hat{\boldsymbol{\alpha}}^{\text{sample}})\mathbf{V}^T(\mathbf{y} - \hat{\mathbf{y}})$.

---

and we evaluate their associated optimal parameters. The difference between the estimators in Sections V-B- V-D is that they are based on different parametrizations and implementations. The minimization in (48) implies that one should choose a graph-filter parametrization, $h(\boldsymbol{\Lambda}; \boldsymbol{\alpha})$, that will be close to the desired graph frequency response, $\hat{f}(\boldsymbol{\Lambda})$, from (36). Therefore, the shape of the desired graph frequency response curve affects the suitable choice of the graph filter parameterization, which is application-dependent. The study of $\hat{f}(\boldsymbol{\Lambda})$ is expected to lead to a better understanding of what kinds of filters are useful. This study can be done, for example, based on theoretical properties such as low-pass and high-pass graph filters [59] and on a simulation study. For example, in the simulations in Section VI, $\hat{f}(\boldsymbol{\Lambda})$ behaves as a low-pass graph filter, and thus, we choose graph filters suitable for this case. The presented framework can be easily extended to other graph filters for different shapes of $\hat{f}(\boldsymbol{\Lambda})$.

B. Filter 1: Sample Linear Pseudo-Inverse GSP Estimator

For the linear pseudo-inverse graph filter from Section II-B, the graph filter from (6) satisfies

$$\text{diag}(h^{\text{LPI}}(\boldsymbol{\Lambda}^{\dagger}; \boldsymbol{\alpha}^{\text{LPI}})) = \bar{\boldsymbol{\Gamma}}_K \boldsymbol{\alpha}^{\text{LPI}}, \tag{52}$$

where $\boldsymbol{\alpha}^{\text{LPI}}$ is the filter coefficients vector from (7) and $\bar{\boldsymbol{\Gamma}}_K$ is a $N \times (K+1)$ matrix with elements

$$[\bar{\boldsymbol{\Gamma}}_K]_{i,j} = \begin{cases} \lambda_i^{-(j-1)}, & \text{for } 2 \leq i \leq N, \, j = 1, \ldots, K+1 \\ 1, & \text{for } i = j = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{53}$$

Therefore, the optimal filter coefficients, $\boldsymbol{\alpha}^{\text{LPI}}$, are obtained by substituting (52) in (50) and removing a constant term, which results in

$$\hat{\boldsymbol{\alpha}}^{\text{LPI}} = \underset{\boldsymbol{\alpha} \in \mathbb{R}^{K+1}}{\arg\min} \, \boldsymbol{\alpha}^T \bar{\boldsymbol{\Gamma}}_K^T \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} \bar{\boldsymbol{\Gamma}}_K \boldsymbol{\alpha} - 2\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}^T \bar{\boldsymbol{\Gamma}}_K \boldsymbol{\alpha}. \tag{54}$$

To avoid overfitting, we replace (54) by the following regularized minimization:

$$\hat{\boldsymbol{\alpha}}^{\text{LPI}} = \underset{\boldsymbol{\alpha} \in \mathbb{R}^{K+1}}{\arg\min} \, \boldsymbol{\alpha}^T \bar{\boldsymbol{\Gamma}}_K^T \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} \bar{\boldsymbol{\Gamma}}_K \boldsymbol{\alpha} - 2\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}^T \bar{\boldsymbol{\Gamma}}_K \boldsymbol{\alpha}$$
$$+ \mu \boldsymbol{\alpha}^T \mathbf{M}_{\text{LPI}} \boldsymbol{\alpha}, \tag{55}$$

where $\mu \geq 0$ is a regularization coefficient and $\mathbf{M}_{\text{LPI}}$ is a positive semidefinite regularization matrix. By equating the gradient of

(55) w.r.t. $\boldsymbol{\alpha}$ to zero, we obtain

$$\hat{\boldsymbol{\alpha}}^{\text{LPI}} = \left(\bar{\boldsymbol{\Gamma}}_K^T \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} \bar{\boldsymbol{\Gamma}}_K + \mu \mathbf{M}_{\text{LPI}} \right)^{-1} \bar{\boldsymbol{\Gamma}}_K^T \hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}. \tag{56}$$

In practice, in order to avoid the numerically unstable problem of inverting the $(K+1) \times (K+1)$ matrix, one can solve the convex optimization problem in (55) by using any existing quadratic programming algorithm. The sample-h-GSP estimator with the linear pseudo-inverse graph filter is implemented by Algorithm 2, where Step 3 is obtained by evaluating the matrix $\bar{\boldsymbol{\Gamma}}$ from (53) and then computing $\hat{\boldsymbol{\alpha}}^{\text{LPI}}$ either by (56) or by using a quadratic programming algorithm to solve (55).

Since the regularization matrix, $\mathbf{M}_{\text{LPI}}$, is a positive semidefinite matrix, the optimization problem in (55) is a convex optimization problem and $\hat{\boldsymbol{\alpha}}^{\text{LPI}}$ is its unique solution, for any $\mu \geq 0$ as long as the matrix $\bar{\boldsymbol{\Gamma}}_K^T \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} \bar{\boldsymbol{\Gamma}}_K$ is a positive definite matrix. This condition holds if $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a positive definite matrix and $\text{rank}(\bar{\boldsymbol{\Gamma}}_K) = K+1$ (see e.g. Chapter 7 in [60]). It is assumed in this paper that $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ from (37) is non-singular (and thus, positive definite). Thus, by taking enough off-line measurements, $\mathbf{x}_1, \ldots, \mathbf{x}_P$, the sample covariance matrix $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a positive definite matrix as well. The following claim states a condition for $\bar{\boldsymbol{\Gamma}}_K$ from (53) to be full (column) rank.

Claim 1: If there are $K+1$ distinct eigenvalues of the matrix $\mathbf{L}$ such that $\lambda_n \neq \lambda_k, \forall n \neq k$, then, $\text{rank}(\bar{\boldsymbol{\Gamma}}_K) = K+1$.

Proof: See Appendix D. ∎

C. Filter 2: Sample ARMA GSP Estimator

For the ARMA graph filter from Section II-C, the graph filter (9) satisfies

$$\text{diag}(h^{\text{ARMA}}(\boldsymbol{\Lambda}; \boldsymbol{\alpha}^{\text{ARMA}}))$$
$$= (\text{diag}(\boldsymbol{\Phi}(N, R)\mathbf{a}))^{-1} \boldsymbol{\Phi}(N, Q)\mathbf{c}, \tag{57}$$

where $\boldsymbol{\alpha}^{\text{ARMA}}$ is the filter coefficients vector from (10), $\boldsymbol{\Phi}(N, O)$ is a $N \times (O+1)$ Vandermonde matrix defined by

$$\boldsymbol{\Phi}(N, O) \triangleq \begin{bmatrix} 1 & \lambda_1 & \ldots & \lambda_1^O \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_N & \ldots & \lambda_N^O \end{bmatrix}. \tag{58}$$

The associated filter coefficients, $\boldsymbol{\alpha}^{\text{ARMA}} = [\mathbf{a}^T, \mathbf{c}^T]^T$, are obtained by substituting (57) in (50) and removing a constant term, which results in

$$(\hat{\mathbf{a}}, \hat{\mathbf{c}}) = \underset{\substack{\mathbf{a} \in \mathbb{R}^{R+1} \\ \mathbf{c} \in \mathbb{R}^{Q+1}}}{\arg\min} \Big\{ \mathbf{c}^T \boldsymbol{\Phi}^T(N, Q) \, (\text{diag}(\boldsymbol{\Phi}(N, R)\mathbf{a}))^{-1}$$
$$\times \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} \, (\text{diag}(\boldsymbol{\Phi}(N, R)\mathbf{a}))^{-1} \boldsymbol{\Phi}(N, Q)\mathbf{c}$$
$$- 2\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}^T \, (\text{diag}(\boldsymbol{\Phi}(N, R)\mathbf{a}))^{-1} \boldsymbol{\Phi}(N, Q)\mathbf{c}$$
$$+ \mu \mathbf{a}^T \mathbf{M_a}\mathbf{a} + \mu \mathbf{c}^T \mathbf{M_c}\mathbf{c} \Big\}, \tag{59}$$

where $a_0 = 1$ and the last two terms are regularization terms that have been added to avoid overfitting, in which $\mu \geq 0$ is a regularization coefficient, and $\mathbf{M_a}$ and $\mathbf{M_c}$ are positive semidefinite regularization matrices. Equating the derivative of (59) w.r.t. $\mathbf{c}$

to zero, results in

$$\hat{\mathbf{c}}^{\text{ARMA}}(\mathbf{a}) = (\mathbf{\Phi}^T(N, Q)(\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}$$
$$\times \hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}(\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}$$
$$\times \mathbf{\Phi}(N, Q) + \mu \mathbf{M}_{\mathbf{c}})^{-1}\mathbf{\Phi}^T(N, Q)$$
$$\times (\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}. \quad (60)$$

By substituting (60) in the objective function from (59), we obtain

$$\hat{\mathbf{a}}^{\text{ARMA}} = \underset{\mathbf{a}\in\mathbb{R}^{R+1}}{\text{argmax}} \left\{ \hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}^T(\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1} \right.$$
$$\left. \times \mathbf{\Phi}(N, Q)\hat{\mathbf{c}}^{\text{ARMA}}(\mathbf{a}) - \mu \mathbf{a}^T\mathbf{M}_{\mathbf{a}}\mathbf{a} \right\}, \quad (61)$$

where $a_0 = 1$. The optimal $\mathbf{c}$ is given by substituting the solution of (61) in (60), i.e., $\hat{\mathbf{c}}^{\text{ARMA}} = \mathbf{c}(\hat{\mathbf{a}}^{\text{ARMA}})$. Finally, the sample-GSP estimator with the ARMA graph filter is implemented by Algorithm 2, where Step 3 is obtained by: I. evaluating the matrices $\mathbf{\Phi}(N, R)$ and $\mathbf{\Phi}(N, Q)$ from (58); II. computing $\hat{\mathbf{a}}$ by solving (61) numerically; and III. Computing $\hat{\mathbf{c}}$ by substituting the result of II in (60). In the simulations we used the Matlab function 'fminsearch' to approximate (61).

Since the regularization matrix, $\mathbf{M}_{\mathbf{c}}$, is a positive semidefinite matrix, (59) is a convex optimization problem w.r.t. $\mathbf{c}$ for any $\mu \geq 0$ as long as $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is positive definite and $\text{rank}((\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}\mathbf{\Phi}(N, Q)) = Q + 1$. The following claim states the condition for this matrix to be full rank.

Claim 2: If there are $Q + 1$ distinct eigenvalues of the matrix $\mathbf{L}$ and $[\mathbf{\Phi}(N, R)\mathbf{a}]_n \neq 0$, $\forall n = 1, \dots, N$, then, $\text{rank}((\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}\mathbf{\Phi}(N, Q)) = Q + 1$.

Proof: See Appendix D. ∎

Therefore, if $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a positive definite matrix and the condition in Claim 2 holds, then the objective function in (59) is a convex optimization problem w.r.t. $\mathbf{c}$ and $\hat{\mathbf{c}}^{\text{ARMA}}(\mathbf{a})$ is its unique solution. It should be noted that the ARMA graph filter involves Vandermonde matrices; in order to obtain a stable solution, R and Q should be chosen to have small values [29].

As explained after (12), the linear graph filter is a special case of the ARMA graph filter. Thus, the optimal coefficients of the linear graph filter are obtained by substituting $\mathbf{a} = a_0 = 1$, $R = 0$, $Q = K$, and $\mu = 0$ in (60), which results in

$$\hat{\boldsymbol{\alpha}}^{\text{lin}} = \hat{\mathbf{c}}^{\text{ARMA}}(\mathbf{a} = 1)$$
$$= \left(\mathbf{\Phi}(N, K)^T\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}\mathbf{\Phi}(N, K) \right)^{-1}\mathbf{\Phi}(N, K)^T\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}, \quad (62)$$

where $\mathbf{\Phi}(\cdot, \cdot)$ is the Vandermonde matrix defined in (58). While all filter designs have an equivalent polynomial filter, the matrix $\mathbf{\Phi}(N, K)$ needs to be well-conditioned in order to obtain good estimation by using the filter in (12) with the coefficients in (62). This will only be the case for small graph sizes N and/or small filter orders K [28]–[30], which leads to limited accuracy of the linear graph filter.

D. Filter 3: Sample Low-Rank ARMA GSP Estimator

For the LR-ARMA graph filter from Section II-D, the graph filter from (16) satisfies

$$h^{\text{LR-ARMA}}(\bar{\mathbf{\Lambda}}; \boldsymbol{\alpha}^{\text{LR}})$$
$$= \left[((\text{diag}(\mathbf{\Phi}(N_s, R)\mathbf{a}^{\text{LR}}))^{-1}\mathbf{\Phi}(N_s, Q)\mathbf{c}^{\text{LR}})^T, \mathbf{0}_{N-N_s}^T \right]^T, \quad (63)$$

where $\boldsymbol{\alpha}^{\text{LR}} = [\mathbf{a}^{\text{LR}^T}, \mathbf{c}^{\text{LR}^T}]^T$ is the filter coefficients vector from (17), and $\mathbf{\Phi}(\cdot, \cdot)$ is the Vandermonde matrix defined in (58). Let $\mathcal{U} = \{1, \dots, N_s\}$, $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}_{\mathcal{U}}}$ denotes the matrix that includes the first N_s rows and columns of $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ and $\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}_{\mathcal{U}}}$ denotes the vector that includes the first N_s elements of $\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$. Then, similarly to (60)-(61), the coefficient vector, $\boldsymbol{\alpha}^{\text{LR}}$, is obtained by minimizing (50) with a regularization term after the substitution of (63), where $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a diagonal matrix and the last $N - N_s$ entries of (63) are zero, which results in

$$\hat{\mathbf{c}}^{\text{LR}}(\mathbf{a}) = \left(\mathbf{\Phi}^T(N_s, Q)(\text{diag}(\mathbf{\Phi}(N_s, R)\mathbf{a}))^{-1}\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}_{\mathcal{U}}} \right.$$
$$\times (\text{diag}(\mathbf{\Phi}(N_s, R)\mathbf{a}))^{-1}\mathbf{\Phi}(N_s, Q) + \mu \mathbf{M}_{\mathbf{c}^{\text{LR}}} \bigg)^{-1}$$
$$\times \mathbf{\Phi}^T(N_s, Q)(\text{diag}(\mathbf{\Phi}(N_s, R)\mathbf{a}))^{-1}\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}_{\mathcal{U}}}, \quad (64)$$

and

$$\hat{\mathbf{a}}^{\text{LR}} = \underset{\mathbf{a}\in\mathbb{R}^{R+1}}{\text{argmax}} \left\{ \hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}_{\mathcal{U}}}^T(\text{diag}(\mathbf{\Phi}(N_s, R)\mathbf{a}))^{-1} \right.$$
$$\left. \times \mathbf{\Phi}(N_s, Q)\hat{\mathbf{c}}^{\text{LR}}(\mathbf{a}) - \mu \mathbf{a}^T\mathbf{M}_{\mathbf{a}^{\text{LR}}}\mathbf{a} \right\}, \quad (65)$$

where $a_0 = 1$ and $\mu \geq 0$, $\mathbf{M}_{\mathbf{a}^{\text{LR}}}, \mathbf{M}_{\mathbf{c}^{\text{LR}}} 0$ are regularization coefficient and positive semidefinite matrices. Then, the optimal $\mathbf{c}$ is given by substituting the solution of (65) in (64), i.e., $\hat{\mathbf{c}}^{\text{LR}} = \mathbf{c}(\hat{\mathbf{a}}^{\text{LR}})$. Similar to Claim 2, the conditions for convexity w.r.t. $\mathbf{c}$ can be derived. Finally, the sample-GSP estimator with the LR-ARMA graph filter is implemented by Algorithm 2, wherein Step 2 evaluates the sample subvectors, $\hat{\mathbf{y}}_{\mathcal{U}}$, $\hat{\mathbf{d}}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}_{\mathcal{U}}}$, and $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}_{\mathcal{U}}}$ from (23), (39), and (40), respectively; and Step 3 is obtained by evaluating the matrices $\mathbf{\Phi}(N_s, R)$ and $\mathbf{\Phi}(N_s, Q)$ from (58) and computing $\hat{\mathbf{c}}^{\text{LR}}(\mathbf{a})$ and $\hat{\mathbf{a}}^{\text{LR}}$ by solving (64) and (65), respectively.

E. Advantages and Discussion

An important advantage of the sample linear pseudo-inverse GSP estimator is that its parameters have a closed-form analytic expression in (56). In contrast, the evaluation of the sample ARMA and the LR-ARMA GSP estimators requires solving nonconvex optimization problems (61) and (65), respectively. On the other hand, since the graph frequency response of the linear pseudo-inverse graph filter from (8) has a discontinuity at $\lambda = 0$, it may be unstable when λ_2 approaches 0 due to topology changes. Since λ_2 describes the graph connectivity [39], this is only a problem when the network becomes disconnected. The relation between the spectrum of the Laplacian matrix of graphs and the graph is extensively discussed in the literature (see, e.g. in [61]). For example, suppose the desired graph frequency response, $\hat{f}(\mathbf{\Lambda})$, from (36) and the graph-filter parametrization, $h(\mathbf{\Lambda}; \boldsymbol{\alpha})$, satisfy some smoothness assumptions. In this case, it can be shown by using results from [33], [62]–[64] that the difference between the optimal graph frequency response *after the change* and the graph-filter parametrization, $h(\mathbf{\Lambda}; \boldsymbol{\alpha})$, that is

computed with the new $\mathbf{\Lambda}$, is bounded when the topology change is bounded.

In terms of computational complexity, the sample-h-GSP from (51) computes the same expressions as the sample-GSP-LMMSE estimator from (41), and thus, has the same computational complexity as described in Section IV-B, with an additional complexity that stems from: a) performing the matrix multiplications of $\mathbf{V}h(\mathbf{\Lambda}; \hat{\boldsymbol{\alpha}}^{\text{sample}})\mathbf{V}^T$, with a computational complexity of $\mathcal{O}(N^3)$; and b) the implementation of the specific graph filter. In detail, implementing the sample linear pseudo-inverse GSP estimator from Section V-B requires: 1) evaluating the optimal filter coefficients $\hat{\boldsymbol{\alpha}}^{\text{LPI}}$ from (56) by computing the inverse of a $K + 1 \times K + 1$ matrix, with complexity $\mathcal{O}(K^3)$, where $K \ll N$; and 2) computing the graph frequency response from (52) with a matrix-vector multiplication of $\mathcal{O}(NK)$. Implementing the sample ARMA and LR-ARMA GSP estimators from Sections V-C and V-D, respectively, requires: 1) evaluating the optimal filter coefficients by solving the nonconvex optimization problem from (61) and (65), respectively, which has a complexity that depends on the chosen optimization algorithm; and 2) computing the graph frequency response from (57) and (63), respectively, with a matrix-vector multiplication and computing the inverse of the diagonal matrix, with a cost of $\mathcal{O}(N(K + 1))$. Finally, all GSP estimators require the EVD of the Laplacian matrix for computing $\mathbf{V}$. This typically requires a $\mathcal{O}(N^3)$ complexity cost, but several fast computation methods for spectral decompositions [65], [66] can be used.

The main advantage of the sample-h-GSP estimators is their low computational complexity needed for updating the estimators when the topology changes. In this case, the updated estimator is evaluated by using the graph filter coefficients, $\hat{\boldsymbol{\alpha}}^{\text{sample}}$, that were evaluated based on the original topology, with the Laplacian matrix of the new topology. As a result, the updated sample-h-GSP estimator from (51) only requires reevaluation of $h(\mathbf{\Lambda}; \hat{\boldsymbol{\alpha}}^{\text{sample}})$, where the graph filter coefficients, $\hat{\boldsymbol{\alpha}}^{\text{sample}}$, are known, which has a maximum complexity of $\mathcal{O}(N^3)$. This is in contrast with the reevaluation needed for the sample-LMMSE and the sample-GSP-LMMSE estimators. In addition, the low complexity and distributed implementation of the sample ARMA GSP estimator is described in [29], [44].

VI. SIMULATION

In this section we evaluate the performance of the proposed GSP-LMMSE estimator from Section IV and the three parametrizations from Sections V-B-V-D for solving the problem of power system state estimation (PSSE), which is essential for various monitoring purposes [18]. The setting of the PSSE problem is presented in Section VI-A. The different estimation methods that are presented in this section are described in Section VI-B. The results for stationary networks and for networks with topology changes are presented in Sections VI-C and VI-D, respectively.

A. Case Study: PSSE in Electrical Networks

A power system can be represented as an undirected weighted graph, $\mathcal{G}(\mathcal{V}, \xi)$, where the set of vertices, $\mathcal{V}$, is the set of buses (generators or loads) and the edge set, ξ, is the set of transmission lines between these buses. The measurement vector of the active powers at the buses, $\mathbf{y}$, can be described by the model in (19),

with nonlinear measurement function

$$[\mathbf{g}(\mathbf{L}, \mathbf{x})]_n \triangleq \sum_{m=1}^{N} |v_n||v_m|(G_{n,m} \cos(x_n - x_m)$$
$$+ B_{n,m} \sin(x_n - x_m)), \qquad (66)$$

$n = 1, \ldots, N$. Here x_n and $|v_n|$ are the voltage phase and amplitude at the nth bus, and $G_{n,m}$ and $B_{n,m}$ are the conductance and susceptance of the transmission line between the buses n and m [18], where $(n, m) \in \xi$. In the graph modeling of the electrical network, the Laplacian matrix, $\mathbf{L}$, is constructed by using $B_{n,m}$, $n, m = 1, \ldots, N$ (see Section II-C in [15]). We assume that $|v_n| = 1$, which is a common assumption [18], and $G_{n,m}$ and $B_{n,m}$ are all known.

The goal of PSSE is to recover the state vector, $\mathbf{x}$, from the power measurements of $\mathbf{g}(\mathbf{L}, \mathbf{x})$, which is known to be a NP-hard problem [17]. The input graph signal, $\mathbf{x}$, is shown to be smooth [14], [47], i.e., its graph smoothness [3], [4], is small. Therefore, we model the distribution of the input graph signal, $\mathbf{x}$, in the graph frequency domain [57], [58], as a smooth Gaussian distribution, as follows:

$$\tilde{\mathbf{x}}_{2:\,\text{end}} \sim \mathcal{N}(\mathbf{0}, \beta \mathbf{\Lambda}_{2:\,\text{end},2:\,\text{end}}^{-1}), \qquad (67)$$

where β is a smoothness level. The smooth distribution of $\tilde{\mathbf{x}}$ from (67) implies in particular that the first graph frequency of $\mathbf{x}$ satisfies $\tilde{x}_1 = 0$. Finally, we assume that the noise term, $\mathbf{w}$, from the model in (19) in this case is zero-mean Gaussian with covariance matrix $\mathbf{C_{ww}} = \sigma^2 \mathbf{I}_N$. It can be seen that for this example, Conditions C.2, C.3, and C.5 from Theorems 2 and 3 are satisfied, but Conditions C.1 and C.4 are not satisfied. It can also be shown that (43) is not satisfied. Therefore, the proposed approach is neither the LMMSE estimator nor the graphical Wiener filter [32] in this case. The values of the different physical parameters in (66) are taken from the test case of a 118-bus IEEE power system [67], where $N = 118$. The MSE of the different estimators is calculated by performing 10,000 Monte Carlo simulations.

B. Methods

In the simulations we compare the performance of the following estimators:

1) The sample-LMMSE estimator from (26).

2) The sample-LMMSE estimator from (26) with a large P ($P = 5 \cdot 10^6 \gg N$), denoted as P_∞-LMMSE. In this asymptotic regime, the sample-LMMSE estimator converges to the LMMSE estimator. Since the MSE of the LMMSE estimator is lower than the MSE of any linear estimator, it can be used as a benchmark for a stationary network.

3) The sample-GSP-LMMSE estimator from Algorithm 1.

4) The sample linear pseudo-inverse GSP estimator from Algorithm 2 with $K = 6$, where Step 3 is implemented as explained after (56), and the regularization matrix is set to $\mathbf{M}_{\text{LPI}} = \text{diag}(\lambda_N^0, \ldots, \lambda_N^K)$ in order to restrict the length of the filter and of the power of the pseudo-inverse of the Laplacian.

5) The sample ARMA GSP estimator, which is implemented as explained after (61), with $R, Q = 3$, $\mathbf{M_a} = \mathbf{I}_R$, $\mathbf{M_c} = \mathbf{I}_Q$.

6) The sample LR-ARMA GSP estimator, which is implemented as explained after (65), with $R_{\text{LR}}, Q_{\text{LR}} = 2$, $\mathbf{M}_{\mathbf{a}^{\text{LR}}} = \mathbf{I}_{R_{\text{LR}}}, \mathbf{M}_{\mathbf{c}^{\text{LR}}} = \mathbf{I}_{Q_{\text{LR}}}$, and the cutoff frequency $N_s = 0.3\,N$. Thus, it uses only the 30% smallest eigenvalues and their associated eigenvectors of the Laplacian matrix, $\mathbf{L}$.
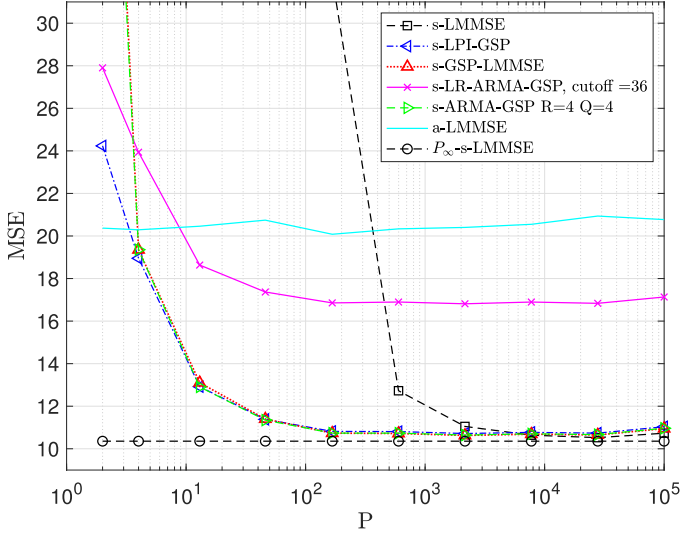
Fig. 1. The MSE versus P for the different estimators, where $\sigma^2 = 0.05$ and $\beta = 3$.

Fig. 2. The MSE versus $\frac{1}{\sigma^2}$ for the different estimators, where $P = 500$ and $\beta = 3$.

7) The LMMSE estimator evaluated for the linear approximation of the model in (66). That is, the nonlinear function from (66) is linearized by [18] $\mathbf{g}(\mathbf{L}, \mathbf{x}) \approx \mathbf{L}\mathbf{x}$. Then, $\mathbb{E}[\mathbf{y}] = \mathbf{0}$, $\mathbf{C}_{\mathbf{yy}} = \beta\mathbf{L} + \sigma^2\mathbf{I}_N$, and $\mathbf{C}_{\mathbf{xy}} = \beta\mathbf{L}^\dagger\mathbf{L}$, resulting in

$$\hat{\mathbf{x}}^{(\text{aLMMSE})} = \beta\mathbf{L}^\dagger\mathbf{L}(\beta\mathbf{L} + \sigma^2\mathbf{I}_N)^{-1}\mathbf{y}. \tag{68}$$

C. Example A: Stationary Network

In this subsection, we investigate the case where the topology is constant and, thus, the statistical properties of $\mathbf{y}, \mathbf{x}$ are constant. In Fig. 1, we present the MSE of the methods from Section VI-B for different values of P, i.e., different numbers of training data points used to evaluate the sample-mean values. It can be seen that the linearized-model based estimator from (68) and the lower bound obtained by P_∞-LMMSE are independent of P, as expected. The sample-LMMSE estimator from (26) uses the inverse of the sample covariance matrix $\hat{\mathbf{C}}_{\mathbf{yy}}$, which requires a large number of training data points to achieve a stable estimation. Thus, it can be seen that for $P < 10\,N$, where $N = 118$, the MSE of the sample-LMMSE estimator is higher than the MSE of the proposed methods: the sample-GSP-LMMSE, the sample linear pseudo-inverse GSP, and the sample ARMA GSP estimators.

In this example, the sample linear pseudo-inverse GSP and the sample ARMA GSP estimators coincide with the sample-GSP-LMMSE estimator. Thus, the chosen parametrizations are an accurate approximation of the desired graph frequency response. In addition, for $P > 10^4$, these GSP estimators and the sample-LMMSE estimator converge. It can be seen that the MSE of the LMMSE estimator, represented by P_∞-LMMSE, provides a lower bound on the MSE of any linear estimator, where the GSP-LMMSE, linear pseudo-inverse GSP, and ARMA GSP estimators achieve this lower bound for a much smaller value of P than the sample-LMMSE estimator. This result holds although Condition C.1 from Theorem 2 is not satisfied and the proposed GSP-LMMSE estimator differs from the LMMSE estimator. Finally, the sample LR-ARMA GSP estimator has a lower MSE than the sample-LMMSE estimator for $P < 3\,N$ and achieves
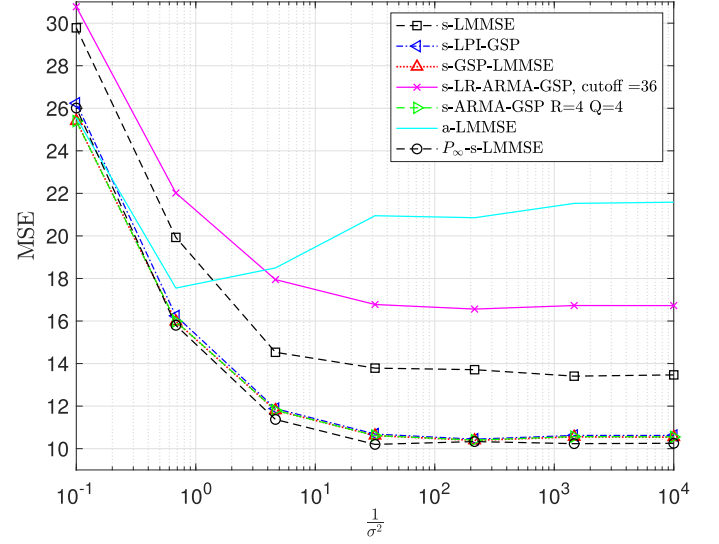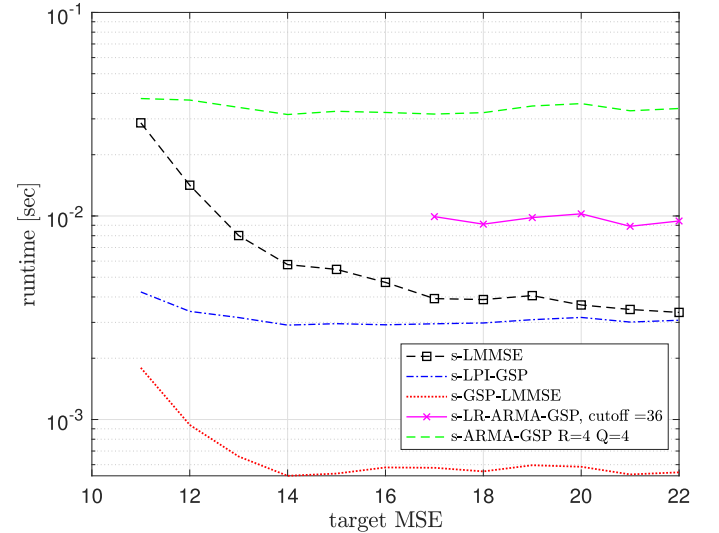

Fig. 3. The runtime for the different estimators versus the target MSE, where $\sigma^2 = 0.05$ and $\beta = 3$.

lower MSE than the LMMSE estimator evaluated for the linear approximation from (68) for $P > 0.1\,N$.

In Fig. 2, we present the MSE versus the noise variance, σ^2 for $P = 500$ and $\beta = 3$. It can be seen that the MSE of all estimators (except the aLMMSE from (68), which is based on a linearization of the model) increases as the noise variance increases. In this case, the sample-GSP-LMMSE, the sample linear pseudo-inverse GSP, and the sample ARMA GSP estimators outperform the sample-LMMSE estimator and approach the lower bound obtained by P_∞-LMMSE for all values of the noise variance.

In order to demonstrate the complexity of the estimators empirically, the average computation time was evaluated using Matlab on an Intel Core(TM) i7-7700 K CPU computer, 4.2 GHz. In Fig. 3, we present the runtime of the different estimators versus the target MSE, where $\sigma^2 = 0.05$ and $\beta = 3$. It can be seen that the runtime of all the estimators increases as the

target MSE decreases. The runtimes of the sample ARMA GSP and the sample LR-ARMA GSP estimators are the highest since finding their optimal coefficients requires solving a nonconvex optimization problem (see Sections V-C and V-D, respectively), while the LR-ARMA has a lower runtime since it has fewer coefficients. The proposed sample-GSP-LMMSE estimator has the lowest runtime for any target MSE, and the proposed sample linear pseudo-inverse GSP estimator is the second-best in terms of runtime.

It should be noted that the topology is stationary in this simulation. Thus, the EVD of the Laplacian matrix, $\mathbf{L}$, is assumed to be known and given in advance. When there is a change in the network, the computational complexity of updating the sample-h-GSP estimators is much lower than those of the other estimators and, thus, has a shorter runtime. This is since the sample-h-GSP estimators use the graph filter coefficients, $\hat{\boldsymbol{\alpha}}^{\text{sample}}$, that were evaluated on the initial topology. In contrast, the sample-LMMSE and the sample-GSP-LMMSE estimators require a reevaluation for each new topology, as shown in the following subsection.

D. Example B: Estimation Under Topology Changes

In this subsection, we discuss the case where the underlying topology changes over time. For example, when a sensor fails or changes its location, the sensor network's topology changes. Similarly, the power grid topology may be changed by failure, opening and closing of switches on power lines, and the presence of new loads and generators. When $\mathbf{L}$ changes, the measurement function, $\mathbf{g}(\mathbf{L}, \mathbf{x})$, from the model in (19) and the distribution from (67), change. Our goal is to estimate $\mathbf{x}$ without generating new dataset. The MSE of the different estimators is considered under random changes in the topology with the constraints that the graph will remain well-connected, i.e., assuming that λ_2, which is related to the connectivity [39] did not reduce significantly. The MSE shown is the average MSE over 100 random changes on the graph.

1) Estimation Under Edge Changes: In this case, the changes in the topology are due to the addition or removal of edges. Thus, the problem dimension did not change. In order to evaluate all the estimators from Section VI-B we use the mean of $\mathbf{x}$, i.e., $\mathbb{E}[\mathbf{x}]$, the eigenvalue and eigenvectors of the Laplacian matrix with the historical sample values, such as $\hat{\mathbf{C}}_{\mathbf{yy}}$ from (25), $\hat{\mathbf{D}}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ from (40). It should be noted that in the following, the sample-LMMSE estimator is based on the initial topology, where the sample-GSP-LMMSE, the sample linear pseudo-inverse GSP, and the sample ARMA GSP estimators have been updated to the new topology, as described after (47).

Fig. 4(a) and 4(b) present the graph frequency response of the different sample-GSP estimators, where $P = 500$, $\sigma^2 = 0.05$ and $\beta = 3$ for the stationary network from Example A (Fig. 4(a)) and for the topology change from Example B, where $M = 7$ new edges were added (Fig. 4(b)). It can be seen that in both cases, all GSP filters achieve almost the same graph frequency response as the sample-GSP-LMMSE estimator and, thus, they can be considered as robust to topology changes. The graph frequency response of the sample GSP-LMMSE at λ_1 is 0, which is approximated by the h-GSP filters as a small value. However, it can be seen that the graph frequency response is a nonzero (small) response (i.e. the output $\mathbf{y}$ is not a perfect graph low-frequency signal) while the graph frequency response of the LR-ARMA GSP estimator is absolutely zero for $\lambda > 36$.
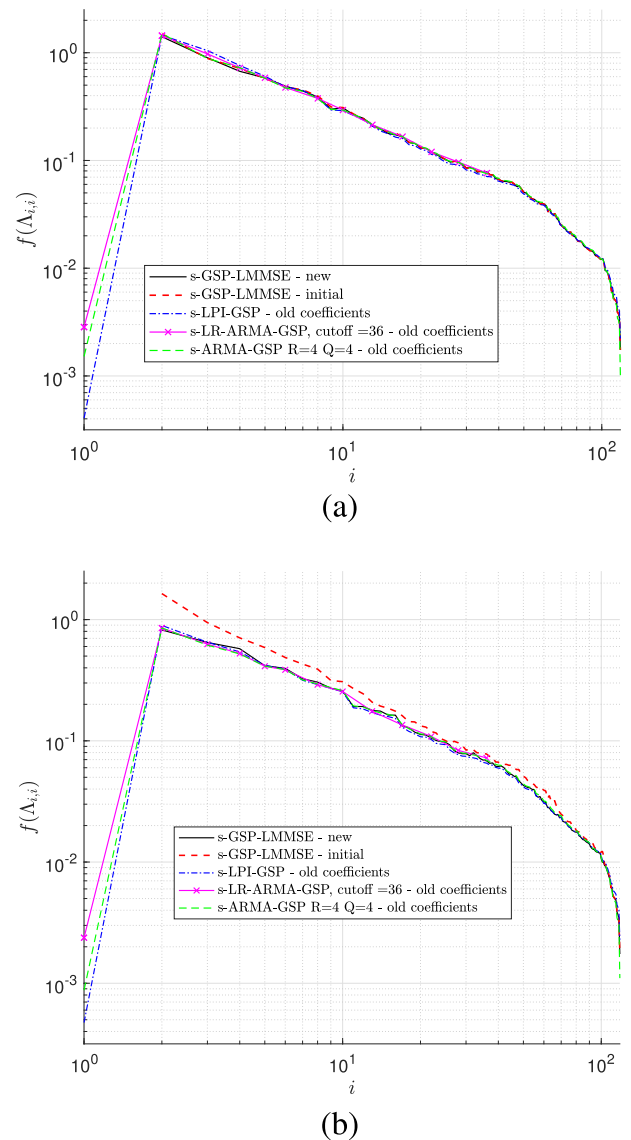

(a)

(b)

Fig. 4. The graph frequency response of the different estimators, where $P = 500$, $\sigma^2 = 0.05$ and $\beta = 3$, for (a) a stationary network; and (b) a network with the addition of $M = 7$ edges.

Therefore, the LR-ARMA GSP estimator does not perform well in the simulations. For perfectly low-frequency signals (not shown here due to space limitations), the LR-ARMA GSP estimator achieves the same performance as the other h-GSP estimators. In addition, it can be seen in Fig. 4(b) that the graph frequency response of the sample-GSP-LMMSE estimator that was evaluated based on the initial topology (red) is a less accurate approximation of the optimal graph frequency response. Finally, the desired graph frequency response, $\hat{f}(\boldsymbol{\Lambda})$, includes a sharp transition since $f(\lambda_1) = 0$; thus, a linear graph filter as in (12) is inappropriate for this case since it requires a high filter order leading to a high implementation cost and limited accuracy [29].
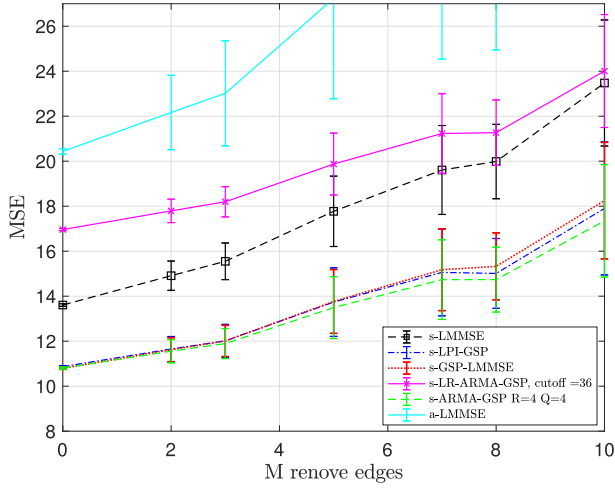
Fig. 5(a) and 5(b) present the case where the sample-mean values were calculated from the dataset which is evaluated on a topology before M edges are added or removed. Since there is no straightforward methodology to update the sample-LMMSE estimator to the new topology, its performance in the sense of the MSE is impaired for both cases: added and removed edges.
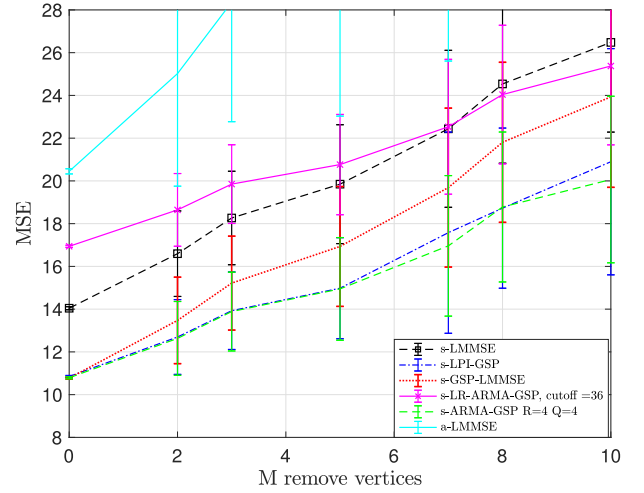
(a)
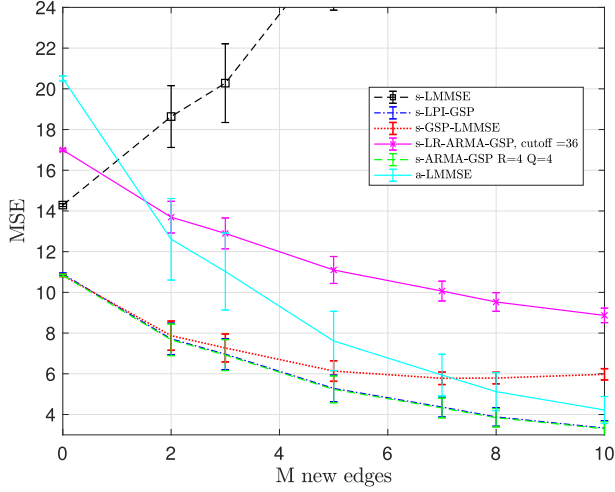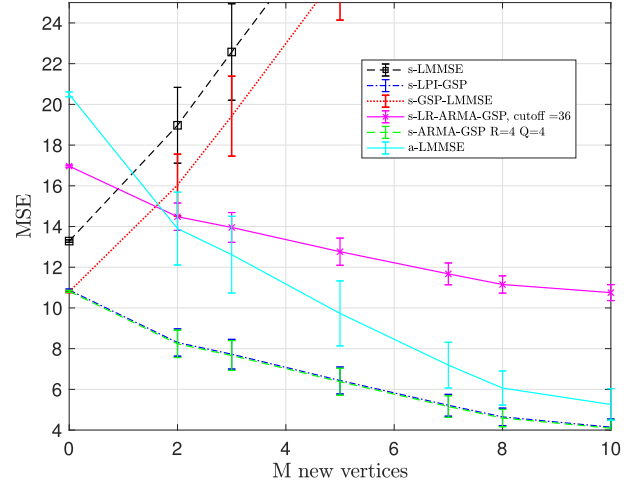
(b)

Fig. 5.   The MSE of the updated GSP estimators, the sample-LMMSE estimator, and the aLMMSE estimator, where $P = 500$, $\sigma^2 = 0.05$, and $\beta = 3$, for (a) the addition of M new edges; and (b) a removal of M edges. The error bars show confidence intervals of ± 0.5 standard deviations.

(a)

(b)

Fig. 6.   The MSE of the updated GSP estimators, the sample-LMMSE estimator, and the aLMMSE estimator, where $P = 500$, $\sigma^2 = 0.05$, and $\beta = 3$, for (a) an addition of M new vertices; and (b) a removal of M. The error bars show confidence intervals of ± 0.5 standard deviations.

Moreover, even when a small number of new edges is added, the MSE of the sample-LMMSE significantly increases. It can be seen that even for 2 new edges, the sample LR-ARMA GSP estimator has a lower MSE than the sample-LMMSE estimator. The sample linear pseudo-inverse GSP and the sample ARMA GSP estimators have a lower MSE for any number of edges added or removed. The sample-GSP-LMMSE estimator that has been updated to the new topology, has the same MSE as the sample linear pseudo-inverse GSP estimator and the sample ARMA GSP estimator for a small number of edges added or removed, and a slightly higher MSE for a large number of edges added or removed. The aMMSE estimator performance increases for edges removed and improves for the case where new edges are added. It should be noted that in addition to their advantage in terms of MSE, the computational complexity of updating the sample-h-GSP estimators is lower than those of the sample-LMMSE and the sample-GSP-LMMSE estimators. This is since the sample-h-GSP estimators use the graph filter

coefficients, $\hat{\boldsymbol{\alpha}}^{\text{sample}}$, that were evaluated on the initial topology, while the sample-LMMSE and the sample-GSP-LMMSE estimators require to reevaluate the estimators from scratch for the new topology.

2) Estimation Under Vertices Changes: When M vertices are removed or added, the problem dimension changes. That is, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{N \pm M}$. Since the sample-LMMSE and the sample-GSP-LMMSE estimators are $\mathbb{R}^N \to \mathbb{R}^N$ estimators, they cannot be implemented in the new problem. Therefore, we use the following methods: 1) for M new vertices, the sample-LMMSE and the sample-GSP-LMMSE estimators estimate the signal at the new vertices by zero and do not use measurements from those vertices; 2) for M removed vertices, the sample-LMMSE and the sample-GSP-LMMSE estimators are updated by removing the appropriate rows and columns. That is, the sample-LMMSE estimator from (26) is given by

$$\hat{\mathbf{x}}_{\mathcal{S}}^{(\text{sLMMSE})} = \mathbb{E}[\mathbf{x}] + [\hat{\mathbf{C}}_{\mathbf{xy}} \hat{\mathbf{C}}_{\mathbf{yy}}^{-1}]_{\mathcal{S}} (\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}), \qquad (69)$$

and the sample-GSP-LMMSE estimator from (41) is given by

$$\hat{\mathbf{x}}_{\mathcal{S}}^{(\text{sGSP-LMMSE})} = \mathbb{E}[\mathbf{x}] + [\mathbf{V}\hat{f}(\mathbf{\Lambda})\mathbf{V}^T]_{\mathcal{S}}(\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}), \quad (70)$$

where we use the historical eigenvalue and eigenvectors of the Laplacian matrix and the historical sample values of the covariance matrices, evaluated using the historical dataset. In (69) and (70) we use the notation that $\mathbf{A}_{\mathcal{S}}$ is the submatrix of $\mathbf{A}$ whose rows and columns are indicated by the set $\mathcal{S}$, where $\mathcal{S}$ is the set of the remaining vertices. In addition, any sample-h-GSP estimator from (51), can be updated to the new topology for both cases: vertices added or removed, as follows:

$$\hat{\mathbf{x}}^{(\text{update-sh-GSP})} = \mathbb{E}[\mathbf{x}] + \mathbf{V}h(\mathbf{\Lambda}; \hat{\boldsymbol{\alpha}}^{\text{sample}})\mathbf{V}^T(\mathbf{y} - \hat{\mathbf{y}}), \quad (71)$$

where we use the true value of $\mathbb{E}[\mathbf{x}]$, the eigenvalue and eigenvectors of the Laplacian matrix with the filter coefficient, $\hat{\boldsymbol{\alpha}}^{\text{sample}}$, that were evaluated using the historical dataset and $\hat{\mathbf{y}} \in \mathbb{R}^{N \pm M}$ that is defined by

$$[\hat{\bar{\mathbf{y}}}]_n = \begin{cases} 0 & \text{if } n \text{ is added} \\ [\hat{\mathbf{y}}]_n, & \text{if } n \text{ is unchanged} \end{cases}, \; n = 1, \dots, N, \quad (72)$$

where $\hat{\mathbf{y}}$ is evaluated for the old topology by (23).

Fig. 6(a) and 6(b) present the case where the historical sample values were calculated from the dataset evaluated on a topology before the addition or removal of the M vertices, respectively. Since there is no straightforward methodology to update the sample-LMMSE and the sample-GSP-LMMSE estimators to the new topology, their MSE is impaired for added and removed vertices. Moreover, even when a small number of new vertices is added, the MSE of the sample-LMMSE and the sample-GSP-LMMSE estimators significantly increases. The other estimators' performance for the cases of added or removed vertices is similar to the result in the cases of added or removed edges in Fig. 5(a) and 5(b). In Fig. 6(a) and 6(b), the confidence intervals of ± 0.5 standard deviations are significant due to the variability in the topology of the different experiments. Thus, increasing the number of experiments does not reduce this confidence interval.

VII. CONCLUSION

In this paper, we discuss a GSP-based Bayesian approach for the recovery of random graph signals from nonlinear measurements. We develop the GSP-LMMSE estimator, which minimizes the MSE among the subset of estimators that are represented as an output of a graph filter. We evaluate the conditions for the GSP-LMMSE estimator to coincide with the LMMSE estimator and with the graphical Wiener filter. If the distributions of the graph signal and the observations are intractable, the sample-mean versions of the different estimators can be used. The diagonal structure of the sample-GSP-LMMSE estimator in the graph frequency domain bypasses the requirement for an extensive dataset to obtain stable estimation of the sample-LMMSE estimator. However, the GSP-LMMSE estimator is a function of the specific graph structure with fixed dimensions, and thus it is not necessarily optimal when the topology changes and is not adaptive to changes in the number of vertices. Therefore, we develop the sample-h-GSP estimators that are the MSE-optimal parametrization of the sample-GSP-LMMSE estimator by graph filters. The sample-h-GSP estimators can be updated when the topology changes without generating a new dataset, even in the case of changes in the number of vertices.

In the simulations, we show that the proposed sample-GSP estimators achieve lower MSE than the sample-LMMSE estimator

for a limited training dataset, and they coincide with the sample-LMMSE estimator for sufficiently large datasets. In addition, it is shown that the three specific parametric implementations of the GSP-LMMSE: the linear pseudo-inverse GSP estimator, ARMA GSP estimator, and the low-rank ARMA GSP estimator, are robust to changes in the topology without the need for generating new training data. The sample linear pseudo-inverse GSP and the sample ARMA GSP estimators achieve the lowest MSE in these cases, where the ARMA GSP estimator requires less filter coefficients. Thus, the proposed approach is a practical method to recover nonlinear graph signals in networks.

There are several directions left for future work. One direction is to study the use of graph neural networks and other nonlinear approaches [33], [34]. In addition, the development of Bayesian bounds on the MSE of general (not necessarily linear) estimators of graph signals, in a similar manner to the non-Bayesian graph Cramér-Rao bound from [46], should be investigated. Finally, it is interesting to consider distributed implementation of the proposed estimators that include the computation of the optimal coefficient vector and the diagonal sample covariance matrices.

APPENDIX A
PROOF OF THEOREM 2

In this Appendix, we show that under the conditions of Theorem 2 the equality in (42) holds.

Since $\mathbf{x}$ and $\mathbf{w}$ are statistically independent under the considered model, the covariance matrix of $\tilde{\mathbf{y}}$ is given by

$$\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \mathbf{C}_{\tilde{\mathbf{g}}\tilde{\mathbf{g}}} + \mathbf{C}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}}, \quad (73)$$

where $\tilde{\mathbf{g}} = \mathbf{V}^T \mathbf{g}(\mathbf{L}, \mathbf{x})$. Similarly,

$$\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} = \mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{g}}}. \quad (74)$$

Since from Condition C.1 the measurement function, $\mathbf{g}(\mathbf{L}, \mathbf{x})$, satisfies (45), we obtain that the off-diagonal elements of the matrix $\mathbf{C}_{\tilde{\mathbf{g}}\tilde{\mathbf{g}}}$ from (73) satisfy

$$\begin{aligned} [\mathbf{C}_{\tilde{\mathbf{g}}\tilde{\mathbf{g}}}]_{n,k} &= \mathbb{E}\left[([\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_n - \mathbb{E}[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_n) \right. \\ &\quad \times ([\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_k - \mathbb{E}[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_k)] \\ &= \mathbb{E}\left[([\tilde{\mathbf{g}}(\mathbf{L}, \tilde{x}_n \mathbf{v}_n)]_n - \mathbb{E}[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_n) \right. \\ &\quad \times ([\tilde{\mathbf{g}}(\mathbf{L}, \tilde{x}_k \mathbf{v}_k)]_k - \mathbb{E}[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_k)] = 0, \quad (75) \end{aligned}$$

for any $n \neq k$, where the last equality follows from Condition C.2. Substituting (75) in (73) and using Condition C.3, which implies that $\mathbf{C}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}}$ is a diagonal matrix, the matrix $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is also a diagonal matrix, which satisfies

$$\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \text{diag}\left(\text{diag}(\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}})\right) = \mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}. \quad (76)$$

Similarly, using Condition C.1, the off-diagonal elements of the cross-covariance matrix, $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{g}}}$ from (74) satisfy

$$\begin{aligned} [\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{g}}}]_{n,k} &= \mathbb{E}\left[([\tilde{\mathbf{x}}]_n - \mathbb{E}[\tilde{\mathbf{x}}]_n)([\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_k - \mathbb{E}[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_k)\right] \\ &= \mathbb{E}\left[([\tilde{\mathbf{x}}]_n - \mathbb{E}[\tilde{\mathbf{x}}]_n)([\tilde{\mathbf{g}}(\mathbf{L}, \tilde{x}_k \mathbf{v}_k)]_k - \mathbb{E}[\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x})]_k)\right] \\ &= 0, \quad (77) \end{aligned}$$

for any $n \neq k$, where the last equality follows from Condition C.2. By substituting (77) in (74), we have

$$\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} = \text{diag}(\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}) = \text{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}). \quad (78)$$

Therefore, (76) and (78) imply that $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ and $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are diagonal matrices, and that (42) holds.

APPENDIX B
PROOF OF THEOREM 3

In this Appendix, we show that under the assumption of Theorem 3 the equality in (42) holds. By using (46) from Condition C.4, we obtain that $\tilde{\mathbf{g}}(\mathbf{L}, \mathbf{x}) = f(\mathbf{\Lambda})\mathbf{V}^T\mathbf{x}$ and thus,

$$\mathbf{C}_{\tilde{\mathbf{g}}\tilde{\mathbf{g}}} = \mathbb{E}[(f(\mathbf{\Lambda})(\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}]))(f(\mathbf{\Lambda})(\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}]))^T]$$

$$= f(\mathbf{\Lambda})\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}f(\mathbf{\Lambda}), \tag{79}$$

where we use the symmetry of $f(\mathbf{\Lambda})$. By substituting (79) in (73) from Appendix A and using Condition C.3 and Condition C.5, which implies that $\mathbf{C}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}}$ and $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$ are diagonal matrices, we obtain that $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a diagonal matrix, which satisfies

$$\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \text{diag}\left(\text{diag}(\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}})\right) = \mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}. \tag{80}$$

Similarly, the cross-covariance matrix, $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{g}}}$ from (74) satisfies

$$\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{g}}} = \mathbb{E}[(\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}])(f(\mathbf{\Lambda})(\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}]))^T] = \mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}f(\mathbf{\Lambda}), \tag{81}$$

where we use the fact that $f(\mathbf{\Lambda})$ is a symmetric matrix. By substituting (81) in (74) from Appendix A, the cross-covariance matrix of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ is a diagonal matrix, and satisfies

$$\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}} = \text{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}), \tag{82}$$

where $\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ is defined in (37). Therefore, (80) and (82) imply that $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ and $\mathbf{C}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are diagonal matrices, and that (42) holds.

APPENDIX C
DERIVATION OF (48)

In this Appendix we show that solving (47) is equivalent to (48). By adding and subtracting $\hat{f}(\mathbf{\Lambda})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}])$ from the r.h.s. of (47), one obtains

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}\in\Omega_{\boldsymbol{\alpha}}}{\arg\min} \mathbb{E}[||h(\mathbf{\Lambda};\boldsymbol{\alpha})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]) - (\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}])$$

$$- \hat{f}(\mathbf{\Lambda})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]) + \hat{f}(\mathbf{\Lambda})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}])||^2]$$

$$= \underset{\boldsymbol{\alpha}\in\Omega_{\boldsymbol{\alpha}}}{\arg\min} \mathbb{E}[|||((h(\mathbf{\Lambda};\boldsymbol{\alpha}) - \hat{f}(\mathbf{\Lambda}))(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]))||^2]$$

$$+ 2\,\mathbb{E}[(\hat{f}(\mathbf{\Lambda})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]) - ((\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}])))^T$$

$$\times ((h(\mathbf{\Lambda};\boldsymbol{\alpha}) - \hat{f}(\mathbf{\Lambda}))(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]))], \tag{83}$$

where the last equality is obtained by removing constant terms w.r.t. $\boldsymbol{\alpha}$. In addition, by substituting $\hat{f}(\mathbf{\Lambda})$ from (36) in the last term of (83) it can be verified that

$$\mathbb{E}[(\hat{f}(\mathbf{\Lambda})(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]) - ((\tilde{\mathbf{x}} - \mathbb{E}[\tilde{\mathbf{x}}])))^T$$

$$\times ((h(\mathbf{\Lambda};\boldsymbol{\alpha}) - \hat{f}(\mathbf{\Lambda}))(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}]))]$$

$$= (\text{diag}(\hat{f}(\mathbf{\Lambda})))^T\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}(\text{diag}(h(\mathbf{\Lambda};\boldsymbol{\alpha})) - \text{diag}(\hat{f}(\mathbf{\Lambda})))$$

$$- \mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}^T(\text{diag}(h(\mathbf{\Lambda};\boldsymbol{\alpha})) - \text{diag}(\hat{f}(\mathbf{\Lambda}))) = 0, \tag{84}$$

where $\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ and $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ are defined in (37). By substituting (84) in (83), one obtains

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}\in\Omega_{\boldsymbol{\alpha}}}{\arg\min} \mathbb{E}[|||(h(\mathbf{\Lambda};\boldsymbol{\alpha}) - \hat{f}(\mathbf{\Lambda}))(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}])||^2]$$

$$= \mathbb{E}[||\text{diag}(\tilde{\mathbf{y}} - \mathbb{E}[\tilde{\mathbf{y}}])\text{diag}(h(\mathbf{\Lambda};\boldsymbol{\alpha}) - \hat{f}(\mathbf{\Lambda}))||^2]$$

$$= (\text{diag}(h(\mathbf{\Lambda};\boldsymbol{\alpha})) - \text{diag}(\hat{f}(\mathbf{\Lambda})))^T\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$$

$$\times (\text{diag}(h(\mathbf{\Lambda};\boldsymbol{\alpha})) - \text{diag}(\hat{f}(\mathbf{\Lambda}))), \tag{85}$$

where the second equality follows since $h(\mathbf{\Lambda};\boldsymbol{\alpha}))$ and $(\hat{f}(\mathbf{\Lambda})$ are diagonal matrices. Then, since $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is a diagonal matrix, and thus, a symmetric matrix and it is assumed that it is also a nonsingular (and therefore, positive definite) matrix, we obtain (48) by substituting $\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{\frac{1}{2}}\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{\frac{1}{2}}$ and $\text{diag}(\text{diag}(\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}})\mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}) = \mathbf{D}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1}\mathbf{d}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}}$ in (85).

APPENDIX D
PROOF OF CLAIM 1 AND CLAIM 2

We prove Claim 1 by showing that $\text{rank}(\bar{\mathbf{\Gamma}}) = K + 1$ under the condition in Claim 1, where $\bar{\mathbf{\Gamma}}_K$ is defined in (53) and $K + 1 \leq N$. First, it can be seen that

$$\bar{\mathbf{\Gamma}}_K = \text{diag}(1, \lambda_2^{-K}, \ldots, \lambda_N^{-K})\mathbf{B}_K(\lambda_1, \ldots, \lambda_N), \tag{86}$$

where

$$\mathbf{B}_K(\lambda_1, \ldots, \lambda_N) \triangleq \begin{bmatrix} 1 & 0 & \ldots & 0 & 0 \\ \lambda_2^K & \lambda_2^{K-1} & \ldots & \lambda_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \lambda_N^K & \lambda_N^{K-1} & \ldots & \lambda_N & 1 \end{bmatrix}.$$

We assume that the graph is connected, i.e. $0 < \lambda_n$, $n = 2, \ldots, N$. Thus, $\text{diag}(1, \lambda_1^{-K}, \ldots, \lambda_N^{-K})$ is a non-singular matrix. The multiplication of $\mathbf{B}_K(\lambda_1, \ldots, \lambda_N)$ by an $N \times N$ non-singular matrix, in (86) implies that (see 0.4.6 in [60])

$$\text{rank}(\bar{\mathbf{\Gamma}}_K) = \text{rank}(\mathbf{B}_K(\lambda_1, \ldots, \lambda_N)). \tag{87}$$

By reordering the columns of $\mathbf{B}_K(\lambda_1, \ldots, \lambda_N)$ and using the properties of the Vandermonde matrix $\mathbf{\Phi}(N - 1, K)$ from (58) (see 0.9.11 [60]), we obtain that if there are $K + 1$ distinct eigenvalues, then $\text{rank}(\mathbf{B}_K(\lambda_1, \ldots, \lambda_N)) = K + 1$. By using (87), this statement implies that if there are $K + 1$ distinct eigenvalues, then $\text{rank}(\bar{\mathbf{\Gamma}}_K) = K + 1$.

Second, we prove Claim 2 by showing that

$$\text{rank}((\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}\mathbf{\Phi}(N, Q)) = Q + 1 \tag{88}$$

under the condition in Claim 2, where $\mathbf{\Phi}(N, R)$ and $\mathbf{\Phi}(N, Q)$ are defined in (58), and $Q + 1 \leq N$. We assume that $[\mathbf{\Phi}(N, R)\mathbf{a}]_n \neq 0, \forall n = 1, \ldots, N$ and thus, $(\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))$ is a non-singular matrix. The multiplication of $\mathbf{\Phi}(N, Q)$ by an $N \times N$ non-singular matrix, $(\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}$, in (88) implies that (see 0.4.6 in [60])

$$\text{rank}((\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}\mathbf{\Phi}(N, Q)) = \text{rank}(\mathbf{\Phi}(N, Q)). \tag{89}$$

Using the properties (see 0.9.11 [60]) of the Vandermonde matrix, $\mathbf{\Phi}(N, Q)$, we obtain that if there are $Q + 1$ distinct eigenvalues, then $\text{rank}(\mathbf{\Phi}(N, Q)) = Q + 1$. By using (89), this statement implies that if there are $Q + 1$ distinct eigenvalues, then $\text{rank}((\text{diag}(\mathbf{\Phi}(N, R)\mathbf{a}))^{-1}\mathbf{\Phi}(N, Q)) = Q + 1$.

REFERENCES

[1] W. Huang, L. Goldsberry, N. F. Wymbs, S. T. Grafton, D. S. Bassett, and A. Ribeiro, "Graph frequency analysis of brain signals," *IEEE J. Sel. Top. Signal Process.*, vol. 10, no. 7, pp. 1189–1203, Oct. 2016.

[2] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. 44th Conf. Decis. Control*, 2005, pp. 6698–6703.

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

[4] 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.

[5] E. Isufi, A. Loukas, A. Simonetto, and G. Leus, "Autoregressive moving average graph filtering," *IEEE Trans. Signal Process.*, vol. 65, no. 2, pp. 274–288, Jan. 2017.

[6] Y. Tanaka, Y. C. Eldar, A. Ortega, and G. Cheung, "Sampling signals on graphs: From theory to applications," *IEEE Signal Process. Mag.*, vol. 37, no. 6, pp. 14–30, Nov. 2020.

[7] A. Anis, A. Gadde, and A. Ortega, "Towards a sampling theorem for signals on arbitrary graphs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 3864–3868.

[8] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Sampling of graph signals with successive local aggregations," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1832–1843, Apr. 2016.

[9] Y. Tanaka and Y. C. Eldar, "Generalized sampling on graphs with subspace and smoothness priors," *IEEE Trans. Signal Process.*, vol. 68, pp. 2272–2286, Mar. 2020.

[10] Z. Xiao, H. Fang, and X. Wang, "Distributed nonlinear polynomial graph filter and its output graph spectrum: Filter analysis and design," *IEEE Trans. Signal Process.*, vol. 69, pp. 1–15, Feb. 2021.

[11] G. B. Giannakis, Y. Shen, and G. V. Karanikolas, "Topology identification and learning over graphs: Accounting for nonlinearities and dynamics," *Proc. IEEE*, vol. 106, no. 5, pp. 787–807, May 2018.

[12] Y. Shen, G. B. Giannakis, and B. Baingana, "Nonlinear structural vector autoregressive models with application to directed brain networks," *IEEE Trans. Signal Process.*, vol. 67, no. 20, pp. 5325–5339, Oct. 2019.

[13] Z. Xiao and X. Wang, "Nonlinear polynomial graph filter for signal processing with irregular structures," *IEEE Trans. Signal Process.*, vol. 66, no. 23, pp. 6241–6251, Dec. 2018.

[14] E. Drayer and T. Routtenberg, "Detection of false data injection attacks in smart grids based on graph signal processing," *IEEE Syst. J.*, vol. 14, no. 2, pp. 1886–1896, Jun. 2020.

[15] S. Grotas, Y. Yakoby, I. Gera, and T. Routtenberg, "Power systems topology and state estimation by graph blind source separation," *IEEE Trans. Signal Process.*, vol. 67, no. 8, pp. 2036–2051, Apr. 2019.

[16] S. Shaked and T. Routtenberg, "Identification of edge disconnections in networks based on graph filter outputs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 7, pp. 578–594, Aug. 2021.

[17] D. Bienstock and A. Verma, "Strong NP-hardness of AC power flows feasibility," *Oper. Res. Lett.*, vol. 47, no. 6, pp. 494–501, 2019.

[18] A. Abur and A. Gomez-Exposito, *Power System State Estimation: Theory and Implementation*. New York, NY, USA: Marcel Dekker, 2004.

[19] E. Björnson, J. Hoydis, M. Kountouris, and M. Debbah, "Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits," *IEEE Trans. Inf. Theory*, vol. 60, no. 11, pp. 7112–7139, Nov. 2014.

[20] Y. Li, C. Tao, G. Seco-Granados, A. Mezghani, A. L. Swindlehurst, and L. Liu, "Channel estimation and performance analysis of one-bit massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4075–4089, Aug. 2017.

[21] I. E. Berman and T. Routtenberg, "Resource allocation and dithering of Bayesian parameter estimation using mixed-resolution data," *IEEE Trans. Signal Process.*, vol. 69, pp. 6148–6164, Oct. 2021.

[22] Y. C. Eldar and N. Merhav, "A competitive minimax approach to robust estimation of random parameters," *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 1931–1946, Jul. 2004.

[23] O. Edfors, M. Sandell, J. J. van de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," in *Proc. Veh. Technol. Conf.*, 1996, vol. 2, pp. 923–927.

[24] N. Geng, X. Yuan, and L. Ping, "Dual-diagonal LMMSE channel estimation for OFDM systems," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4734–4746, Sep. 2012.

[25] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal denoising on graphs via graph filtering," in *Proc. IEEE Glob. Conf. Signal Inf. Process.*, 2014, pp. 872–876.

[26] F. Zhang and E. R. Hancock, "Graph spectral image smoothing using the heat kernel," *Pattern Recognit.*, vol. 41, pp. 3328–3342, 2008.

[27] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovačević, "Semi-supervised multiresolution classification using adaptive graph filtering with application to indirect bridge structural health monitoring," *IEEE Trans. Signal Process.*, vol. 62, no. 11, pp. 2879–2893, Jun. 2014.

[28] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3042–3054, Jun. 2014.

[29] J. Liu, E. Isufi, and G. Leus, "Filter design for autoregressive moving average graph filters," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 1, pp. 47–60, Mar. 2019.

[30] 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, Aug. 2017.

[31] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovačević, "Signal recovery on graphs: Variation minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4609–4624, Sep. 2015.

[32] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, pp. 3462–3477, Apr. 2017.

[33] L. Ruiz, F. Gama, and A. Ribeiro, "Graph neural networks: Architectures, stability, and transferability," *Proc. IEEE*, vol. 109, no. 5, pp. 660–682, May 2021.

[34] F. Gama, E. Isufi, G. Leus, and A. Ribeiro, "Graphs, convolutions, and neural networks: From graph filters to graph neural networks," *IEEE Signal Process. Mag.*, vol. 37, no. 6, pp. 128–138, Nov. 2020.

[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] F. Hua, R. Nassif, C. Richard, H. Wang, and A. H. Sayed, "Online distributed learning over graphs with multitask graph-filter models," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 63–77, Jan. 2020.

[37] S. Rey and A. G. Marques, "Robust graph-filter identification with graph denoising regularization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 5300–5304. doi: 10.1109/ICASSP39728.2021.9414909.

[38] A. Kroizer, Y. C. Eldar, and T. Routtenberg, "Modeling and recovery of graph signals and difference-based signals," in *Proc. IEEE Glob. Conf. Signal Inf. Process.*, 2019, pp. 1–5.

[39] M. Newman, *Networks: An Introduction*. New York, NY, USA: Oxford Univ. Press, 2010.

[40] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Graph filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 6163–6166.

[41] F. Fouss, A. Pirotte, J. Renders, and M. Saerens, "Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 3, pp. 355–369, Mar. 2007.

[42] F. Fouss, L. Yen, A. Pirotte, and M. Saerens, "An experimental investigation of graph kernels on a collaborative recommendation task," in *Proc. 6th Int. Conf. Data Mining*, 2006, pp. 863–868.

[43] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*, 1st ed. USA: Wiley, 1996.

[44] X. Shi, H. Feng, M. Zhai, T. Yang, and B. Hu, "Infinite impulse response graph filters in wireless sensor networks," *IEEE Signal Process. Lett.*, vol. 22, no. 8, pp. 1113–1117, Aug. 2015.

[45] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Trans. Signal Process.*, vol. 63, no. 24, pp. 6510–6523, Dec. 2015.

[46] T. Routtenberg, "Non-Bayesian estimation framework for signal recovery on graphs," *IEEE Trans. Signal Process.*, vol. 69, pp. 1169–1184, Jan. 2021.

[47] L. Dabush, A. Kroizer, and T. Routtenberg, "State estimation in unobservable power systems via graph signal processing tools," 2021. [Online]. Available: https://arxiv.org/abs/2106.02254

[48] L. K. Letting, Y. Hamam, and A. M. Abu-Mahfouz, "Estimation of water demand in water distribution systems using particle swarm optimization," *Water*, vol. 9, no. 8, pp. 1–16, 2017.

[49] A. E. Alaoui and A. Montanari, "On the computational tractability of statistical estimation on amenable graphs," *Probability Theory Related Fields*, vol. 181, no. 4, pp. 815–864, 2021.

[50] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory.*, vol. 1. Englewood Cliffs, NJ, USA: Prentice Hall PTR, 1993.

[51] J. Serra and M. Nájar, "Asymptotically optimal linear shrinkage of sample LMMSE and MVDR filters," *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3552–3564, Jul. 2014.

[52] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Hoboken, NJ, USA: Wiley, 2004.

[53] F. Rubio and X. Mestre, "Consistent reduced-rank LMMSE estimation with a limited number of samples per observation dimension," *IEEE Trans. Signal Process.*, vol. 57, no. 8, pp. 2889–2902, Aug. 2009.

[54] Y. Chen, A. Wiesel, Y. C. Eldar, and A. O. Hero, "Shrinkage algorithms for MMSE covariance estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5016–5029, Oct. 2010.

[55] O. Ledoit and M. Wolf, "A well-conditioned estimator for large-dimensional covariance matrices," *J. Multivariate Anal.*, vol. 88, no. 2, pp. 365–411, 2004.

[56] L. Dieci and T. Eirola, "On smooth decompositions of matrices," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 3, pp. 800–819, 1999.

[57] 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.

[58] M. Ramezani-Mayiami, M. Hajimirsadeghi, K. Skretting, R. S. Blum, and H. V. Poor, "Graph topology learning and signal recovery via Bayesian inference," in *Proc. IEEE Data Sci. Workshop*, 2019, pp. 52–56.

[59] R. Ramakrishna, H. T. Wai, and A. Scaglione, "A user guide to low-pass graph signal processing and its applications: Tools and applications," *IEEE Signal Process. Mag.*, vol. 37, no. 6, pp. 74–85, Nov. 2020.

[60] R. Horn and C. R. Johnson, *Matrix Analysis.*, 2nd ed. New York, NY, USA: Cambridge Univ. Press, 2012.

[61] B. Mohar, "The Laplacian spectrum of graphs," in *Graph Theory, Combinatorics., Appl.*, Y. Alavi, G. Chartrand, O. Ollermann, and A. Schwenk, Eds., New York, NY, USA: Wiley, 1991, pp. 871–898.

[62] F. Gama, J. Bruna, and A. Ribeiro, "Stability properties of graph neural networks," *IEEE Trans. Signal Process.*, vol. 68, pp. 5680–5695, Sep. 2020.

[63] Z. Gao, E. Isufi, and A. Ribeiro, "Stability of graph convolutional neural networks to stochastic perturbations," *Signal Process.*, vol. 188, 2021, Art. no. 108216.

[64] H. Kenlay, D. Thanou, and X. Dong, "Interpretable stability bounds for spectral graph filters," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 5388–5397.

[65] L. Le Magoarou, R. Gribonval, and N. Tremblay, "Approximate fast graph Fourier transforms via multilayer sparse approximations," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 2, pp. 407–420, Jun. 2018.

[66] K.-S. Lu and A. Ortega, "Fast graph Fourier transforms based on graph symmetry and bipartition," *IEEE Trans. Signal Process.*, vol. 67, no. 18, pp. 4855–4869, Sep. 2019.

[67] "Power systems test case archive." [Online]. Available: http://www.ee.washington.edu/research/pstca/

Ariel Kroizer received the B.Sc. (*magna cum laude*) and M.Sc. degrees in electrical and computer engineering from the Ben-Gurion University of the Negev, Beer-Sheva, Israel, in 2019 and 2022, respectively. He is currently an Algorithm Engineer. His research interests include statistical signal processing, estimation theory, detection theory, and optimization.

Tirza Routtenberg (Senior Member, IEEE) received the B.Sc. degree (*magna cum laude*) in biomedical engineering from the Technion Israel Institute of Technology, Haifa, Israel, in 2005, and the M.Sc. (*magna cum laude*) and Ph.D. degrees in electrical engineering from the Ben-Gurion University of the Negev, Beer-Sheva, Israel, in 2007 and 2012, respectively. She was a Postdoctoral Fellow with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, USA, from 2012 to 2014. Since October 2014, she is a Faculty Member with the School of Electrical and Computer Engineering and Ben-Gurion University of the Negev. Her research interests include signal processing in smart grid, statistical signal processing, estimation and detection theory, and signal processing on graphs. She is currently an Associate Editor of IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS and of IEEE SIGNAL PROCESSING LETTERS, and also a Member of the IEEE Signal Processing Theory and Methods Technical Committee. She was the recipient of the Best Student Paper Award in International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2011, in IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP) 2013 (co-author), in ICASSP 2017 (co-author), and in IEEE Workshop on Statistical Signal Processing (SSP) 2018 (co-author). She was awarded the Negev Scholarship in 2008, the Lev-Zion Scholarship in 2010, the Marc Rich Foundation Prize in 2011, and the Toronto Prize for Excellence in research in 2021.

Yonina C. Eldar (Fellow, IEEE) received the B.Sc. degree in physics and the B.Sc. degree in electrical engineering from Tel-Aviv University (TAU), Tel-Aviv, Israel, in 1995 and 1996, and the Ph.D. degree in electrical engineering and computer science in 2002 from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. She is currently a Professor with the Department of Mathematics and Computer Science, Weizmann Institute of Science, Rehovot, Israel. She was a Professor with the Department of Electrical Engineering, Technion - Israel Institute of Technology, Haifa, Israel. She is also a Visiting Professor with MIT, a Visiting Scientist with the Broad Institute, Cambridge, MA, USA, and an Adjunct Professor with Duke University, Durham, NC, USA, and was a Visiting Professor with Stanford University, Stanford, CA, USA. She is author of the book *Sampling Theory: Beyond Bandlimited Systems* and co-author of five other books published by Cambridge University Press. Her research interests include statistical signal processing, sampling theory and compressed sensing, learning and optimization methods, and their applications to biology, medical imaging and optics. She is a Member of the Israel Academy of Sciences and Humanities (elected 2017), an EURASIP Fellow, a Horev Fellow of the Leaders in Science and Technology program at the Technion and an Alon Fellow.

Dr. Eldar was the recipient of many awards for excellence in research and teaching, including the IEEE Signal Processing Society Technical Achievement Award (2013), IEEE/AESS Fred Nathanson Memorial Radar Award (2014), and IEEE Kiyo Tomiyasu Award (2016), Michael Bruno Memorial Award from the Rothschild Foundation, Weizmann Prize for Exact Sciences, Wolf Foundation Krill Prize for Excellence in Scientific Research, Henry Taub Prize for Excellence in Research (twice), Hershel Rich Innovation Award (three times), Award for Women with Distinguished Contributions, the Andre and Bella Meyer Lectureship, Career Development Chair at the Technion, Muriel & David Jacknow Award for Excellence in Teaching, and the Technion's Award for Excellence in Teaching (two times), several best paper awards and best demo awards together with her research students and colleagues including the SIAM outstanding Paper Prize, UFFC Outstanding Paper Award, the Signal Processing Society Best Paper Award and IET Circuits, Devices and Systems Premium Award, was selected as one of the 50 most influential women in Israel and in Asia, and is a highly cited Researcher.

She was a Member of the Young Israel Academy of Science and Humanities and the Israel Committee for Higher Education. She is the Editor in Chief of *Foundations and Trends in Signal Processing*, a Member of the IEEE Sensor Array and Multichannel Technical Committee and serves on several other IEEE committees. She was a Signal Processing Society Distinguished Lecturer, Member of the IEEE Signal Processing Theory and Methods and Bio Imaging Signal Processing technical committees, and was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the *EURASIP Journal of Signal Processing*, the *SIAM Journal on Matrix Analysis and Applications*, and the *SIAM Journal on Imaging Sciences*. She was the Co-Chair and Technical Co-Chair of several international conferences and workshops.