Extended Adjacency and Scale-dependent Graph Fourier Transform via Diffusion Distances

Vitor R. M. Elias, Wallace A. Martins, *Senior Member, IEEE* and Stefan Werner, *Senior Member, IEEE*

***Abstract*—This paper proposes the augmentation of the adjacency model of networks for graph signal processing. It is assumed that no information about the network is available, apart from the initial adjacency matrix. In the proposed model, additional edges are created according to a Markov relation imposed between nodes. This information is incorporated into the extended-adjacency matrix as a function of the diffusion distance between nodes. The diffusion distance measures similarities between nodes at a certain diffusion scale or time, and is a metric adopted from diffusion maps. Similarly, the proposed extended-adjacency matrix depends on the diffusion scale, which enables the definition of a scale-dependent graph Fourier transform. We conduct theoretical analyses of both the extended adjacency and the corresponding graph Fourier transform and show that different diffusion scales lead to different graph-frequency perspectives. At different scales, the transform discriminates shifted ranges of signal variations across the graph, revealing more information on the graph signal when compared to traditional approaches. The scale-dependent graph Fourier transform is applied for anomaly detection and is shown to outperform the conventional graph Fourier transform.**

***Index Terms*—diffusion distances, diffusion maps, extended adjacency, graph signal processing, scale-dependent graph Fourier transform.**

I. Introduction

LARGE quantities of heterogeneous data are constantly collected by numerous sensors, which are often geographically dispersed. Real networks and their corresponding data come in vastly different shapes and applications, ranging from genetic interaction networks [1] and the human brain [2] to sensor networks and smart cities [3]. The increased connectivity and availability of abundant data calls for methods that can uncover hidden connections between seemingly unrelated things in complex and irregular structures.

Graph signal processing (GSP) explores pairwise relations between signals residing on nodes of a graph [4]–[6]. In GSP, elements of networks are modeled as vertices (or nodes) of a mathematical structure – the graph – and relations between two connected elements are represented by edges [1]–[4]. The dimensionality of the data matches that of the graph, such that each entry is associated with a vertex.

Most GSP tools are functions of a graph-shift operator (GSO) matrix [7]–[9] that encodes relations between the graph nodes. For instance, the graph Fourier transform (GFT) is defined as the signal expansion in terms of the eigenbasis of the GSO. The literature contains several GSO definitions that suit different applications [2]–[20]. The two most commonly used GSOs are the adjacency matrix of the graph [7], and the graph Laplacian [6].

The application dependency of the GSO is related to the more fundamental problem of modeling the original network by a graph; different models have different properties that can be explored by GSP tools [11]–[20]. For a particular network and application, it is desirable to define a GSO that best describes node relations, so that the corresponding network signals can be better analyzed/processed. Consider, for instance, the frequency analysis yielded by the GFT. The spectrum of a graph is directly related to the eigenvalues of the GSO. As a consequence, changes in the GSO entries are reflected in the graph spectrum, possibly allowing the discrimination of different frequency contents of a same network signal. We assume here that adjacency matrices are initially sparse, rendering GSOs with a reduced number of edges. This is due to sparsity constraints commonly imposed upon adjacency matrices or application limitations, e.g., sensors identifying their own neighbors. Moreover, if only the adjacency information is available but accurate network information is unknown, updating or deriving a new GSO becomes a challenging task. This work proposes a method for augmenting an initial adjacency matrix for frequency analysis of networked data.

We derive virtual Markov relations between nodes and incorporate the Markov property into the GSO as a function of diffusion distances (DDs) between network elements. Markov relations occur naturally in some applications, such as in consensus [21]–[23] and random-walk-driven networks [24], [25]. For generic networks, we propose a derivation of the Markov matrix based on the consensus algorithm [21]. DDs are part of the diffusion-maps (DMs) framework introduced in [26]. DMs are applicable to datasets composed by states of high-dimensional data points. These data points can be interpreted as data states as they vary with time. As a function of the Euclidean distance between data points, a Markov matrix can be defined by describing transition probabilities of a random walk between data states. Here, the set of data states can be represented by the nodes of a graph. Note that each node of the graph is associated with an entire data state of the network, such that edges define transition relations between

The work of V. R. M. Elias was supported, in part, by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) -– Grant number: 88887.310189/2018-00. The work of Dr. Martins was supported, in part, by the ERC project AGNOSTIC and, in part, by the Brazilian funding agency Faperj. The work of Dr. Werner was supported in part by the Research Council of Norway and Academy of Finland under Grant 296849.

Vitor R. M. Elias and Stefan Werner are with the Department of Electronic Systems, NTNU-Norwegian University of Science and Technology, Trondheim 7491, Norway (e-mail: vitor.elias@ntnu.no and stefan.werner@ntnu.no).

Wallace A. Martins is with the Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, L-1855 Luxembourg (e-mail: wallace.alvesmartins@uni.lu).

Vitor R. M. Elias and Wallace A. Martins are also with the UFRJ-Federal University of Rio de Janeiro, Rio de Janeiro 21941-972, Brazil.

these high-dimensional data states. This view is in contrast with that of GSP, where edges associate individual network elements. Using eigenvectors of the corresponding Markov matrix, DMs uncover descriptions of the underlying geometry of the dataset [27]–[29]. In this framework, DDs provide a metric for relating two states of data according to the random walk.

We consider elements of the network as nodes of a graph, as in the GSP framework, and the relations between these elements depend on a concept of transition-probability distance, as in the DM framework. The use of DDs yields an augmented version of the initial adjacency model. For instance, DDs relate nodes that are beyond the local reach of the physical neighborhood. This relation depends on how many transition steps of the Markov chain are considered in the computation of the DD. The resulting GSO, called extended-adjacency matrix, depends on the number of transition steps. We demonstrate the benefits of the extended adjacency by implementing a method that uses the proposed model together with the GFT to get a scale-dependent graph-frequency analysis, which we call scale-dependent graph Fourier transform (sGFT). We note that the proposed GSO model can be used with other GSP tools, including other graph-frequency representations. Moreover, the proposed mapping is not restricted to networks that inherently present the Markov property; indeed, it is possible to derive a Markov chain from a generic adjacency matrix as we show in Section V-A.

The combination of DM and GSP has been considered in [30] and [31]. The work in [30] proposes the use of Markov matrices as GSO. In the context of GSP, Markov matrices, of the form in [30], have desirable properties, e.g., they are diagonalizable and the inverse eigenvector matrix can be computed efficiently. The use of Markov matrices also allows DM-related tools, such as dimensionality reduction and clustering, to be incorporated in GSP [30]. Furthermore, the work in [30] studies the similarities between both frameworks, making explicit how some operations from GSP can be interpreted from a DM perspective. For instance, both graph-shifting and graph-filtering operations can be written in terms of embeddings from the DM framework. In [31], a method for graph-signal interpolation is derived using the Nyström extension [32] when employing a Markov matrix as GSO.

The use of DM for classical digital signal processing (DSP) tasks has been studied in different applications in [29]. The authors introduce two filtering schemes that leverage on properties of DMs: *non-local filtering* updates a state $\mathbf{x}_i$ according to the affinity between $\mathbf{x}_i$ and other states $\mathbf{x}_j$, while this affinity is computed by using the Gaussian kernel over the DD between $\mathbf{x}_i$ and $\mathbf{x}_j$; and *graph-based processing* explores subsets of eigenvectors acquired by DMs to extract the desired component of noisy data states. Moreover, the authors in [29] present applications of DMs in single-channel source localization and in the suppression of transient interference for speech enhancement.

In contrast to previous works, we incorporate DDs into the GSP framework and construct a graph model that captures the interaction between elements of the Markov network. Specifically, while the work in [30] proposes a Markov matrix as GSO, we use a Markov matrix only as the starting point of our work. Moreover, the Markov matrix proposed in [30] is not symmetric, whereas we adopt a doubly-stochastic matrix based on the discrete-time consensus algorithm [21]. The proposed model is derived by further implementing concepts of DMs given the initial Markov matrix. Namely, we use the Markov matrix to compute DDs between nodes and generate additional edges. The main contributions of this paper are as follows:

- We propose an extended-adjacency matrix that captures dependencies between non-adjacent nodes of the graph. The model augments the original adjacency using DDs between nodes. We show that the extended-adjacency matrix can be derived for a non-Markov network for which an associated Markov model can be constructed.
- We present a scale-dependent graph Fourier transform (sGFT), as a function of the extended-adjacency matrix, that describes the frequency content of the graph signal. The sGFT reveals the graph frequency versus time, or scale, of the associated Markov chain. The sGFT is applied for anomaly detection using synthetic and real data, and we show that the proposed GSO improves the GFT in the anomaly-detection task when compared to other GSO models.

The rest of the paper is organized as follows. Sections II and III present the fundamentals of graph signal processing and diffusion maps. Section IV introduces the proposed extended-adjacency matrix and scale-dependent graph Fourier transform. In Section V we present numerical experiments that validate the proposed methodology. In Section VI we consider the application of the scale-dependent graph Fourier transform to the problem of anomaly detection in sensor networks. Finally, conclusions are given in Section VII.

II. Graph Signal Processing

This section introduces the notation adopted throughout this work and some fundamental concepts of GSP.

A. Graphs and network modeling

Let a graph be represented by $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{v_1, \dots, v_N\}$ denotes the set of vertices (or nodes) and $\mathcal{E} = \{e_{11}, \dots, e_{NN}\}$ denotes the set of edges. Each vertex corresponds to one element of the network [3], [4], [6], [10], [19], [20], [33]. Each edge e_{ij} represents a pairwise connection between nodes v_i and v_j. Edges reflect the relation between elements of the original structure, if this relation exists. Nodes connected by an edge are adjacent nodes. A mapping $w : \mathcal{E} \to \mathbb{C}$ is used to model weighted edges, such that w_{ij} denotes the weight value for edge e_{ij}. A graph is often represented by the *adjacency matrix* $\mathbf{A} \in \mathbb{C}^{N \times N}$, whose (i, j)th element is $A_{ij} = w_{ij}$.

The set of vertices that are adjacent to a vertex v_j is referred to as the neighborhood of v_j. The set $\mathcal{N}_j$ comprises the indexes of vertices in the neighborhood of v_j. For an undirected graph, where $w_{ij} = w_{ji}$, the weighted degree of node v_j is given by $\deg(v_j) = \sum_{i \in \mathcal{N}_j} w_{ij}$ and we define the diagonal *degree matrix* $\mathbf{D}$, such that $D_{jj} = \deg(v_j)$. The graph *Laplacian matrix* of an undirected graph is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$.

B. Signals over graphs and GSP

Data collected from or generated by elements of the networks can be viewed as a *graph signal*. A complex-valued graph signal is given by the mapping $s : \mathcal{V} \to \mathbb{C}$. We represent a graph signal as a vector $\mathbf{s} \in \mathbb{C}^N$, where the ith entry s_i is given by $s(v_i)$, i.e., the signal component at vertex v_i. The graph signal represents a snapshot of the network state.

The graph shift operator (GSO), denoted by $\mathbf{S}$, is an $N \times N$ matrix employed to generate the graph-shifted signal $\tilde{\mathbf{s}} = \mathbf{S}\mathbf{s}$. That is, the graph-shifted signal on node v_i is a combination of signals s_j, given by $\tilde{s}_i = \sum_{j=1}^{N} S_{ij} s_j$. One choice of $\mathbf{S}$, presented in [10], is the adjacency matrix $\mathbf{A}$. This choice is partly motivated by direct analogy with discrete-time processing of periodic signals. In this case, the resulting graph-shifted signal at node v_i is a local combination of the signal in its neighborhood.

Another popular choice for $\mathbf{S}$ is the graph Laplacian, $\mathbf{L}$, which is a local difference operator. The choice is motivated by graph spectral theory [34]. Motivations and applications using the graph Laplacian as GSO are thoroughly reviewed in [6]. As we show in Section II-C, the eigenvectors of the GSO compose the Fourier basis in graph domain. This construction of the Fourier basis has a particular link with conventional DSP, since the classical Fourier transform may be interpreted as the expansion of a continuous-time function in terms of complex exponentials, which are the eigenfunctions of the one-dimensional Laplace operator.

The aforementioned approaches yield local operators, i.e., $S_{ij} > 0$, for $i \neq j$, if and only if $A_{ij} \neq 0$ [6], [10]. We note that other matrices than $\mathbf{A}$ or $\mathbf{L}$ can be used as GSO, and the choice of the GSO depends on the application at hand [12], [13]. In this work, we propose non-local GSOs that extend the initial adjacency relations between nodes. Once a shift is defined, many results and techniques from conventional DSP theory can be extended to graph domain, e.g., convolution, filtering, transforms (Fourier and wavelets), and spectral analysis.

C. Graph Fourier transform

We assume a graph with real-valued weighted edges. Consider the diagonalizable GSO matrix $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$, where $\mathbf{\Lambda}$ is a diagonal matrix whose entries are the eigenvalues λ_i of $\mathbf{S}$, and $\mathbf{U}$ has as columns the eigenvectors $\mathbf{u}_i$ of $\mathbf{S}$. The GFT coefficients of a graph signal $\mathbf{s}$ are obtained from the analysis equation [6], [10]

$$\hat{\mathbf{s}} = \mathbf{U}^{-1}\mathbf{s}, \tag{1}$$

and the graph signal is recovered from the synthesis equation

$$\mathbf{s} = \mathbf{U}\hat{\mathbf{s}}. \tag{2}$$

The eigenvalues λ_i of $\mathbf{S}$, with $i \in \{1, \ldots, N\}$, correspond to the graph-frequency spectrum. The eigenvectors $\mathbf{u}_i$, with $i \in \{1, \ldots, N\}$, are the graph-frequency components [6], [10], [35]. In this work, we adopt the graph Laplacian as GSO, i.e., we set $\mathbf{S} = \mathbf{L}$. We show next that larger eigenvalues of the graph Laplacian correspond to higher graph frequencies. For this purpose, consider a variation metric for signal $\mathbf{x} \neq \mathbf{0}$ on $\mathcal{G}$ given by

$$\begin{aligned}\nu(\mathbf{x}) &= \frac{\mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x}}{\mathbf{x}^{\mathrm{T}}\mathbf{x}} && (3)\\ &= \frac{\mathbf{x}^{\mathrm{T}}\big(\mathbf{D} - \mathbf{A}\big)\mathbf{x}}{\mathbf{x}^{\mathrm{T}}\mathbf{x}} \\ &= \frac{\sum\limits_{i \neq j} A_{ij}\big(x_i - x_j\big)^2}{\mathbf{x}^{\mathrm{T}}\mathbf{x}}, && (4)\end{aligned}$$

which measures the total difference between the signal values on different vertices, weighted by the edge values. Equation (3) is the Rayleigh quotient of $\mathbf{L}$, which is bounded below and above by the extreme eigenvalues of $\mathbf{L}$, $\lambda_1 = 0$ and λ_N, respectively. As the GFT is defined as the expansion of a signal over the eigenvectors of $\mathbf{L}$, λ_2 and λ_N correspond, respectively, to the smallest and the largest non-zero variations, or graph frequencies. The eigenvalue $\lambda_1 = 0$ corresponds to frequency equal to zero, associated with a constant graph signal. The eigenvalue λ_2 is called *graph spectral gap*, and λ_N is called *graph spectral radius*.

As indicated by (4), the graph frequencies provide information on how fast a signal varies across the vertices. In this context, the signal is a single snapshot of the network state. High frequency means that the signal sample on a given vertex differs considerably from samples on neighboring vertices. Low frequency means that the graph signal is smooth across all nodes. Here, we highlight a fundamental motivation for our work: the adjacency model of a network directly affects its graph-frequency analysis, given the definition of the graph spectrum and its dependency on the elements of the adjacency matrix, as shown in (4). However, this implication of (4) is often neglected when adjacency models are constructed. Thus, we aim for a model that is capable of capturing node relations while taking into account its influence on GSP applications.

III. Diffusion Maps

This section introduces the basics of diffusion maps (DMs) and diffusion distances (DDs) [26], necessary for the development of the proposed extended adjacency and sGFT.

Let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_K] \in \mathbb{R}^{L \times K}$ be a data matrix with K data points, also called states, each of dimension L. For example, matrix $\mathbf{X}$ can describe the evolution of the state $\mathbf{x}_k$ of a network with L elements for time instants $k \in \{1, \ldots, K\}$. It is assumed that there is an underlying (hidden) process that relates the different data points, possibly driving the way data is generated. However, note that there is no underlying graph associated with $\mathbf{X}$. The objective of the DM framework is to make this underlying process explicit [26], [29], [30].

A. Construction of the similarity graph

The first step when constructing a DM is to create a graph that associates data points $\mathbf{x}_i$ with nodes, and quantifies their interrelationship [26]. In contrast to the GSP framework, this association is merely an alignment of the data with the structure. That is, the data point is not treated as a signal on the node, but rather as the node itself. In order to make the contrast between the GSP and DM approaches clear, we highlight that, throughout this section, the number of data states K corresponds to the number of nodes in the graph constructed for the DM framework.

Let $\mathcal{X} \subset \mathbb{R}^L$ represent the dataset that contains the columns of $\mathbf{X}$. The edges are created through a symmetric kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$, i.e., $k(\mathbf{x}, \mathbf{y}) = k(\mathbf{y}, \mathbf{x}) \geq 0$. The obtained graph is undirected and possibly weighted, and serves as a preliminary geometric description of the data based on the underlying driving process [26]. The choice of the kernel depends on the application. For instance, a common choice is the radial-basis-function (RBF) kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma_{\text{RBF}}^2}\right), \tag{5}$$

where $\sigma_{\text{RBF}} > 0$ is a free parameter that controls the bandwidth of the kernel.

The RBF kernel expresses a relationship based on the affinity between data points $\mathbf{x}_i$ and $\mathbf{x}_j$, in terms of the Euclidean distance. Once a metric for similarity is established for the data points, an adjacency matrix $\mathbf{A} \in \mathbb{R}^{K \times K}$ can be defined with entries $A_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. The corresponding degree matrix is $\mathbf{D} \in \mathbb{R}^{K \times K}$ such that $D_{ii} = \sum_{j \in \mathcal{N}_i} k(\mathbf{x}_i, \mathbf{x}_j)$.

B. Construction of the random walk

To capture how data is influenced by an underlying process, a random walk on the data is defined. The idea is to characterize how one state of the high-dimensional data transitions into another state [26]. For this purpose, the similarity between two data points is normalized as [26]

$$p(\mathbf{x}_j|\mathbf{x}_i) = \frac{k(\mathbf{x}_i, \mathbf{x}_j)}{\sum\limits_j k(\mathbf{x}_i, \mathbf{x}_j)} = \frac{A_{ij}}{D_{ii}}, \tag{6}$$

where $p(\mathbf{x}_j|\mathbf{x}_i)$ is interpreted as the transition probability from $\mathbf{x}_i$ to $\mathbf{x}_j$, which establishes a Markov chain. Using matrix notation, the Markov chain can be described in terms of a right-stochastic matrix $\mathbf{M} = \mathbf{D}^{-1}\mathbf{A}$, commonly referred to as a Markov matrix, with entries $M_{ij} = p(\mathbf{x}_j|\mathbf{x}_i)$. Taking t steps of the random walk is captured by $\mathbf{M}^t$, i.e., the (i, j)th entry of $\mathbf{M}^t$ gives the transition probability, denoted by $p_t(\mathbf{x}_j|\mathbf{x}_i)$, from $\mathbf{x}_i$ to $\mathbf{x}_j$ in t steps. The probability $p_t(\mathbf{x}_j|\mathbf{x}_i)$ considers all possible paths composed of t edges that connect $\mathbf{x}_i$ to $\mathbf{x}_j$, including self-loops. The probability in (6) is the same as $p_t(\mathbf{x}_j|\mathbf{x}_i)$ for $t = 1$.

Consider the decomposition of $\mathbf{M}$ in terms of its right and left eigenvectors $\boldsymbol{\psi}_k$ and $\boldsymbol{\phi}_k$ and the eigenvalues γ_k, with $k \in \{1, \ldots, K\}$.[1] The transition probabilities can be written as [26]

$$p_t(\mathbf{x}_j|\mathbf{x}_i) = \sum_{k=1}^{K} \gamma_k^t \psi_{k,i} \phi_{k,j}. \tag{7}$$

The eigenvalues of a right-stochastic matrix satisfy $|\gamma_k| \leq 1$. Assuming the graph is connected, the Markov chain is irreducible and $\gamma_1 = 1$ [26]. We adopt the ordering $\gamma_1 = 1 > |\gamma_2| \geq \cdots \geq |\gamma_K|$. Consequently, a random walk driven by these transition probabilities has an asymptotic behavior governed by γ_1 [26], i.e.,

$$\lim_{t \to \infty} p_t(\mathbf{x}_j|\mathbf{x}_i) = \phi_{1,j}, \tag{8}$$

where $\phi_{1,j}$ is the jth entry of the first left eigenvector of $\mathbf{M}$, $\boldsymbol{\phi}_1$, normalized as $\|\boldsymbol{\phi}_1\|_1 = 1$. In other words, $\phi_{1,j}$ is the asymptotic probability of reaching state $\mathbf{x}_j$ from any initial state. As the graph is connected, this quantity is non-zero for every j, as given by (7). Note that the right eigenvector $\boldsymbol{\psi}_1$, associated with $\gamma_1 = 1$, is a constant vector, as $\mathbf{M} \cdot \mathbf{1} = \mathbf{1}$.

C. Diffusion distances

The *diffusion distance* (DD) at a certain diffusion, or time, scale t [26] is a metric for the (inverse) affinity between two data points as a function of transition probabilities, and is given by

$$D_t^2(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{K} \frac{\left(p_t(\mathbf{x}_k|\mathbf{x}_i) - p_t(\mathbf{x}_k|\mathbf{x}_j)\right)^2}{\phi_{1,k}}. \tag{9}$$

The DD extends local relations, in terms of adjacency and direct similarity between nodes, into a global metric by assimilating probabilities of diffusion paths [27]. If two points have similar posterior distributions, they are well connected through the end-points, indexed by k, of these distributions. This means that if there are high-probability paths between two data points, they are considered to be close in terms of diffusion distance, even if not adjacent. Conversely, the diffusion distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ is large when the probability of reaching $\mathbf{x}_j$ from $\mathbf{x}_i$ is small even by considering non-direct paths through $\mathbf{x}_k$. An alternative interpretation for the diffusion distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ is to consider a limited amount of energy placed on both nodes. This energy is then diffused through the network for t diffusion steps according to $\mathbf{M}^t$. Finally, the distance is computed as the difference between the contributions from these two nodes to the network, in the sense of how much energy was diffused to each node. If the two points have similar contribution to a given node, it means that they are well connected through that node. Note that the diffusion distance depends on t, which serves as a diffusion-scale parameter. An increased t equals more steps of the random walk, which corresponds to a larger-scale diffusion over the network.

IV. GSP for Markov Networks

The DM framework in [26] treats the entire network signal as a state of a Markov chain and generates a Markov-based graph over these states. In contrast, we adapt the diffusion distances to develop a comprehensive network model by treating the original network itself as the Markov-based graph. That is, the states of the Markov chain are no longer associated with the data (entire network state), but with the vertices (individual network elements).

We introduce the extended-adjacency matrix, which captures node collaboration of Markov networks. Some common cases of these networks are consensus networks [21]–[23],

[1] Not every Markov matrix is diagonalizable, but $\mathbf{M} = \mathbf{D}^{-1}\mathbf{A} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}})\mathbf{D}^{\frac{1}{2}}$ is, for it is similar to the symmetric matrix $\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$.

conservative diffusion networks [18], [36], [37], and random-walk driven networks [24], [25]. Moreover, we introduce the sGFT and analyze the spectral behavior of the graph through the perspective of the extended-adjacency model.

A. Extended adjacency

The networks considered here are initially constrained by adjacency rules of the network topology. For example, the connections between nodes in a wireless sensor network (WSN) depend on their communication capabilities, usually dictated by physical distance. Furthermore, the sum of the weights of the edges connecting a node to its neighbors is equal to unity, yielding a stochastic adjacency. Nodes communicate directly with their immediate neighbors. However, nodes that are not initially connected can be related to each other through collaboration. In other words, network adjacency is associated with one step of the collaboration process, such that the network operates on the data through iterative multiplications of the stochastic matrix. Similarly to the conventional adjacency, which depends on the physical distance between nodes, we develop the extended adjacency based on diffusion distances.

Let the network be represented by a connected graph $\mathcal{G} = \{\mathcal{V}, \mathbf{B}\}$ with a symmetric, irreducible, and stochastic adjacency matrix $\mathbf{B}$ with positive real edges. In analogy with the theory presented in Section III, this matrix is equivalent to $\mathbf{M}$ and the states of the Markov chain are the nodes of the graph. The diffusion distance between nodes v_i and v_j of the graph is given by

$$D_t^2(v_i, v_j) = \sum_{n=1}^{N} \frac{\left(B_{in}^{(t)} - B_{jn}^{(t)}\right)^2}{(1/N)}, \tag{10}$$

where $(1/N)$ corresponds to the elements of the first left eigenvector $\mathbf{q}_1$ of $\mathbf{B}$, as in (9), and $B_{ij}^{(t)}$ denotes the (i, j)th entry of $\mathbf{B}^t$. Since we assume $\mathbf{B}$ symmetric, its left and right eigenvectors are the same and $\mathbf{q}_1 = (1/N)\mathbf{1}$. Similar to the DM framework, the DDs between network elements depends on the scale t. The metric in (10) expresses distances between nodes, including nodes that are not neighbors as defined by $\mathbf{B}$. We derive similarity from diffusion distances in similar manner as conventional adjacency matrices are derived from geographic distances. The extended-adjacency matrix $\bar{\mathbf{A}}(t)$ is such that

$$\bar{A}_{ij}(t) = \begin{cases} B_{ij} + \exp\left(-\frac{D_t^2(v_i,v_j)}{\rho N}\right) & i \neq j \\ 0 & i = j, \end{cases} \tag{11}$$

where $\rho > 0$ is a free parameter and N is the size of the network. The term B_{ij} in (11) guarantees that original edges are maintained, whereas the RBF term is responsible for extending the adjacency. The term ρN makes the argument of the RBF kernel independent of the network size (cf. (10)). The range of the kernel output can be adjusted for different applications according to the free parameter ρ. Moreover, although the extended adjacency is defined for $\rho > 0$, it is possible to obtain B_{ij}, with $i \neq j$, through (11) by making $\rho \to 0^+$. That is, the original adjacency is a particular case of the extended adjacency.

Although traditional GSOs are local with respect to network connections, we note that this property is not present in the proposed adjacency model. A local GSO offers a straightforward visualization of the physical structure of the graph and facilitates the implementation of distributed GSP algorithms. However, in many applications, the definition of locality is unknown, or a local GSO fails to model implicit node relations. In this work, we aim to derive a model that is not restricted by locality assumptions and is useful for networks where non-adjacent nodes interact. Hence, we propose a non-local model that offers a trade-off between locality and representation of node interactions. As we show in the next section, this trade-off is indirectly controlled via the diffusion-scale parameter. In this same context, other desirable features inherent to specific graph structures might be lost after the implementation of the extended adjacency. For instance, the correspondence between the GFT over a ring graph and the conventional discrete Fourier transform (DFT) is lost when the extended-adjacency matrix is considered. Our proposal, however, is aimed at graphs whose topologies are not well-structured or perfectly known and we show that some GSP tools can benefit from the proposed model.

Moreover, the proposed extended-adjacency matrix models a graph with no self-loops. Although this assumption is also common in the GSP framework, we note that it is not necessary. Further modifications of $\bar{\mathbf{A}}(t)$ are possible, such as imposing an upper bound on edge values, or applying a threshold on the matrix values to enforce sparsity.

B. Analysis of the extended-adjacency matrix

The resulting extended-adjacency matrix $\bar{\mathbf{A}}(t)$ is a symmetric matrix with positive entries that depend on the diffusion-scale t and bandwidth ρ.

Proposition 1. *The DD given in* (10) *is a non-increasing function of the diffusion scale t.*

Proof. The symmetric matrix $\mathbf{B}$ can be decomposed as $\mathbf{B} = \sum_{l=1}^{N} \sigma_l \mathbf{q}_l \mathbf{q}_l^{\mathrm{T}}$, where σ_l and $\mathbf{q}_l$, with $l \in \{1, \ldots, N\}$, are the eigenvalues and orthonormal eigenvectors of $\mathbf{B}$. We can write

$$B_{in}^{(t)} - B_{jn}^{(t)} = \sum_{l=1}^{N} \sigma_l^t (q_{l,i} - q_{l,j}) q_{l,n}. \tag{12}$$

Substituting (12) into (10), we have

$$D_t^2(v_i, v_j) = N \sum_{l=1}^{N} \sum_{m=1}^{N} \sigma_l^t \sigma_m^t (q_{l,i} - q_{l,j})(q_{m,i} - q_{m,j}) \zeta_{l,m}, \tag{13}$$

where $\zeta_{l,m} = \sum_{n=1}^{N} q_{l,n} q_{m,n} = \delta(l - m)$ and $\delta(\cdot)$ is the Kronecker delta function. Thus,

$$D_t^2(v_i, v_j) = N \sum_{l=1}^{N} \sigma_l^{2t} (q_{l,i} - q_{l,j})^2. \tag{14}$$

Since $(q_{l,i} - q_{l,j})^2 \geq 0$ and $\sigma_l^2 \in [0, 1]$, then $D_t^2(v_i, v_j) \geq D_{t+1}^2(v_i, v_j)$, $\forall t \geq 1$. □

In accordance with Proposition 1, the following corollary and lemma can be established:

Corollary 1. *Edge weights are non-decreasing with t according to* (11)*. Assuming that an edge exists only if its weight exceeds a given threshold, the number of edges is also non-decreasing with increasing t. In other words, increasing t for a fixed ρ possibly creates new edges, given the reduction in the DD.*

Lemma 1. *The entries of the asymptotic extended-adjacency matrix $\bar{\mathbf{A}} = \lim_{t\to\infty} \bar{\mathbf{A}}(t)$ are given by*

$$\bar{A}_{ij} = \begin{cases} B_{ij} + 1 & i \neq j \\ 0 & i = j, \end{cases} \tag{15}$$

and correspond to those of an adjacency matrix of a complete graph without self-loops. That is, each node v_i is connected to every node v_j in the graph, with $i \neq j$, by an edge of value $1 + B_{ij}$.

Proof of Lemma 1. From (14), $t \to \infty$ implies that the diffusion distance between any two vertices tends to zero, since σ_l^{2t} tends to zero for $l \neq 1$ and the only remaining non-zero term σ_l^{2t} corresponds to $l = 1$, for which $(q_{1,i} - q_{1,j}) = 0$, since $\mathbf{q}_1 = (1/N)\mathbf{1}$. With $D_t^2(v_i, v_j) = 0$, we have $\bar{A}_{ij}(t) = B_{ij} + 1$, $\forall i \neq j$, and $\bar{A}_{ii}(t) = 0$, with $i, j \in \{1, \ldots, N\}$. □

C. Scale-dependent graph Fourier transform

Similar to the extended-adjacency matrix $\bar{\mathbf{A}}(t)$, the definition of other graph-related matrices and the graph Fourier transform also depend on the scale t. For each diffusion scale, there is a corresponding Laplacian matrix defined as

$$\bar{\mathbf{L}}(t) = \bar{\mathbf{D}}(t) - \bar{\mathbf{A}}(t), \tag{16}$$

where $\bar{\mathbf{D}}(t)$ is the diagonal degree matrix associated with $\bar{\mathbf{A}}(t)$. Given the eigendecomposition $\bar{\mathbf{L}}(t) = \bar{\mathbf{U}}(t)\bar{\mathbf{\Lambda}}(t)\bar{\mathbf{U}}^{\mathrm{T}}(t)$, we define the scale-dependent graph Fourier analysis of signal $\mathbf{x}$ as

$$\hat{\mathbf{x}}(t) = \bar{\mathbf{U}}^{\mathrm{T}}(t)\mathbf{x}, \tag{17}$$

where, in contrast to the conventional GFT, the coefficients $\hat{\mathbf{x}}(t)$ depend on the diffusion-scale t. The scale-dependent graph Fourier synsthesis equation is given by

$$\mathbf{x} = \bar{\mathbf{U}}(t)\hat{\mathbf{x}}(t). \tag{18}$$

Remark 1. *Note that the analysis and synthesis equations result from the use of the conventional GFT together with the proposed extended-adjacency model. Thus,* (17) *and* (18) *do not establish a novel transform. However, we will refer to* (17) *as the scale-dependent graph Fourier transform (sGFT) for simplicity throughout the text.*

D. Analysis of the sGFT

Now, we analyze how the proposed adjacency model affects the graph-frequency analysis. Recall that graph frequencies are directly associated with the eigenvalues of the graph Laplacian, as shown in (4). Hence, the analysis is conducted in terms of which graph frequencies are discriminated by the Laplacian eigenvalues for varying diffusion scales. For this purpose, we determine bounds for the spectral gap and spectral radius of the graph in function of t. Let the eigenvalues of the Laplacian be $\bar{\lambda}_1(t)$, $\bar{\lambda}_2(t)$, …, $\bar{\lambda}_N(t)$, in ascending order with $\bar{\lambda}_1(t) = 0$. The spectral gap is $\bar{\lambda}_2(t)$ and the spectral radius is $\bar{\lambda}_N(t)$, and both depend on t. Let θ_2 denote the smallest non-zero eigenvalue of $\mathbf{L_B}$, the Laplacian of $\mathbf{B}$, and θ_N denote the maximum eigenvalue of $\mathbf{L_B}$. Moreover, let $\mathbf{L_C}$ denote the Laplacian of the unweighted complete graph with N nodes, whose entries are given by

$$L_{\mathbf{C}_{ij}} = \begin{cases} -1 & i \neq j \\ N - 1 & i = j. \end{cases} \tag{19}$$

Proposition 2. *For a graph with N nodes, if t is increased, the range of graph-frequencies discriminated by the sGFT shifts into higher frequencies. Asymptotically, the sGFT discriminates graph-frequency ranges up to the interval $[N + \theta_2, N + \theta_N]$.*

Proof. The eigenvalues of the Laplacian of a connected graph are non-decreasing with the addition of new edges in the graph [38]. As follows from Corollary 1, edges can only be added, and not removed, as t increases. Consequently, the spectral gap and radius are non-decreasing for increasing t.

As follows from Lemma 1, the asymptotic extended adjacency $\bar{\mathbf{A}}$ corresponds to that of a complete graph. The resulting graph Laplacian is $\mathbf{L}_{\bar{\mathbf{A}}} = \mathbf{L_B} + \mathbf{L_C}$. Given the structure of $\mathbf{L_C}$ and the fact the $\mathbf{L_B}$ is symmetric, these matrices commute and the eigenvalues of $\mathbf{L_B} + \mathbf{L_C}$ are given by the sums of eigenvalues of each matrix. Eigenvalues of $\mathbf{L_C}$ are: 0 with multiplicity 1 and the eigenvalue N with multiplicity $N - 1$. Consequently, for $\mathbf{L}_{\bar{\mathbf{A}}}$, the spectral gap achieves the value $N + \theta_2$, and the spectral radius achieves $N + \theta_N$. □

As t increases, the interval of graph frequencies discriminated by the sGFT is shifted into higher frequencies. We note that the maximum frequency discriminated by the sGFT, $N + \theta_N$, matches the largest possible variation, according to (4), of signals defined over the proposed adjacency matrix.

The sGFT is a frequency-analysis tool tailored for each stage of the Markov chain. From a node-collaboration perspective, the effect of node collaboration on the graph spectrum can be interpreted in an intuitive manner: if more steps of collaboration are taken, more edges are introduced. Consequently, variations in graph signals are observed by additional node pairs and are perceived as larger frequencies. Hence, by incorporating node collaboration into the graph model, we provide a frequency-analysis tool that reveals more information on the network signal than that offered by the conventional GFT.

Remark 2. *The proposed implementation of the Fourier analysis based on the extended adjacency adds a degree of freedom for tools that use the graph-spectrum. Therefore, some applications may require training data along with a hyperparameter-training method, such as grid-search and cross-validation, for determining the adequate diffusion scale.*

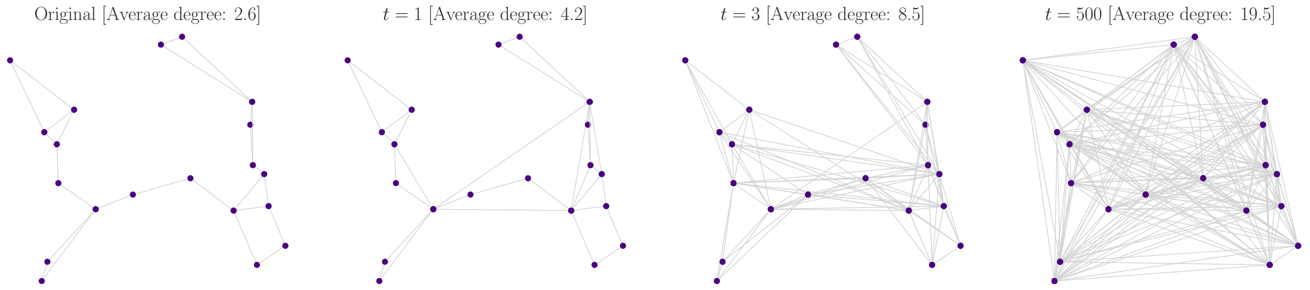

Original [Average degree: 2.6]
$t = 1$ [Average degree: 4.2]
$t = 3$ [Average degree: 8.5]
$t = 500$ [Average degree: 19.5]

Fig. 1. Connectivity versus diffusion scale. Only edges of $\bar{\mathbf{A}}(t)$ that exceed 30% of the highest edge value are shown.

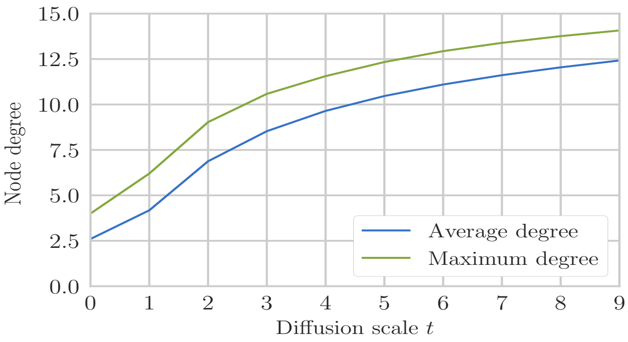


Fig. 2. Average and maximum degrees versus diffusion scale t. For $t = 0$, the values correspond to those of the original network.

V. Numerical Examples

In this section we verify through numerical examples the analyses conducted in Section IV-B and Section IV-D.

A. Nearest-neighbors and consensus network

We consider a sensor network with N sensors that collectively estimate a common parameter through collaborations, more specifically through consensus averaging [21], [22].

In an average consensus algorithm, the global average of initial sensor states, $x_i[0]$, with $i \in \{1, \ldots, N\}$, is computed in a distributed fashion through local computations and local message exchanges. More specifically, sensor v_i implements the following iterative algorithm

$$x_i[k+1] = P_{ii} x_i[k] + \sum_{j \in \mathcal{N}_i} P_{ij} x_j[k], \quad k \in \mathbb{N} \tag{20}$$

where P_{ij} are weights given to local and neighbor node values. In the case of sensor networks, the neighborhood $\mathcal{N}_i$ is usually defined by sensor nodes within the transmission radius of sensor v_i. Equation (20) can be written as

$$\mathbf{x}[k+1] = \mathbf{P}\mathbf{x}[k], \tag{21}$$

where $\mathbf{x}[k] = [x_1(k) \ldots x_N(k)]^{\mathrm{T}}$. If $\mathbf{P}$ is doubly stochastic, then the sensor states will converge to $(1/N)\mathbf{1}^{\mathrm{T}} \cdot \mathbf{x}[0]$.

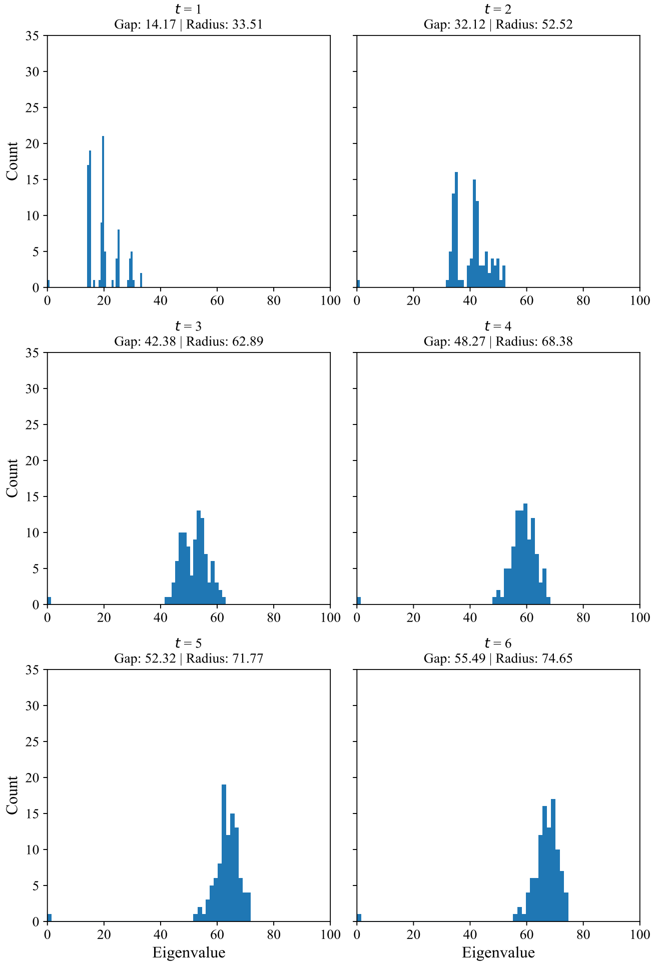


Fig. 3. Histogram of eigenvalues of diffusion Laplacian matrices $\bar{\mathbf{L}}(t)$, for $t \in \{1, \ldots, 6\}$.

Let the network be modeled by an unweighted and undirected graph $\mathcal{G} = \{\mathcal{V}, \mathbf{A}\}$ with graph Laplacian $\mathbf{L}$. One particular form of $\mathbf{P}$, which leads to Laplacian based consensus, is given by

$$\mathbf{P} = \mathbf{I} - \epsilon \mathbf{L}, \tag{22}$$

where ϵ is the consensus step size. In this work, we set $\epsilon = 1/(1.25\Delta_{\max})$, where $\Delta_{\max}$ is the maximum degree in $\mathcal{G}$ (more information on convergence of consensus algorithms and the choice of ϵ can be found in [21]). Thus, we are able to associate the network with a stochastic matrix $\mathbf{P}$, since $\mathbf{P} \cdot \mathbf{1} = \mathbf{1} - \epsilon \mathbf{L} \cdot \mathbf{1} = \mathbf{1}$.

B. Increasing connectivity

Given matrix $\mathbf{P}$, the diffusion distances between sensors, with $\mathbf{B} = \mathbf{P}$, and the extended-adjacency matrix $\bar{\mathbf{A}}(t)$ are given by (10) and (11), respectively. We consider a sensor network with $N = 20$ sensors, with an average node degree equal to 2.6, and $\rho = 0.35$, Fig. 1 shows the connections of the original network and the new edges introduced at larger diffusion scales. As expected, the network becomes more connected tending to a complete graph as t grows large. Fig. 2 shows the increase in average and maximum degrees versus diffusion scale. This experiment illustrates the effects of changing diffusion scales presented in Proposition 1 and Lemma 1.

C. Spectrum analysis

For a network with $N = 100$ nodes, we conduct an experimental analysis of the sGFT spectrum for different diffusion scales, while the DDs are computed for $\rho = 0.4$. Fig. 3 shows histograms of the eigenvalues of the graph Laplacian $\bar{\mathbf{L}}(t)$ versus diffusion scale. An increase in the diffusion scale yields an increase in the spectral gap (Proposition 2), from around 14 for $t = 1$ up to around 55 for $t = 6$, and in the spectral radius, from around 34 up to over 74. At each scale, the sGFT yields a different spectrum according to the number of steps of node collaboration. In contrast, the conventional GFT yields a fixed spectrum. The additional information provided by the sGFT can benefit applications that make decisions based on spectrum-related features, such as classifiers and detectors.

Numerical results indicate that the new eigenvectors $\bar{\mathbf{u}}_i(t)$, with $i \in \{1, \dots, N\}$, virtually preserve the notion of smoothness with respect to the initial Laplacian matrix $\mathbf{L}$. For this analysis, consider $\mathbf{v} = [\mathbf{u}_1^{\mathrm{T}}\mathbf{L}\mathbf{u}_1 \dots \mathbf{u}_N^{\mathrm{T}}\mathbf{L}\mathbf{u}_N]$, where $\mathbf{u}_i$, with $i \in \{1, \dots, N\}$, are the eigenvectors of $\mathbf{L}$, and $\bar{\mathbf{v}}(t) = [\bar{\mathbf{u}}_1(t)^{\mathrm{T}}\mathbf{L}\bar{\mathbf{u}}_1(t) \dots \bar{\mathbf{u}}_N(t)^{\mathrm{T}}\mathbf{L}\bar{\mathbf{u}}_N(t)]$. We assess the cosine similarity between $\bar{\mathbf{v}}(t)$ and $\mathbf{v}$ given by

$$\mathrm{c}_{\bar{\mathbf{v}}(t)\mathbf{v}} = \frac{(\bar{\mathbf{v}}(t) - \mu_{\bar{\mathbf{v}}(t)})^{\mathrm{T}}(\mathbf{v} - \mu_{\mathbf{v}})}{\|\bar{\mathbf{v}}(t) - \mu_{\bar{\mathbf{v}}(t)}\|_2 \|\mathbf{v} - \mu_{\mathbf{v}}\|_2}, \tag{23}$$

where $\mu_{\mathbf{v}}$ denotes the mean of $\mathbf{v}$. For a total of 1000 random graphs, with N in the interval $[20, 200]$ and number of neighbors in the interval $[2, 16]$, we obtain an average value for $\mathrm{c}_{\bar{\mathbf{v}}(1)\mathbf{v}}$ of 0.92 when $t = 1$. This value decreases slightly as the scale increases: $\mathrm{c}_{\bar{\mathbf{v}}(2)\mathbf{v}} = 0.90$ and $\mathrm{c}_{\bar{\mathbf{v}}(3)\mathbf{v}} = 0.86$.

VI. Application

In this section, we illustrate how the sGFT can be used for anomaly detection in synthetic and real networks. The application is motivated by increasing connectedness of real-world elements [39]–[42], which demands security and reliability in networks [43]–[49]. The free parameters of the sGFT allow for a tailored frequency decomposition when constructing an anomaly detector based on spectral information of the network state. We compare results from detectors based on the sGFT with detectors based on other different GSO approaches.

Given an initial adjacency matrix $\mathbf{A}$, the proposed method is compared to the GFT based on the eigenvectors of the corresponding Laplacian and to the GFT using the eigenvectors of the Markov GSO as proposed in [30]. Additionally, a scale-dependent GSO model based on shortest-path distances is implemented and used for comparison. This GSO depends on the length of the shortest-paths as follows: for $\mathbf{A}$, the shortest-paths and corresponding distances from each node to the rest of the graph are computed using Dijkstra's algorithm [52]. Once the paths are computed, a GSO can be constructed by connecting each node to other nodes reachable through shortest-paths no longer than a given length, with edge-weights equal to the inverse of the shortest-path distances. Note that, given an unweighted adjacency matrix, the GSO based on shortest-paths for lengths up to 1 is equal to $\mathbf{A}$.

In the following simulation results and figures, the GSO approaches are coded as follows:

- GFT: conventional GFT using the graph Laplacian;
- DF1: sGFT using scale $t = 1$;
- DF2: sGFT using scale $t = 2$;
- SP2: GFT using shortest-path-based GSO with paths up to length equal to 2 hops;
- SP3: GFT using shortest-path-based GSO with paths up to length equal to 3 hops;
- MRK: GFT using the eigenvectors of the Markov matrix from [30].

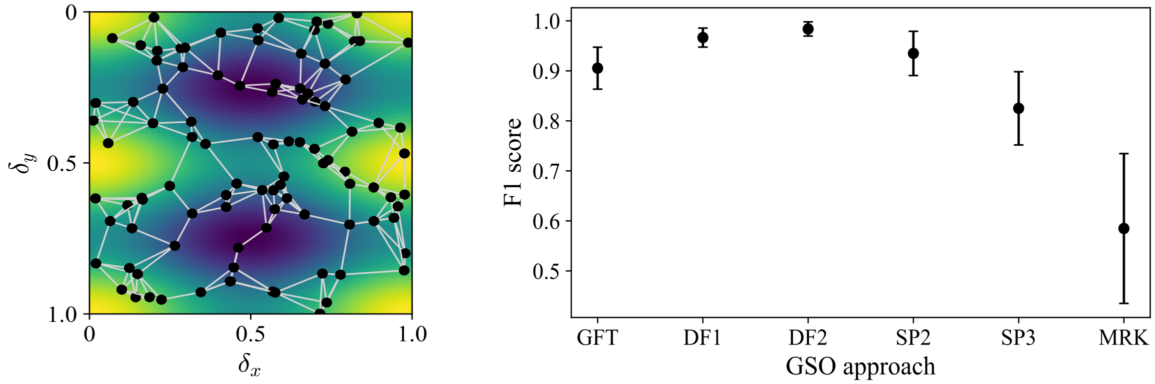
A. Anomaly detection task

We construct classifiers based on the graph-spectral information generated by the sGFT and the conventional GFT along with the aforementioned GSO-construction methods. The approach for constructing the anomaly detector is similar to those in [35], [50], [51]. In particular, [35] and [50] compute the respective high-frequency components from the adjacency matrix and the graph Laplacian. On the other hand, in [51], the Laplacian eigenvectors are obtained from a constrained optimization problem that enforces properties not considered here, such as sparsity. More specifically, assuming that smoothness is expected in the healthy signal, we apply a high-pass filter with cut-off frequency λ_{cut} and conduct the classification based on the filtered coefficients. If one of the coefficients exceeds a threshold τ, the signal is classified as anomalous. Consider a training dataset $\mathcal{X}_{\mathrm{train}} = \{\mathcal{X}_{\mathrm{H}}, \mathcal{X}_{\mathrm{A}}\}$, where $\mathcal{X}_{\mathrm{H}}$ and $\mathcal{X}_{\mathrm{A}}$ indicate the healthy and anomalous parts of the training dataset, respectively, with "healthy" indicating a signal free from anomalies. Elements of these sets are graph signals, that is, vectors with length equal to the number of nodes. The detection threshold is determined as follows:

1) graph-frequency coefficients are computed for each healthy signal in $\mathcal{X}_{\mathrm{H}}$;
2) high-pass filter is applied, so that coefficients corresponding to graph-frequencies higher than λ_{cut} are kept;
3) for each signal, we select a partial τ_{p} that corresponds to the largest coefficient after filtering;
4) Once τ_{p} is computed for all signals, the detection threshold is computed as:

$$\tau = \mu_{\tau_{\mathrm{p}}} + \alpha\sigma_{\tau_{\mathrm{p}}}, \tag{24}$$

where $\mu_{\tau_{\mathrm{p}}}$ is an estimation of the average value and $\sigma_{\tau_{\mathrm{p}}}$ is an estimation of the standard deviation of all partial



(a)

(b)

Fig. 4. Experiment 1 – setup and results: (a) spatial distribution of the sensors and their interconnections plotted over a snapshot of the observed signal; and (b) f1-scores achieved by each of the GSO-construction approaches.

τ_p computed. The non-negative parameter α scales a confidence factor associated with the standard deviation of the partial thresholds.

Using the conventional GFT, with the graph Laplacian, and the GFT from the Markov matrix, λ_cut and α are the parameters that require training. The scale-dependent approaches, however, have more free parameters: the path length for the shortest-path-based approach, and both the diffusion scale t and the normalization parameter ρ for the sGFT. Two different diffusion scales $t \in \{1, 2\}$ are tested separately, as well as two maximum path lengths equal to 2 and 3. Parameters λ_cut, α, and ρ are optimized via grid-search. That is, a set of pre-determined values is given for each parameter and all combinations are tested according to some metric. We choose the f1 score as metric, which is given by

$$\mathrm{f1} = \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}, \tag{25}$$

where

$$\mathrm{precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{26}$$

and

$$\mathrm{recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{27}$$

with TP indicating true positives (correctly classifying anomaly as anomaly), FP indicating false positives (wrongly classifying healthy signal as anomaly), and FN indicating false negatives (wrongly classifying anomaly as healthy signal). Moreover, we conduct cross-validation with 5 folds, such that the training data is split into 5 sets and, for each combination of parameters, the training (computing τ) is performed over 4 sets and the metric is evaluated over the remaining set. The combination of parameters yielding the best average result across the folds is selected. Our simulations are conducted in Python and we use the *GridSearchCV* class (which performs the grid-search with cross-validation) from the *scikit-learn* library to train the classifier.

Once the classifiers are trained, their performance is assessed by measuring the f1 score achieved over a test dataset $\mathcal{X}_\mathrm{test}$. All simulation results presented in the next subsections are averaged over 50 random training and test datasets, which are specified according to each simulation.

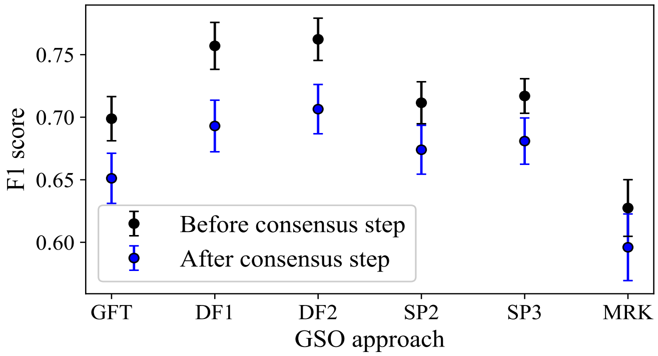
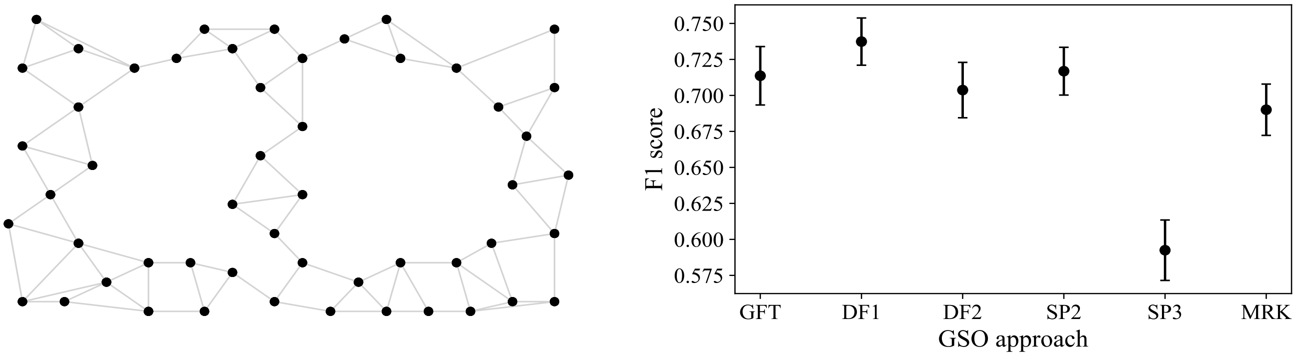


Fig. 5. F1 scores achieved by each GSO approach for the two cases of Experiment 2.

B. Simulations over synthetic networks

1) Experiment 1: spatially-spread anomaly: We consider first $N = 100$ sensors randomly distributed in the square space $[0, 1] \times [0, 1]$. A network is constructed by connecting each sensor to its 4 nearest-neighbors, and the corresponding adjacency matrix $\mathbf{A}$ is known. The sensors measure a spatially-smooth wave signal given by $s(\delta_x, \delta_y) = \cos(2\pi\delta_x + \theta_x) + \cos(2\pi 2\delta_y + \theta_y)$, where $\delta_x, \delta_y \in [0, 1]$ are, respectively, the horizontal and vertical spatial coordinates and θ_x and θ_y are varying phase values uniformly and independently sampled from $[0, 2\pi]$. The graph structure and a snapshot of the signal s are depicted in Fig. 4a.

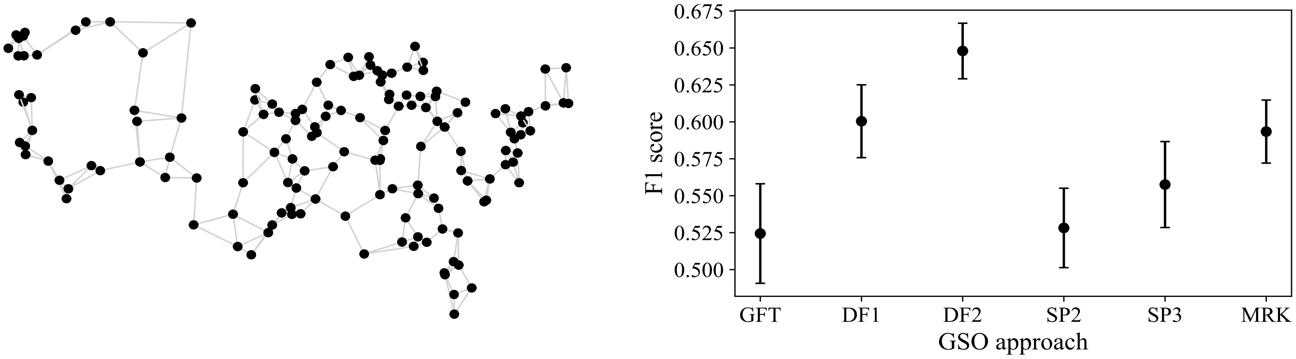
We consider the problem of detecting an additive (space-wise) high-frequency interference signal given by $s_i(\delta_x, \delta_y) = 0.1\,(\cos(2\pi 5\delta_x + \theta_x) + \cos(2\pi 6\delta_y + \theta_y))$. Training and test datasets, $\mathcal{X}_\mathrm{train}$ and $\mathcal{X}_\mathrm{test}$, have 150 healthy samples and 150 anomalous samples each. Results for the f1 score achieved over the 50 independent runs are presented in Fig. 4b. Results show that the detector based on the spectral information provided by the extended-adjacency matrices outperform detectors based on other GSO approaches. Additionally, using



(a)

(b)

Fig. 6. Experiment 3 – setup and results: (a) graph structure for the Intel Lab dataset; and (b) f1 scores achieved by each of the GSO-construction approaches.



(a)

(b)

Fig. 7. Experiment 4 – setup and results: (a) graph structure for the GSOD dataset; and (b) f1 scores achieved by each of the GSO-construction approaches.

the Markov matrix alone offered worse results than using the Laplacian of the original adjacency matrix.

2) Experiment 2: anomaly/attack on few sensors: Consider now a network with $N = 150$ sensors in the same square space $[0,1] \times [0,1]$, each connected to its 6 nearest neighbors, that measure healthy signals $\mathbf{x} \sim \mathcal{N}(20 \cdot \mathbf{1}, 0.4 \cdot \mathbf{I})$. Up to two sensors are randomly selected as anomalous, i.e., the network will have one or two out of 150 sensors with anomalous data. This selection is conducted independently for each snapshot, such that the anomalous sensors of a given graph signal do not depend on the anomalous sensors of other signals. Each anomalous sensor measures a random value $20 \pm b$, with b drawn uniformly from $\{1,2,3,4,5\}$ and independently for each sensor and each signal. The possible values for b are heuristically chosen so that the anomaly is sufficiently strong to be detected, but still not be trivially detected, by all approaches.

We assume the anomaly is present in the initial data measured by the network and that the network performs consensus over the data according to (20). This makes the anomalies smoother as they are diffused according to the consensus algorithm. We conduct the anomaly-detection task before the consensus step is taken, and also after the consensus step is taken. These simulations are independent of each other. For each case, training and test datasets have 200 data samples each, of which 100 are anomalous samples, and experiments are run for 50 independent randomizations of these datasets.

Fig. 5 shows results for the f1 scores obtained by the best classifiers over the test datasets. Results show that the sGFT achieves better detection scores than using the other GSO approaches, before and after consensus. In both cases, the detectors based on the shortest-path approaches outperform the conventional GFT, whereas using the eigenvectors of the Markov matrix yields the worst results for both cases.

C. Simulations over real networks

Two real networks are used and their structures are presented in Fig. 6a and Fig. 7a. For both cases, sensors' positions and healthy data are extracted from the available databases [53], [54]. Networks used in this example are basic sensor networks for which a Markovian relation is not initially defined, and we construct a Markovian relation between sensors as in (22).

For these databases, we assume that all data available are healthy. Thus, anomaly must be manually introduced in the data. For Experiments 3 and 4, this is conducted with minor differences from the method described for anomaly construction in Experiment 2. The anomaly is given by an additive Gaussian noise with non-zero mean over the healthy data. The mean value of the noise is allowed to vary according to a discrete uniform distribution, assuming non-zero integer values in the interval $[-b_{\max}, +b_{\max}]$. Similar to the synthetic case, a maximum possible number of anomalous sensors is defined and specified next for each simulation. For Experiment 5, we generate an anomaly similar to the one used in Experiment 1. The anomaly is given interference signal given by $s_i(\delta_x, \delta_y) = 5\left(\cos(2\pi 0.1\delta_x + \theta_x) + \cos(2\pi 0.1\delta_y + \theta_y)\right)$. For all cases, training is conducted in similar manner as that employed for the synthetic data, with 5 folds in cross validation.

1) Experiment 3: Intel lab data [53] - sensor malfunction: The network is modeled as a κNN sensor network with $\kappa = 3$ according to the available sensor positions. We use temperature data, in degrees Celsius, from 52 of the 54 sensors measured between 00 AM and 07 AM during the week from March 01, 2003 to March 05, 2003. Sensors named 5 and 15 are not used due to unavailable measurements. Healthy data range from 13.6 °C to 21.2 °C. The anomaly is described by $b_{\max} = 3$ °C, noise variance equal to 0.4 °C^2, and up to 2 anomalous sensors

The total number of available samples from the database is 373. For each independent run, 350 samples are randomly selected, of which half receive the anomaly. The 350 samples are then equally split into training and test datasets.

Results for the Intel lab data are presented in Fig 6b. In this case, the conventional approach is competitive against the scale-dependent methods. Still, the detectors based on the extended-adjacency for scale $t = 1$ outperform all other approaches. Moreover, using the eigenvectors of the Markov matrix also yields competitive results, with average f1 score approximately 0.02 behind the conventional approach.

2) Experiment 4: Global Surface Summary of the Day (GSOD) [54] - sensor malfunction: The database provides measurements from weather stations distributed across the territory of the United States of America. We use temperature measurements, converted from degrees Fahrenheit to degrees Celsius, obtained during the year 2010 by 150 randomly-selected stations from the conterminous United States (excluding Alaska, Hawaii, and other off-shore insular areas) in order to keep the graph connected. Network structure is derived from available stations' latitudes and longitudes. Healthy data range from -29.4 °C to 38.6 °C. We use $\kappa = 3$, $b_{\max} = 7$ °C, noise variance equal to 1 °C^2, and up to 7 anomalous sensors.

Daily samples are available, for a total of 365 signals. From these, we randomly select 350 signals for each run and generate training and test datasets as in Experiment 3.

In Fig. 7b, results show that the proposed approach outperforms the oher approaches for both scales $t = 1$ and $t = 2$, while the latter offers the best result. Here, the Markov-matrix-based approach outperforms the ones using the conventional Laplacian and the shortest-path-based GSOs.

3) Experiment 5: GSOD - spatially-spread anomaly: Using the GSOD data and network, we simulate the presence of a spatially-spread additive interference signal, given by $s_i(\delta_x, \delta_y) = 5\left(\cos(2\pi 0.1\delta_x + \theta_x) + \cos(2\pi 0.1\delta_y + \theta_y)\right)$, where δ_x and δ_y correspond to weather station's longitude and latitude, respectively, and with θ_x and θ_y randomly sampled from $[0, 2\pi]$. For each independent experiment, the interference is constrained to a randomly selected interval of 5 degrees of longitude, such that only the sensors in that interval are anomalous. Training and test datasets are generated as in Experiment 4.

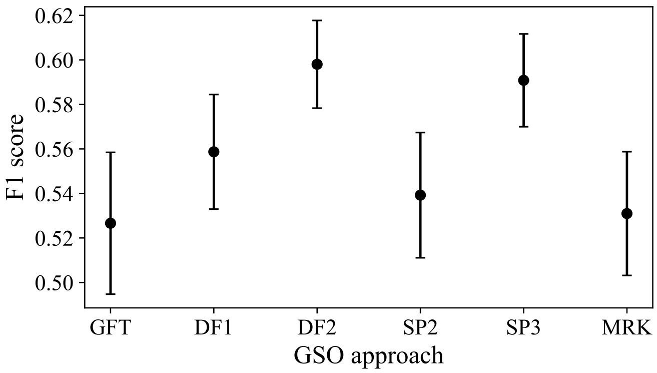


Fig. 8. F1 scores achieved by each GSO approach for Experiment 5.

Results are presented in Fig. 8 and show that the GFT considerably benefits from the proposed model when compared to the conventional approach. Moreover, the approach based on shortest-paths, for maximum length equal to 3, exhibits competitive results, while the approach based on the Markov matrix yields results similar to those of the conventional approach.

VII. Conclusion

We proposed the extended-adjacency matrix, which incorporates relations between non-adjacent nodes on a certain diffusion scale or time. We use the extended adjacency to augment the modeling of node relations in order to improve the efficiency of GSP tools. We also presented the scale-dependent graph Fourier transform for data defined over these networks, by using the conventional GFT together with the proposed scale-dependent model. We showed that increasing the diffusion scale results in an increased connectivity in the network, such that each different scale for the sGFT yields a different perspective of graph frequency, as a tailored connectivity is considered. We developed a theoretical analysis that shows that changing the diffusion scale shifts the spectral range yielded by the sGFT and corroborated the analysis with numerical experiments. Tools that operate on the graph spectrum can leverage on the additional information. For instance, we employed the sGFT for anomaly detection in synthetic and real networks. We used the free parameters of the sGFT to conduct a frequency analysis tailored for the given network. The proposed method was compared to the GFT based on the conventional graph Laplacian and to the GFT based on other augmented GSO models, and results showed that anomaly detectors based on the sGFT achieved better results than the other approaches.

References

[1] B. Boucher and S. Jenna, "Genetic interaction networks: better understand to better predict," *Front. in Genet.*, vol. 4, pp. 1–16, Dec. 2013.

[2] W. Huang, T. A. W. Bolton, J. D. Medaglia, D. S. Bassett, A. Ribeiro, and D. Van De Ville, "A graph signal processing perspective on functional brain imaging," *Proc. IEEE*, vol. 106, pp. 868–885, May 2018.

[3] I. Jablonski, "Graph signal processing in applications to sensor networks, smart grids, and smart cities," *IEEE Sensors J.*, vol. 17, pp. 7659–7666, Dec. 2017.

[4] A. Ortega, P. Frossard, J. Kovacevic, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proc. IEEE*, vol. 106, pp. 808–828, May 2018.

[5] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, pp. 80–90, Sep. 2014.

[6] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, pp. 83–98, May 2013.

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

[8] O. Teke and P. P. Vaidyanathan, "Extending classical multirate signal processing theory to graphs - Part I: Fundamentals," *IEEE Trans. Signal Process.*, vol. 65, pp. 409–422, Jan. 2017.

[9] O. Teke and P. P. Vaidyanathan, "Extending classical multirate signal processing theory to graphs - Part II: M-channel filter banks," *IEEE Trans. Signal Process.*, vol. 65, no. 2, pp. 423–437, Jan. 2017.

[10] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, pp. 1644–1656, Apr. 2013.

[11] B. Girault, "Stationary graph signals using an isometric graph translation," in *Proc. 23rd Eur. Signal Process. Conf.*, Aug. 2015, pp. 1516–1520.

[12] B. Girault, P. Goncalves, and E. Fleury, "Translation on graphs: An isometric shift operator," *IEEE Signal Process. Lett.*, vol. 22, pp. 2416–2420, Dec. 2015.

[13] A. Gavili and X.-P. Zhang, "On the shift operator, graph frequency, and optimal filtering in graph signal processing," *IEEE Trans. Signal Process.*, vol. 65, pp. 6303–6318, Dec. 2017.

[14] T. Summers, I. Shames, J. Lygeros, and F. Dorfler, "Topology design for optimal network coherence," in *Proc. Eur. Control Conf.*, July 2015, pp. 575–580.

[15] S. Shahrampour and V. M. Preciado, "Topology identification of directed dynamical networks via power spectral analysis," *IEEE Trans. Autom. Control*, vol. 60, pp. 2260–2265, Aug. 2015.

[16] M. Babakmehr, M. G. Simoes, M. B. Wakin, and F. Harirchi, "Compressive sensing-based topology identification for smart grids," *IEEE Trans. Ind. Informat.*, vol. 12, pp. 532–543, Apr. 2016.

[17] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning laplacian matrix in smooth graph signal representations," *IEEE Trans. Signal Process.*, vol. 64, pp. 6160–6173, Dec. 2016.

[18] D. Thanou, X. Dong, D. Kressner, and P. Frossard, "Learning heat diffusion graphs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, pp. 484–499, Sep. 2017.

[19] G. B. Giannakis, Y. Shen, and G. V. Karanikolas, "Topology identification and learning over graphs: Accounting for nonlinearities and dynamics," *Proc. IEEE*, vol. 106, pp. 787–807, May 2018.

[20] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Process. Mag.*, vol. 36, pp. 16–43, May 2019.

[21] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, pp. 215–233, Jan. 2007.

[22] J. Qin, Q. Ma, Y. Shi, and L. Wang, "Recent advances in consensus of multi-agent systems: A brief survey," *IEEE Trans. Ind. Electron.*, vol. 64, pp. 4972–4983, June 2017.

[23] B. Kailkhura, S. Brahma, and P. K. Varshney, "Data falsification attacks on consensus-based detection systems," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, pp. 145–158, Mar. 2017.

[24] L. Lovasz, "Random walks on graphs: A survey," *Combinatorics, Paul Erdos is Eighty*, vol. 2, pp. 1–46, 1993.

[25] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks," in *Proc. IEEE Conf. Comput. Commun.*, Mar. 2004, vol. 1, pp. 120–130.

[26] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, pp. 5–30, July 2006.

[27] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proc. Natl. Acad. Sci.*, vol. 102, pp. 7426–7431, May 2005.

[28] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Multiscale methods," *Proc. Natl. Acad. Sci.*, vol. 102, pp. 7432–7437, May 2005.

[29] R. Talmon, I. Cohen, S. Gannot, and R. R. Coifman, "Diffusion maps for signal processing: A deeper look at manifold-learning techniques based on kernels and graphs," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 75–86, June 2013.

[30] A. Heimowitz and Y. C. Eldar, "A unified view of diffusion maps and signal processing on graphs," in *Proc. 12th Int. Conf. on Sampl. Theory Appl.*, July 2017, pp. 308–312.

[31] A. Heimowitz and Y. C. Eldar, "The Nystrom extension for signals defined on a graph," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2018, pp. 4199–4203.

[32] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the Nyström method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, pp. 214–225, Feb. 2004.

[33] J. Mei and J. M. F. Moura, "Signal processing on graphs: causal modeling of *un*structured data," *IEEE Trans. Signal Process.*, vol. 65, pp. 2077–2092, Apr. 2017.

[34] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.

[35] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, pp. 3042–3054, June 2014.

[36] O. Teke and P. P. Vaidyanathan, "Time estimation for heat diffusion on graphs," in *Proc. 51st Asilomar Conf. Signals, Syst. Comput.*, Oct. 2018, pp. 1963–1967.

[37] N. Masuda, M. A. Porter, and R. Lambiotte, "Random walks and diffusion on networks," *Phys. Rep.*, vol. 716-717, pp. 1–58, Nov. 2017.

[38] B. Mohar, "The Laplacian spectrum of graphs", in *Graph Theory, Combinatorics, and Applications*, vol. 2, Wiley, 1991, pp. 871–898

[39] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Netw.*, vol. 54, pp. 2787–2805, Oct. 2010.

[40] B. Rashid and M. H. Rehmani, "Applications of wireless sensor networks for urban areas: A survey," *J. Net. Comput. Appl.*, vol. 60, pp. 192–219, Jan. 2016.

[41] E. Fadel, V. Gungor, L. Nassef, N. Akkari, M. A. Malik, S. Almasri, and I. F. Akyildiz, "A survey on wireless sensor networks for smart grid," *Comput. Commun.*, vol. 71, pp. 22–33, Nov. 2015.

[42] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Syst. Mag.*, vol. 27, pp. 71–82, Apr. 2007.

[43] M. A. Mahmood, W. K. Seah, and I. Welch, "Reliability in wireless sensor networks: A survey and challenges ahead," *Comput. Netw.*, vol. 79, pp. 166–187, 2015.

[44] B. Mostefa and G. Abdelkader, "A survey of wireless sensor network security in the context of internet of things," in *4th Int. Conf. Inf. Commun. Techol. Disaster Manag.*, Dec 2017, pp. 1–8.

[45] S. Kar and J. Moura, "Distributed consensus algorithms in sensor networks: Quantized data and random link failures," *IEEE Trans. Signal Process.*, vol. 58, pp. 1383–1400, Mar. 2010.

[46] I. Kayes and A. Iamnitchi, "Privacy and security in online social networks: A survey," *Online Soc. Netw. Media*, vol. 3-4, pp. 1–21, Oct. 2017.

[47] Y. Chen, S. Kar, and J. M. F. Moura, "Resilient distributed estimation through adversary detection," *IEEE Trans. Signal Process.*, vol. 66, pp. 2455–2469, May 2018.

[48] Y. Chen, S. Kar, and J. M. Moura, "The internet of things: Secure distributed inference," *IEEE Signal Process. Mag.*, vol. 35, pp. 64–75, Sep. 2018.

[49] Y. Chen, S. Kar, and J. M. F. Moura, "Resilient distributed estimation: Sensor attacks," *IEEE Trans. Autom. Control*, pp. 1–1, 2018.

[50] E. Drayer and T. Routtenberg, "Detection of false data injection attacks in power systems with graph Fourier transform," in *Proc. IEEE Global Conf. Signal. Inf. Process.*, Nov. 2018, pp. 890–894.

[51] M. Khatua, S. H. Safavi, and N. M. Cheung, "Sparse laplacian component analysis for internet traffic anomalies detection," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 4, pp. 697–711, Dec. 2018.

[52] D.E. Knuth, "A generalization of Dijkstra's algorithm", *Inf. Signal Lett.*, vol 6, pp 1–5, Feb. 1997.

[53] "Intel Lab Data." http://db.csail.mit.edu/labdata/labdata.html. Accessed: 2019-11-28.

[54] "Global Surface Summary of the Day - GSOD - Data.gov." https://catalog.data.gov/dataset/global-surface-summary-of-the-day-gsod. Accessed: 2019-11-28.

Vitor R. M. Elias received the B.Sc. degree in Electronics and Computer Engineering from the Federal University of Rio de Janeiro (UFRJ) in 2013. In 2015, he received the M.Sc. degree in Electrical Engineering from the COPPE/UFRJ. He is currently pursuing a joint Ph.D. degree at the UFRJ and the Norwegian University of Science and Technology (NTNU), where he was a visiting researcher from 2019 to 2020. He is currently with the Signals, Multimedia, and Telecommunications Lab (SMT/UFRJ) and the Signal Processing Group in the Department of Electronic Systems at NTNU. His experience and research interests include image and video compression, machine learning, adaptive learning, and digital signal processing on graphs.

Wallace A. Martins (SM'20) received the Electronics Engineer degree from the Federal University of Rio de Janeiro (UFRJ, Brazil) in 2007, and both M.Sc. and D.Sc. degrees in Electrical Engineering also from UFRJ in 2009 and 2011, respectively. He was a Research Visitor at University of Notre Dame (USA, 2008), at Université de Lille 1 (France, 2016), and at Universidad de Alcalá (Spain, 2018). From 2010 to 2013 he was an Associate Professor of the Federal Center for Technological Education (CEFET/RJ, Brazil). Since 2013 he has been with the Department of Electronics and Computer Engineering (DEL/Poli) and Electrical Engineering Program (PEE/COPPE) at UFRJ, where he is presently a tenured Associate Professor. He is currently a researcher working with the Interdisciplinary Centre for Security, Reliability and Trust (SnT) at Université du Luxembourg. His research interests are in the fields of digital signal processing, especially adaptive signal processing and graph signal processing, as well as digital communications, with focus on equalization and precoding for wireless communications. Dr. Martins has authored more than 60 technical papers in refereed journals and conferences, and 1 book. Also, he received the Best Student Paper Award from EURASIP at EUSIPCO-2009, Glasgow, Scotland, and the 2011 Best Brazilian D.Sc. Dissertation Award from Capes.

Stefan Werner (SM'07) received the M.Sc. degree in electrical engineering from the Royal Institute of Technology, Stockholm, Sweden, in 1998 and the D.Sc. degree (Hons.) in electrical engineering from the Signal Processing Laboratory, Helsinki University of Technology, Espoo, Finland, in 2002. He is currently a Professor in the Department of Electronic Systems, Norwegian University of Science and Technology, Trondheim, Norway. He is also an Adjunct Professor with Aalto University in Finland, and an Adjunct Senior Research Fellow with the Institute for Telecommunications Research, University of South Australia. He was a visiting Melchor Professor with University of Notre Dame during summer 2019, and was holding an Academy Research Fellowship, funded by the Academy of Finland, from 2009 to 2014. His research interests include adaptive and statistical signal processing, signal processing for communications, and security and privacy in cyberphysical systems. He is a member of the editorial boards for the EURASIP Journal of Signal Processing and the IEEE Transactions on Signal and Information Processing over Networks.