Digital Signal Processing 107 (2020) 102802

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Vertex-frequency graph signal processing: A comprehensive review

Ljubiša Stanković [a,*], Danilo Mandic [b], Miloš Daković [a], Bruno Scalzo [b], Miloš Brajović [a], Ervin Sejdić [c], Anthony G. Constantinides [b]

[a] University of Montenegro, Podgorica, Montenegro
[b] Imperial College London, London, United Kingdom
[c] University of Pittsburg, Pittsburg, PA, USA

ARTICLE INFO

Article history:
Available online 12 August 2020

Keywords:
Graph signal processing
Vertex-frequency analysis
Time-frequency analysis
Graph wavelet transform

ABSTRACT

Graph signal processing deals with signals which are observed on an irregular graph domain. While many approaches have been developed in classical graph theory to cluster vertices and segment large graphs in a signal independent way, signal localization based approaches to the analysis of data on graph represent a new research direction which is also a key to big data analytics on graphs. To this end, after an overview of the basic definitions of graphs and graph signals, we present and discuss a localized form of the graph Fourier transform. To establish analogy with classical signal processing, spectral domain and vertex domain definitions of the localization window are given next. The spectral and vertex localization kernels are then related to the wavelet transform, followed by their polynomial approximations and a study of filtering and inversion operations. For rigor, the analysis of energy representation and frames in the localized graph Fourier transform is extended to the energy forms of vertex-frequency distributions, which operate even without the requirement to apply localization windows. Another link with classical signal processing is established through the concept of local smoothness, which is subsequently related to the paradigm of signal smoothness on graphs, a lynchpin which connects the properties of the signals on graphs and graph topology. This all represents a comprehensive account of the relation of general vertex-frequency analysis with classical time-frequency analysis, an important but missing link for more advanced applications of graph signal processing. The theory is supported by illustrative and practically relevant examples.

1. Introduction

Analysis and processing of data may significantly benefit from appropriate exploitation of relations between sensing locations, signal values, and analyzed data types (objects). In this way, a new data domain in the form of a graph arises as a natural choice [1]. It comprehensively takes into account irregular data relations as a part of problem definition, along with the corresponding data connectivity. Although graph theory, as a branch of mathematics, has been established long time ago, it has largely focused on analyzing the underlying graphs rather than a conjoint analysis of graphs themselves and signals on graphs, a research topic of recent significant interest. Over-view of graph signal processing theory can be found in [2–5], along with relevant discussions regarding the relations with classical digital signal processing, recent advances, methods, and applications. The spectral domain analysis, processing, and filtering of signals on graphs are analyzed in [6,7], the sampling theory adapted to this emerging field has been also a topic of recent studies [8], while the principal relation of classical signal forms to the corresponding graph structures was presented in [9,10]. The Big Data paradigm has revealed the possibility of using smaller and localized subsets of the available information to enable a reliable mathematical analysis and local characterization of subsets of data of interest as elaborated in [11].

Oftentimes in practical applications concerned with large graphs, we may not be interested in the analysis of the entire graph signal, but rather in its local behavior. It is therefore natural and practically useful to characterize the localized signal behavior in the joint vertex-frequency domain, with the first important step to establish links to classical time-frequency analysis [12–14]. Indeed, the concept of vertex-frequency analysis was introduced in [15], by extending the principle of the signal localization by applying localization window functions to signals defined on graphs. This concept was further developed in [16], with the extensions

* Corresponding author.
E-mail addresses: ljubisa@ucg.ac.me (L. Stanković), d.mandic@imperial.ac.uk (D. Mandic), milos@ucg.ac.me (M. Daković), bruno.scalzo-dees12@imperial.ac.uk (B. Scalzo), milosb@ucg.ac.me (M. Brajović), esejdic@pitt.edu (E. Sejdić), a.constantinides@imperial.ac.uk (A.G. Constantinides).

https://doi.org/10.1016/j.dsp.2020.102802
1051-2004/

of this approach including the multi-window form [17], a short-graph Fourier transform combined with page-rank vectors [18], or vertex domain localization windows [19]. Window forms have also been adapted to define the frequency-varying localized graph Fourier transform [20] and spectral domain wavelet transform-based vertex-frequency kernels, including the signal adaptive kernels with polynomial approximations and recursive realizations, [21–23].

The extension of time-frequency analysis to vertex-frequency analysis is not straightforward, since, owing to inherent properties of graphs which are irregular but interconnected domains, even an operation which is very simple in classical time-domain analysis, like the time shift, cannot be straightforwardly generalized to graph signal domain. This has resulted in several approaches to the definition of the graph shift operator, and much ongoing research in this domain.

A common approach to windowing in the graph domain is to utilize the signal eigenspectrum to obtain window functions for each graph vertex [24]. Another possibility is to define a window support as a local neighborhood for each vertex [19]. In either case, the localization window is defined by a set of vertices that contain the current vertex, n, and all vertices that are close to the vertex n, that is, a neighborhood of vertex n. Special attention is paid to the local graph Fourier transform approaches that can be implemented in the vertex domain, since this domain can be a basis for numerically efficient analysis in the case of very large graphs and big data.

As in the classical signal analysis, a localization window should be narrow enough in order to provide good localization of the signal properties in the vertex domain but wide enough to produce high resolution in the graph spectral domain. To automatize the process of making this compromise, optimization approaches are typically involved, some of which are related to the uncertainty principle. This forms the basis for vertex-frequency analysis which can be used for graph signal estimation, filtering, and efficient graph signal representation, together with a framework for signal reconstruction in the vertex-frequency domain. Two forms of the local graph Fourier transform inversion approaches (corresponding to the constant overlap-add and weighted overlap-add methods in classical time-frequency analysis) are considered in this work, while the inversion condition is also defined within the frames framework. For generality, the local graph Fourier transform and its inversion are further related to the graph wavelet transform.

Finally, the energy versions of the vertex-frequency representations are considered and shown to admit implementation even without a localization window, through their relation with the local smoothness index estimators. The reduced interference vertex-frequency distributions, which satisfy the marginal property and localize graph signal energy in the vertex-frequency domain are subsequently reviewed and related to the classical time-frequency analysis, as a special case. All concepts are illustrated through intuitive examples.

The paper is organized as follows. The basic definitions of graph signals and spectral graph domain are given in Section 2. The localized graph Fourier transform is presented in Section 3, where various approaches to define this transform are considered. The local graph Fourier transform is further related to the graph wavelet transform. The topic of Section 4 is the optimization of the graph signal localization window, while Section 5 gives the inversion relations and conditions for the considered graph transforms. The uncertainty principle in graph signals is reviewed in Section 6. The graph spectrogram is related to the frames in Section 7. The energy vertex-frequency representations are defined and analyzed in Section 8.

Table 1
Summary of graph spectral bases.

Operator	Eigenanalysis relation
Graph Laplacian	$\mathbf{L}\mathbf{u}_k = \lambda_k \mathbf{u}_k$
Generalized eigenvectors of graph Laplacian	$\mathbf{L}\mathbf{u}_k = \lambda_k \mathbf{D}\mathbf{u}_k$
Normalized graph Laplacian	$(\mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2})\mathbf{u}_k = \lambda_k \mathbf{u}_k$
Adjacency matrix	$\mathbf{A}\mathbf{u}_k = \lambda_k \mathbf{u}_k$
Normalized adjacency matrix	$\left(\frac{1}{\lambda_{\max}}\mathbf{A}\right)\mathbf{u}_k = \lambda_k \mathbf{u}_k$

2. Review of basic background concepts

Consider a graph with N vertices, denoted by $n \in \mathcal{V} = \{1, 2, \ldots, N\}$, which are connected with edges whose weights are W_{mn}. If the vertices m and n are not connected, then $W_{mn} = 0$. The edge weights W_{mn} are commonly written in an $N \times N$ matrix form, $\mathbf{W}$. For undirected graphs the weight matrix $\mathbf{W}$ is symmetric, $\mathbf{W} = \mathbf{W}^T$. In the case of unweighted graphs, all nonzero elements in $\mathbf{W}$ are equal to unity, and this specific form is called the connectivity or adjacency matrix, denoted by $\mathbf{A}$.

For an enhanced graph description, in addition to matrix $\mathbf{A}$ or $\mathbf{W}$, it is common to use a diagonal degree matrix $\mathbf{D}$, whose elements D_{nn} are equal to the sum of all edge weights connected to the considered vertex, n, that is, $D_{nn} = \sum_m W_{mn}$, thus indicating the vertex importance. The weight matrix and the degree matrix can be combined to produce the graph Laplacian, given by $\mathbf{L} = \mathbf{D} - \mathbf{W}$. The Laplacian of an undirected graph is symmetric, $\mathbf{L} = \mathbf{L}^T$.

Spectral analysis of graphs is most commonly based on the eigendecomposition of the graph Laplacian $\mathbf{L}$ or the adjacency matrix, $\mathbf{A}$. Some of the common eigendecompositions in graph signal analysis are given in Table 1. By default, we will assume the decomposition of the graph Laplacian, $\mathbf{L}$, if not stated otherwise. The eigenvectors, $\mathbf{u}_k$, and the eigenvalues, λ_k, of the graph Laplacian are calculated based on the usual definition $\mathbf{L}\mathbf{u}_k = \lambda_k \mathbf{u}_k$, $k = 1, 2, \ldots, N$.

The graph Fourier transform (GFT), denoted by

$$\mathbf{X} = [X(1), X(2), \ldots, X(N)]^T,$$

of a graph signal, $\mathbf{x} = [x(1), x(2), \ldots, x(N)]^T$, is defined as an expansion onto a set of orthonormal basis functions, $\mathbf{u}_k$, the elements of which are $u_k(n)$, $n = 1, 2, \ldots, N$, that is

$$X(k) = \mathrm{GFT}\{x(n)\} = \sum_{n=1}^{N} x(n)\, u_k(n). \tag{1}$$

The inverse graph Fourier transform (IGFT) is then defined accordingly as

$$x(n) = \mathrm{IGFT}\{X(k)\} = \sum_{k=1}^{N} X(k)\, u_k(n). \tag{2}$$

More details on the GFT definition and interpretation can be found in Appendix A.

Remark 1. The graph Fourier transforms in (1) and (2) reduces to the classical discrete Fourier transform (DFT) and the inverse DFT, if the graph is circular and directed. The eigenvectors of a directed graph are complex-valued and for this case we should use complex-conjugate basis functions, $u_k^*(n)$, in (1). The eigendecomposition of the adjacency matrix $\mathbf{A}$ is commonly used for the directed graphs [25,26].

Since the graph Laplacian eigenvectors are a key concept underpinning vertex-frequency representations, we shall now review their properties related to the presented analysis. The Laplacian matrix is defined in such a way that the sum of elements in each its row (column) is zero. As a consequence, this enforces the inner products of every row of $\mathbf{L}$ with any constant vector to be zero-valued. This also means that at least one eigenvalue of the Laplacian is zero, $\lambda_1 = 0$, and its corresponding eigenvector is given by $\mathbf{u}_1 = [1, 1, \ldots, 1]^T/\sqrt{N}$. Since the graph Laplacian eigenvectors are orthogonal, $\mathbf{u}_k^T\mathbf{u}_l = 0$ for $k \neq l$, this means that $\sum_{n=1}^{N} u_k(n)u_l(n) = \delta(k-l)$. Also, the eigenvectors for $k > 1$ are orthogonal to the constant eigenvector, $\mathbf{u}_1$, resulting in $\sum_{n=1}^{N} u_k(n) = 0$, for $k > 1$.

From the eigenvector definition, $\mathbf{L}\mathbf{u}_k = \lambda_k\mathbf{u}_k$, it follows that $\mathbf{u}_k^T\mathbf{L}\mathbf{u}_k = \lambda_k$, or in an element-wise form

$$\mathbf{u}_k\mathbf{L}\mathbf{u}_k = \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} W_{mn}\Big(u_k(n) - u_k(m)\Big)^2 = \lambda_k.$$

In other words, the above quadratic form of the graph Laplacian is equal to the sum of squared differences of the eigenvector value at a vertex, n, and its neighboring vertices, m. In this way, the eigenvalue can be considered as a measure of the rate of change (smoothness) of the considered eigenvector, whereby faster changing eigenvectors (corresponding to high-frequency signals) have larger smoothness values. Therefore, the smoothness of an eigenvector, λ_k, in graph signal processing plays the role of frequency in classical spectral analysis (see Appendix A).

3. Localized graph Fourier transform (LGFT)

The localized graph Fourier transform (LGFT) can be considered an extension of the standard localized time (short time) Fourier transform (STFT), and can be calculated as the GFT of a signal, $x(n)$, multiplied by an appropriate vertex localization window function, $h_m(n)$, to yield

$$S(m,k) = \sum_{n=1}^{N} x(n)h_m(n)\,u_k(n). \tag{3}$$

In general, is assumed that a graph window function, denoted by $h_m(n)$, should be such that it localizes the signal content around the vertex m. In other words, its values should be close to 1 at a vertex m and vertices in its close neighborhood, while approaching 0 for vertices far from vertex m.

Remark 2. The classical time-frequency analysis (STFT) form of the relation in (3) is

$$S(m,k) = \frac{1}{\sqrt{N}}\sum_{n=1}^{N} x(n)h_m(n)e^{-j2\pi(n-1)(k-1)/N},$$

with the localization window, $h_m(n) = h(n-m)$.

In a matrix notation, the local GFT, $\mathbf{S}$, contains all the elements, $S(m,k)$, $m = 1, 2, \ldots, N$, $k = 1, 2, \ldots, N$, while the columns of $\mathbf{S}$ which correspond to a vertex m are given by

$$\mathbf{s}_m = \mathrm{GFT}\{x(n)h_m(n)\} = \mathbf{U}^H\mathbf{x}_m,$$

where $\mathbf{x}_m$ is the vector whose elements, $x(n)h_m(n)$, are equal to the graph signal samples, $x(n)$, multiplied by the window function, $h_m(n)$, centered at the vertex m, and matrix $\mathbf{U}$ is composed of eigenvectors $\mathbf{u}_k$, $k = 1, 2, \ldots, N$ as its columns.

In general, the set of vertices, m, and spectral indices, k, where the LGFT is calculated may be reduced from the original N^2 vertex-frequency points, (m,k), to N points and corresponding values, $S(m,k)$, in a non-redundant representation [27].

Special cases:

- For $h_m(n) = 1$, the localized vertex spectrum is equal to the standard spectrum $S(m,k) = X(k)$ in (1) for each m, meaning that no vertex localization is performed.
- If $h_m(m) = \delta(m-n)$, the localized vertex spectrum for $k = 1$ is equal to $S(m,1) = x(m)/\sqrt{N}$.

The subsequent subsections outline ways to create appropriate windows for vertex-frequency analysis. Several approaches to the definition of a localization kernel will be presented, including:

- A spectrum domain defined window function, which is graph shifted and modulated to the vertex-frequency domain point;
- A spectral shifted form of the kernel using a band-pass type function in the spectral domain;
- Spectral varying forms of the spectral shifted kernels, as in the spectral graph wavelet transform or a signal adaptive kernel;
- Pure vertex domain forms and graph shifts of the window.

For the spectral shifted forms of the LGFT, the polynomial approximations shall be presented together with appropriate recursive realizations in the localized vertex domain, leading to the vertex domain implementation of the LGFT.

3.1. Windows defined in the GFT domain

Generalized convolution of graph signals. Consider two signals, $x(n)$ and $y(n)$, defined on a graph, with the corresponding GFTs denoted by $X(k)$ and $Y(k)$. *A generalized convolution*, $z(n)$, of signals $x(n)$ and $y(n)$ can then be defined in the GFT domain, in analogy with the classical definition of convolution, as

$$z(n) = x(n) * y(n) = \mathrm{IGFT}\{Z(k)\}, \text{ where}$$
$$Z(k) = \mathrm{GFT}\{x(n) * y(n)\} = X(k)Y(k). \tag{4}$$

Shift operator for graph signals. A "shift" on a graph cannot be extended to graph signals in a direct and unique analogy to the classical signal shift. Among several forms of signal shift on a graph which have been proposed in graph theory, the most commonly used in the graph signal filtering and the GFT definition are those based on the matrix multiplication of the signal with one of the graph connectivity operators (listed in Table 1 and discussed in Appendix A). Within vertex-frequency analysis, an interesting attempt has been made to generalize the convolution and to define a corresponding shift operator on a graph [15]. Because of its significance in this field, this approach will be reviewed next.

Consider the graph signal, $h(n)$, and the delta function located at a vertex m, given by $\delta_m(n) = \delta(n-m)$. The GFT of delta function, $\delta_m(n)$, is then given by

$$\Delta_m(k) = \mathrm{GFT}\{\delta_m(n)\} = \sum_{n=1}^{N}\delta_m(n)u_k(n) = u_k(m).$$

We will use the symbol $h_m(n)$ to denote a shifted version of the graph signal, $h(n)$, "toward"a vertex m. Based on (4), this kind of graph signal shift will be defined following the same reasoning as in classical signal processing, where the shifted signal is obtained as a convolution of the original signal and an appropriately shifted

delta function. Therefore, a graph shifted signal can be defined as a generalized graph convolution, $h(n) * \delta_m(n)$, the GFT of which is equal to

$$\text{GFT}\{h(n) * \delta_m(n)\} = H(k)u_k(m). \tag{5}$$

The graph-shifted signal then represents the IGFT of $H(k)u_k(m)$, so that from (4) the window localized at the vertex m, denoted by $h_m(n)$, is given by [16]

$$h_m(n) = h(n) * \delta_m(n) = \sum_{k=1}^{N} H(k)u_k(m)u_k(n). \tag{6}$$

The basic form of this window, $h(n)$, can be conveniently defined in the spectral domain, for example as

$$H(k) = C\exp(-\lambda_k \tau), \tag{7}$$

where C denotes the window amplitude and $\tau > 0$ is a constant which determines the window width. An example of two windows obtained in this way is illustrated in Fig. 2(a), (b). Observe that the exponential function in (7) corresponds to a Gaussian window in classical analysis (thus offering the best time-frequency concentration [12–14]), since graph signal processing on a circular or path graph reduces to classical signal analysis. In that case, the eigenvalues of the graph Laplacian, λ, may be related to the frequency, ω, in the classical signal analysis as $\lambda \sim \omega^2$, see Appendix A.

Properties of graph window functions. The window localized at the vertex m, and defined by (6), satisfies the following properties:

W_1: Symmetry, $h_m(n) = h_n(m)$. This property follows from the definition in (6), for real-valued eigenvectors.

W_2: A sum of all localized window coefficients, $h_m(n)$, is equal to $H(1)$, since

$$\sum_{n=1}^{N} h_m(n) = \sum_{k=1}^{N} H(k)u_k(m) \sum_{n=1}^{N} u_k(n)$$
$$= \sum_{k=1}^{N} H(k)u_k(m)\delta(k-1)\sqrt{N} = H(1),$$

with $\sum_{n=1}^{N} u_k(n) = \delta(k-1)\sqrt{N}$, by definition of the graph Laplacian eigenvectors, $u_k(n)$.

W_3: The Parseval theorem for $h_m(n)$ has the form

$$\sum_{n=1}^{N} |h_m(n)|^2 = \sum_{k=1}^{N} |H(k)u_k(m)|^2. \tag{8}$$

These properties will be used in the sequel in the inversion analysis of the LGFT.

For the window form in (6), the LGFT can be written as

$$S(m,k) = \sum_{n=1}^{N} x(n)h_m(n)\, u_k(n) \tag{9}$$

$$= \sum_{n=1}^{N}\sum_{p=1}^{N} x(n)H(p)u_p(m)u_p(n)\, u_k(n). \tag{10}$$

The modulated (spectral shifted) version of the window centered at vertex m and for spectral index k will be referred to as the *vertex-frequency kernel*, $\mathcal{H}_{m,k}(n)$, defined as

$$\mathcal{H}_{m,k}(n) = h_m(n)u_k(n) = \Big(\sum_{p=1}^{N} H(p)u_p(m)u_p(n)\Big)u_k(n). \tag{11}$$

Using the kernel notation, it becomes obvious that the LGFT, for a given vertex m and spectral index k, physically represents *a projection of a graph signal $x(n)$ onto the graph kernel $\mathcal{H}_{m,k}(n)$*, that is,

$$S(m,k) = \langle \mathcal{H}_{m,k}(n), x(n)\rangle = \sum_{n=1}^{N} \mathcal{H}_{m,k}(n)x(n). \tag{12}$$

Remark 3. The classical STFT, a basic tool in time-frequency analysis, can be obtained as a special case of the GFT when the graph is directed and circular. For this type of graph, the adjacency matrix decomposition produces complex-valued eigenvectors of the form (see Appendix A)

$$u_k(n)\sqrt{N} = e^{j2\pi(n-1)(k-1)/N},$$

$n = 1, 2, \ldots, N$, $k = 1, 2, \ldots, N$. Then, having in mind the complex nature of eigenvectors, the value of $S(m,k)$ in (9) becomes the standard STFT, that is

$$S(m,k) = \frac{1}{N^{3/2}} \sum_{n=1}^{N}\sum_{p=1}^{N} x(n)H(p)e^{-j\frac{2\pi}{N}(m-1)(p-1)}$$
$$\times e^{j\frac{2\pi}{N}(n-1)(p-1)}e^{-j\frac{2\pi}{N}(n-1)(k-1)}$$
$$= \frac{1}{\sqrt{N}} \sum_{n=1}^{N} x(n)h(n-m)e^{-j2\pi(n-1)(k-1)/N}, \tag{13}$$

where $h(n)$ is the inverse DFT of $H(k)$.

Example 1. To illustrate the principle of local vertex-frequency representation, consider the graph and the graph signal from Fig. 1. A graph with $N = 100$ vertices, randomly placed on the so called Swiss roll surface, is shown in Fig. 1(a). The vertices are connected with edges whose weights are defined as $W_{mn} = \exp(-r_{mn}^2/\alpha)$, where r_{mn} is the Euclidean distance between vertices m and n, measured on the Swiss roll manifold, and α is a constant [5]. Small weight values are hard-thresholded to zero, to reduce the number of edges associated with each vertex to only a few strongest ones, to produce the graph as in Fig. 1(b), the two-dimensional presentation of which is shown in Fig. 1(c). Vertices are ordered so that the values of the Fiedler eigenvector, $u_2(n)$, are nondecreasing [5].

The signal on this graph, shown in Fig. 1(d), is composed of parts of three Laplacian eigenvectors. For the subset of all vertices $\mathcal{V}$, denoted by $\mathcal{V}_1$, which comprises vertices with indices from $m = 1$ to $m = 40$, the eigenvector with spectral index $k = 72$ was used. For the subset $\mathcal{V}_2$, with vertex indices from $m = 41$ to $m = 70$, the signal was equal to the eigenvector $u_k(n)$ with $k = 50$. The remaining vertices form the vertex subset $\mathcal{V}_3$, and the signal on this subset was equal to the eigenvector with the spectral index $k = 6$. Amplitudes of the eigenvectors were scaled too.

Consider now the vertex-frequency localization kernels, $\mathcal{H}_{m,k}(n) = h_m(n)u_k(n)$, shown in Fig. 2. The constant eigenvector, $u_1(n) = 1/\sqrt{N}$, is used in the panel shown in Fig. 2(a) at $m = 28$, with $C = 1$ and $\tau = 1$ in (7). In this case, the localization window, $h_{28}(n)$, is presented since $\mathcal{H}_{28,1}(n) = h_{28}(n)/\sqrt{N}$. The illustration is repeated in the panel in Fig. 2(c) for the vertex $m = 94$. Frequency shifted versions of these two kernels, shown in Figs. 2(a) and (c), are given respectively in Figs. 2(b) and (d), where $\mathcal{H}_{m,21}(n) = h_m(n)u_{21}(n)$ is shown for $m = 28$ and $m = 94$, respectively.

The vertex-frequency representation, $S(n,k)$, using the LGFT and the localization window defined in the spectral domain is

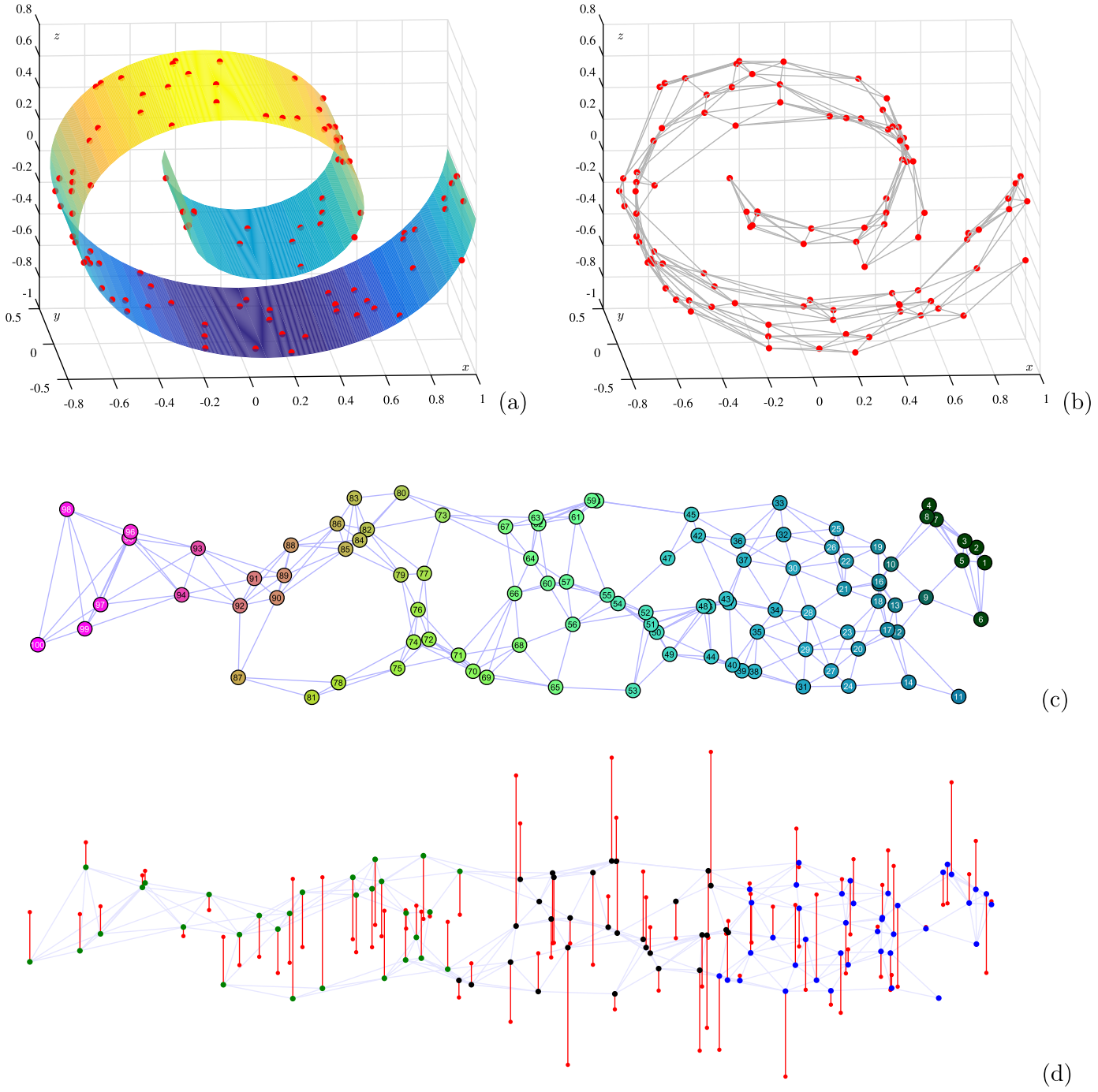

Fig. 1. Concept of a graph and a signal on a graph. (a) Vertices on a three-dimensional Swiss roll surface. (b) A graph representation on the Swiss roll manifold. (c) Two-dimensional presentation of the three-dimensional graph from (b), with vertex colors defined by the graph Laplacian eigenvectors $u_1(n)$, $u_2(n)$, and $u_3(n)$. (d) A signal on the graph in (c), which is composed of three eigenvectors (signal components). Supports of these three components are designated by different vertex colors. The vertex-frequency representations are then assessed based on their ability to clearly resolve and localize these three graph signal components. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

shown in Fig. 3. From this representation, we can clearly follow the three constituent signal components, within their intervals of support. The marginal properties, such as the projections of $S(n,k)$ to the vertex index and spectral index axis, are also clearly distinguishable. From the marginal properties we can conclude that the graph signal in hand is spread over all vertex indices, while its spectral localization is dominated by the three spectral indices which correspond to the three components of the original graph signal. In an ideal case of the vertex-frequency analysis, these marginals should respectively be equal to $|x(n)|^2$ and $|X(k)|^2$, which is not the case here.

3.2. Spectral domain localization of the LGFT

Recall that the classical STFT admits frequency localization using a window in the spectral domain, whereby the dual form of STFT is obtained using the DFT of the original signal and spectral

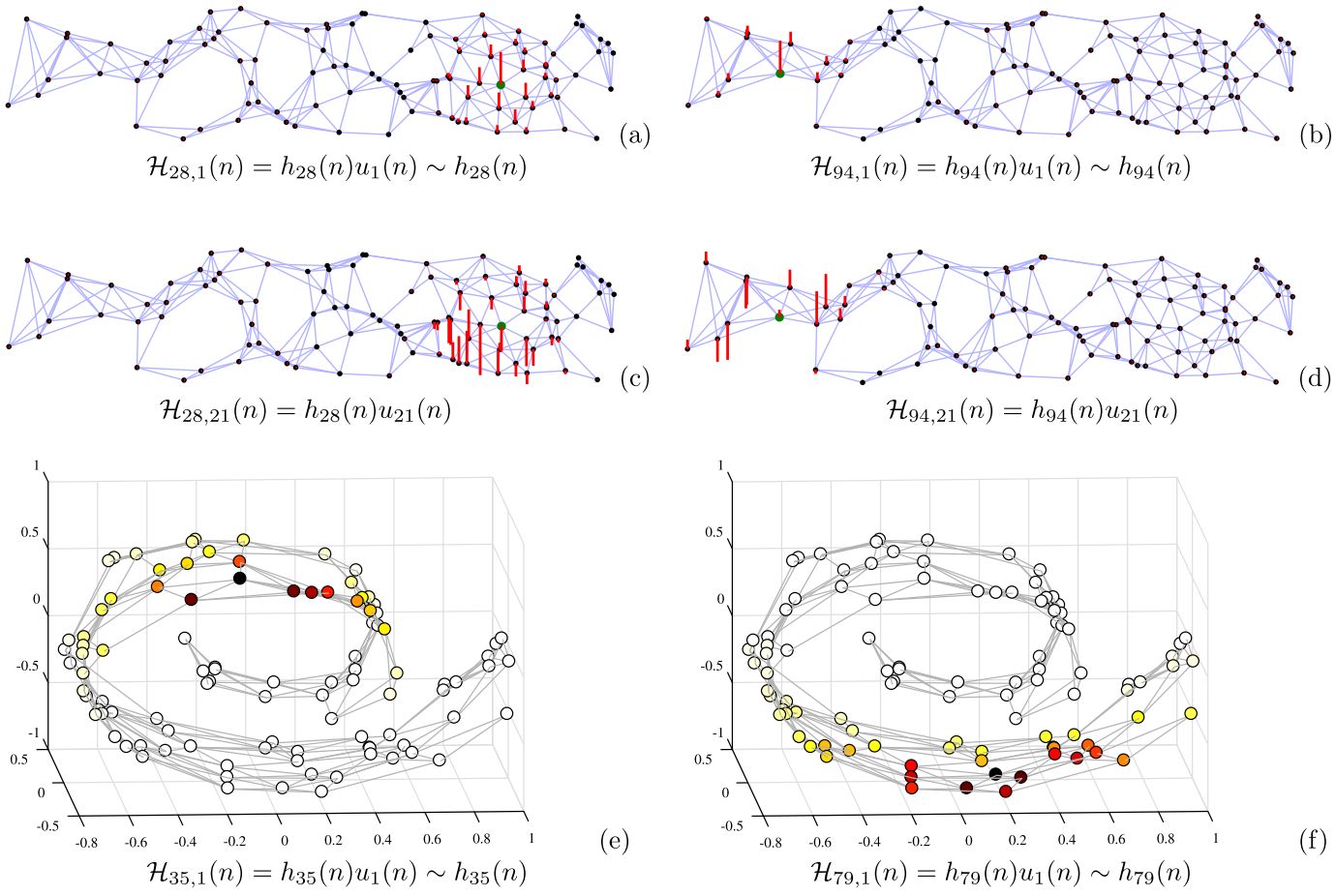

Fig. 2. Illustration of localization kernels, $\mathcal{H}_{m,k}(n) = h_m(n)u_k(n)$, for vertex-frequency analysis based on *spectral domain defined windows* in the local graph Fourier transform, $S(m,k) = \sum_{n=1}^{N} x(n)\mathcal{H}_{m,k}(n)$. (a) Localization kernel $\mathcal{H}_{28,1}(n) = h_{28}(n)u_1(n) \sim h_{28}(n)$, for a constant eigenvector, $u_1(n) = 1/\sqrt{N}$, centered at the vertex $m = 28$. (b) The same localization kernel as in (a) but centered at the vertex $m = 94$. (c) Localization kernel, $\mathcal{H}_{28,21}(n) = h_{28}(n)u_{21}(n)$, centered at the vertex $m = 28$ and frequency shifted by $u_{21}(n)$. Notice that the variations in kernel amplitude indicate the effects of modulation of the localization window $h_m(n)$. (d) The same localization kernel as in (c), but centered at the vertex $m = 94$. (e) Three-dimensional representation of kernel $\mathcal{H}_{35,1}(n) = h_{35}(n)u_1(n)$, (f) Three-dimensional representation of kernel $\mathcal{H}_{79,1}(n) = h_{79}(n)u_1(n)$.

domain window. For graph signals, we can also use this approach to perform localization in the spectral domain, whereby the LGFT is obtained as an inverse GFT of $X(p)$ localized by a spectral domain window, $H(k-p)$, which is centered around spectral index k, that is

$$S(m,k) = \sum_{p=1}^{N} X(p)H(k-p)\,u_p(m). \tag{14}$$

Note that this form of the LGFT can be entirely implemented in the graph spectral domain, without a graph shift operator in the vertex domain.

Remark 4. The classical time-frequency analysis counterpart of (14) is given by [12]

$$S(m,k) = \frac{1}{\sqrt{N}} \sum_{p=1}^{N} X(p)H(k-p)e^{j\frac{2\pi}{N}(m-1)(p-1)},$$

where $H(k)$ is a frequency domain window (a band-pass function).

The spectral domain LGFT form in (14) can be implemented using band-pass transfer functions, $H_k(\lambda_p) = H(k-p)$, as

$$S(m,k) = \sum_{p=1}^{N} X(p)H_k(\lambda_p)\,u_p(m). \tag{15}$$

The elements $S(m,k)$, $m = 1,2,\ldots,N$ of the LGFT matrix $\mathbf{S}$ can also be written in a matrix form

$$\mathbf{s}_k = \text{IGFT}_p\{X(p)H_k(\lambda_p)\} = \mathbf{U}\,H_k(\mathbf{\Lambda})\mathbf{X}, \tag{16}$$

where $H_k(\mathbf{\Lambda})$ is a diagonal matrix with elements $H_k(\lambda_p)$, $p = 1,2,\ldots,N$.

Remark 5. The kernel in (11) is defined based on a low-pass transfer function $H(k)$, which is appropriately shifted in the spectral domain using the modulation term, $u_k(n)$. The transfer function in (15), $H_k(\lambda_p)$, is centered (shifted) at a spectral index, k, by definition. Hence, in this case, the modulation term, $u_k(n)$, is not needed and the kernel is now of the form

$$\mathcal{H}_{m,k}(n) = \sum_{p=1}^{N} H_k(\lambda_p)u_p(m)u_p(n). \tag{17}$$

3.3. LGFT realization with band-pass functions

Consider the spectral domain localization windows in a form of a transfer function of a band-pass graph system, centered at an eigenvalue, λ_k, and around it, and that it is defined in the form of a polynomial given by

$$H_k(\lambda_p) = h_{0,k} + h_{1,k}\lambda_p + \cdots + h_{M-1,k}\lambda_p^{M-1}, \tag{18}$$

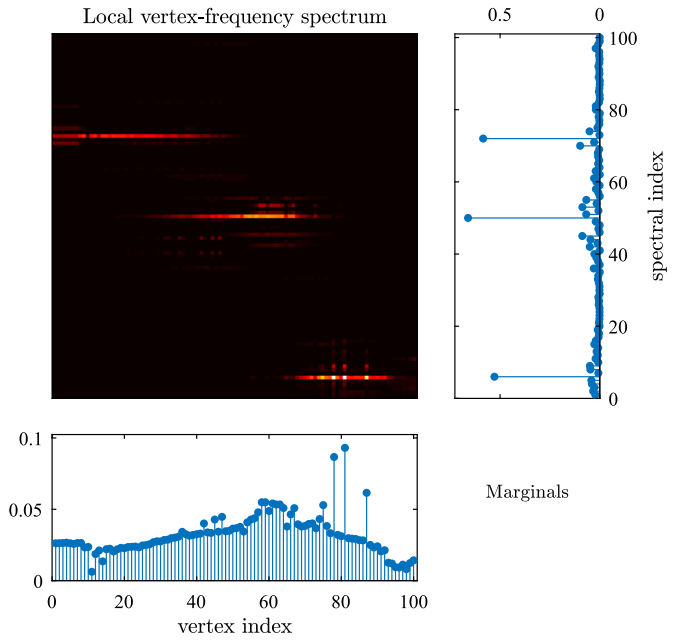

Fig. 3. Local vertex-frequency spectrum calculated using the LGFT and the vertex-frequency localized kernels defined in the spectral domain, as in (11). From this representation, we can see that the graph signal consists of three components located at spectral indices $k = 72$, $k = 50$, and $k = 6$, with the corresponding vertex indices subsets $\mathcal{V}_1$, $\mathcal{V}_2$, and $\mathcal{V}_3$, where $\mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{V}_3 = \mathcal{V}$. The marginal (vertex and spectrum-wise) properties are shown in the panels right and below the vertex-frequency representation. Observe that, while the graph signal is spread across all vertices, its spectral content is localized at three spectral indices which correspond to the constituent signal components. In an ideal case of vertex-frequency analysis, these marginals should be respectively equal to $|x(n)|^2$ and $|X(k)|^2$.

with M being the polynomial order, and $k = 0, 1, \ldots, K-1$, where K is the number of bands (see Appendix A).

The vertex-frequency transform, $S(m, k)$, for the vertex m and spectral index k then assumes the form as in (17), which can be written in a vector form using (16) as

$$\mathbf{s}_k = \mathbf{U}H_k(\mathbf{\Lambda})\mathbf{X} = \mathbf{U}H_k(\mathbf{\Lambda})\mathbf{U}^T\mathbf{x}$$
$$= H_k(\mathbf{L})\mathbf{x} = \sum_{p=0}^{M-1} h_{p,k}\mathbf{L}^p\,\mathbf{x}, \tag{19}$$

where $\mathbf{s}_k$ is the column vector with elements $S(m, k)$, $m = 1, 2, \ldots, N$, and the property of eigendecomposition of a matrix polynomial is used in the derivation (see Appendix A). *In this case, the number of shifted windows, K, is not related to the total number of indices N.*

Example 2. Consider the simplest decomposition into a low-pass and high-pass part of a graph signal, with $K = 2$. In this case, two values $k = 0$ and $k = 1$ represent the low-pass part and high-pass part of the graph signal. Such a decomposition can be achieved by using the graph Laplacian with $h_{0,0} = 1$, $h_{0,1} = -1/\lambda_{\max}$, and $h_{1,0} = 0$, $h_{1,1} = 1/\lambda_{\max}$, where the coefficients are chosen to form a simple linearly decreasing function of λ_p for the low-pass part, and a linearly increasing function of λ_p for the high-pass part, in the corresponding transfer functions. These transfer functions are respectively given by

$$H_0(\lambda_p) = (1 - \frac{\lambda_p}{\lambda_{\max}}), \quad H_1(\lambda_p) = \frac{\lambda_p}{\lambda_{\max}},$$

leading to the vertex domain implementation of the LGFT as

$$\mathbf{s}_0 = (\mathbf{I} - \frac{1}{\lambda_{\max}}\mathbf{L})\,\mathbf{x}, \quad \mathbf{s}_1 = \frac{1}{\lambda_{\max}}\mathbf{L}\mathbf{x}.$$

This simple decomposition system is used with a normalized graph Laplacian in [28], eq. (7), as the key model for graph convolution networks (GCN), derived from the first order Chebyshev polynomial approximation.

To improve the spectral resolution, we can continue with the same transfer function by dividing the low-pass part into its low-pass and high-pass part. The same can be performed for the high-pass part, to obtain

$$\mathbf{s}_{00} = \Big(\mathbf{I} - \frac{\mathbf{L}}{\lambda_{\max}}\Big)^2\mathbf{x}, \quad \mathbf{s}_{01} = 2\Big(\mathbf{I} - \frac{\mathbf{L}}{\lambda_{\max}}\Big)\frac{\mathbf{L}}{\lambda_{\max}}\mathbf{x}, \quad \mathbf{s}_{11} = \frac{\mathbf{L}^2}{\lambda_{\max}^2}\mathbf{x},$$

the factor of 2 appears in the new middle pass-band, $\mathbf{s}_{01}$, since the low-high-pass and the high-low-pass components are the same.

The division can be extended to a general case of K bands, corresponding to the terms of a binomial form

$$\Big((\mathbf{I} - \mathbf{L}/\lambda_{\max}) + \mathbf{L}/\lambda_{\max}\Big)^{K-1}\mathbf{x},$$

with the transfer functions in the vertex domain given by

$$H_k(\mathbf{L}) = \binom{K-1}{k}\Big(\mathbf{I} - \frac{1}{\lambda_{\max}}\mathbf{L}\Big)^{K-1-k}\Big(\frac{1}{\lambda_{\max}}\mathbf{L}\Big)^k.$$

Example 3. The transfer functions $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$ in the spectral domain, corresponding to the binomial form terms for $K = 26$, are shown in Fig. 4(a). These functions are used for the LGFT calculation at vertex indices $m = 1, 2, \ldots, N$ in $k = 0, 1, \ldots, K-1$ bands. Since the bands are quite spread out, the resulting LGFT is also spread along the frequency axis. The concentration can be improved by reassigning the values of $S(m, k)$ to the position of their maximum value along the frequency band index, k, for each vertex index, m. Such a reassigned LGFT values are given in Fig. 5 (a), which shows that this extremely simple vertex-frequency representation can be quite acceptable for a rough analysis.

3.3.1. Band-pass functions of the Hann window form

Of course, any other set of band-pass functions, $H_k(\mathbf{\Lambda})$, can be used to produce the LGFT from (16) in the form (19), to give

$$\mathbf{s}_k = H_k(\mathbf{L})\mathbf{x}. \tag{20}$$

Commonly used examples of such band-pass functions are the spline or raised cosine (Hann window) functions. We will next use the general form of the shifted raised cosine functions as the transfer functions, defined by

$$H_k(\lambda) = \begin{cases} \sin^2\Big(\frac{\pi}{2}\frac{a_k}{b_k - a_k}(\frac{\lambda}{a_k} - 1)\Big), & \text{for } a_k < \lambda \le b_k \\ \cos^2\Big(\frac{\pi}{2}\frac{b_k}{c_k - b_k}(\frac{\lambda}{b_k} - 1)\Big), & \text{for } b_k < \lambda \le c_k \\ 0, & \text{elsewhere,} \end{cases} \tag{21}$$

where $(a_k, b_k]$ and $(b_k, c_k]$, $k = 0, 1, \ldots, K-1$, define the spectral bands for $H_k(\mathbf{\Lambda})$. For uniform bands within $0 \le \lambda \le \lambda_{\max}$, the intervals can be defined by

$$a_k = a_{k-1} + \frac{\lambda_{\max}}{K-1},$$
$$b_k = a_k + \frac{\lambda_{\max}}{K-1},$$
$$c_k = a_k + 2\frac{\lambda_{\max}}{K-1} \tag{22}$$

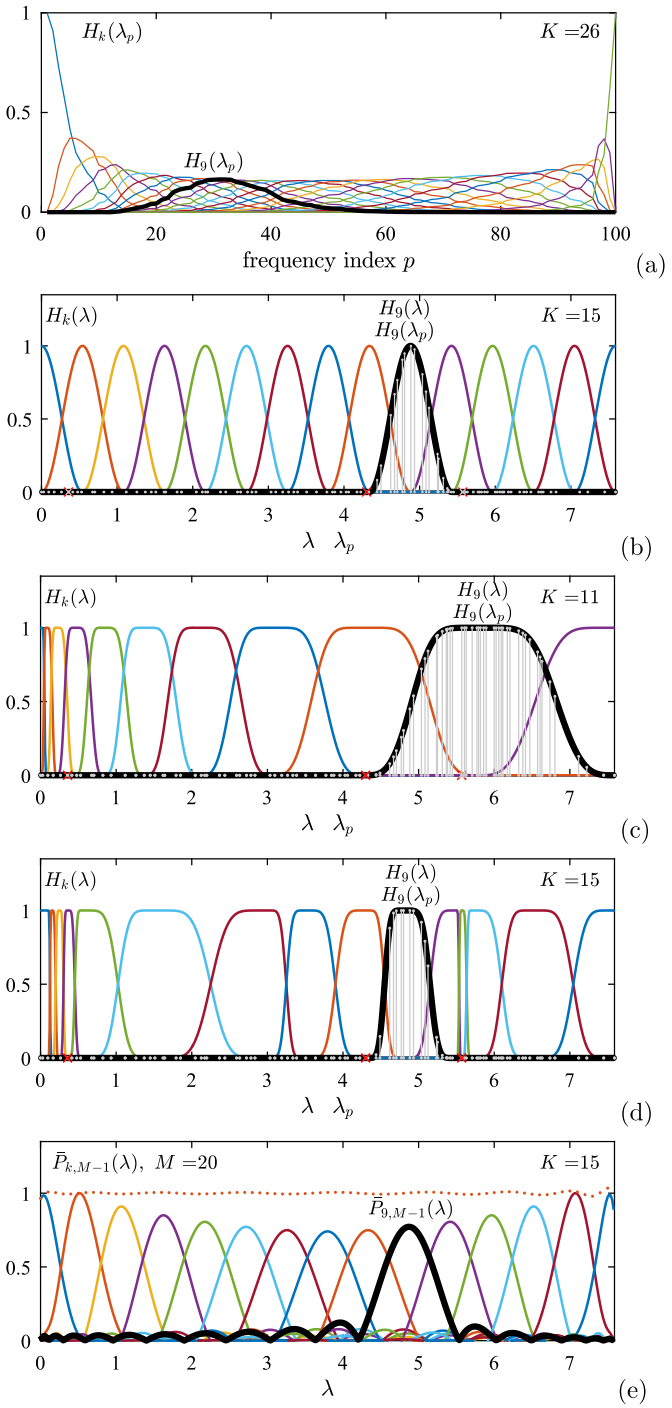

Fig. 4. Exemplar transfer functions in the spectral domain. (a) The spectral domain transfer functions $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$, which correspond to the binomial form terms for $K = 26$. (b) The transfer functions $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$, which correspond to the half-cosine form terms for $K = 15$. (c) The spectral index-varying (wavelet-like) transfer functions $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$, which correspond to the half-cosine form terms for $K = 11$. (d) The spectral domain signal adaptive transfer functions $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$, which satisfy the condition $\sum_{k=0}^{K-1} H_k^2(\lambda_p) = 1$, with $K = 17$. (e) Polynomial Chebyshev approximations of transfer functions from panel (b), $\bar{P}_{k,M-1}(\lambda), k = 0, 1, \ldots, K-1$, with $M = 20$, where dotted horizontal line designates $\sum_{k=0}^{K-1} H_k(\lambda)$. The transfer function $H_9(\lambda)$ is designated by the thick black line for each considered domain in panels (a)–(d), while its discrete values at λ_p, $H_9(\lambda_p)$, are shown in gray in panels (b)–(d).

with $a_1 = 0$ and $\lim_{\lambda\to 0}(a_1/\lambda) = 1$. The initial transfer function, $H_0(\lambda)$, is defined using only $0 = b_0 \le \lambda \le c_0 = \lambda_{\max}/K$, while the last transfer function, $H_{K-1}(\lambda)$, is defined using the interval $a_{K-1} < \lambda \le b_{K-1} = \lambda_{\max}$ in (21).

The raised cosine transfer function in (21) satisfies the following condition

$$\sum_{k=0}^{K-1} H_k(\lambda_p) = 1. \tag{23}$$

This is a frequency domain counterpart of the well-known condition for the classical STFT inversion which requires that the lag-window satisfies the *constant overlap-add* relation $\sum_{n=0}^{K-1} w(nR - m) = 1$, where R is the STFT calculation time step. Many other classical window forms satisfy this condition.

3.3.2. Band-pass functions of the sine window form

When the synthesis window (the same as the analysis lag-window) is used in the signal reconstruction then the condition

$$\sum_{k=0}^{K-1} H_k^2(\lambda_p) = 1 \tag{24}$$

is required. If the squares are omitted in (21), then (24) is immediately satisfied. This condition is a frequency domain counterpart of the *weighted overlap-add* method in classical STFT analysis.

A simple way to construct a window (function) for the weighted overlap-add method is to take the square root of the constant overlap-add window (for, example, the sine window as the square root of the Hann window). In this case, the transfer functions become

$$H_k(\lambda) = \begin{cases} \sin\left(\frac{\pi}{2}\left(\frac{a_k}{b_k - a_k}\left(\frac{\lambda}{a_k} - 1\right)\right)\right), & \text{for } a_k < \lambda \le b_k \\ \cos\left(\frac{\pi}{2}\left(\frac{b_k}{c_k - b_k}\left(\frac{\lambda}{b_k} - 1\right)\right)\right), & \text{for } b_k < \lambda \le c_k \\ 0, & \text{elsewhere,} \end{cases} \tag{25}$$

with $a_{k+1} = b_k$, $b_{k+1} = c_k$ and the initial and the last intervals defined as in (22).

3.3.3. Band-pass functions of the Meyer's wavelet form

The sine functions in (25) are not differentiable at the interval-end points, a_k, b_k, and c_k. This property degrades their convergence in the transformation domain. With the aim to make the functions differentiable (and to improve the convergence), the argument

$$\frac{a_k}{b_k - a_k}\left(\frac{\lambda}{a_k} - 1\right) = x,$$

in (25) is commonly mapped as

$$v_x(x) = x^4(35 - 84x + 70x^2 - 20x^3), \tag{26}$$

for $0 \le x \le 1$, producing Meyer's wavelet-like transfer functions [29]

$$H_k(\lambda) = \begin{cases} \sin\left(\frac{\pi}{2} v_x\left(\frac{a_k}{b_k - a_k}\left(\frac{\lambda}{a_k} - 1\right)\right)\right), & \text{for } a_k < \lambda \le b_k \\ \cos\left(\frac{\pi}{2} v_x\left(\frac{b_k}{c_k - b_k}\left(\frac{\lambda}{b_k} - 1\right)\right)\right), & \text{for } b_k < \lambda \le c_k \\ 0, & \text{elsewhere,} \end{cases} \tag{27}$$

with $a_{k+1} = b_k$, $b_{k+1} = c_k$ and the initial and the last intervals defined as in (22). The functions $\sin(\frac{\pi}{2} v_x(x))$ and $\cos(\frac{\pi}{2} v_x(x))$ are continuous and differentiable at the interval-end points ($x = 0$ and $x = 1$). Notice that if we used $v_x(x) = x$ in (27) this would produce the sine functions (25).

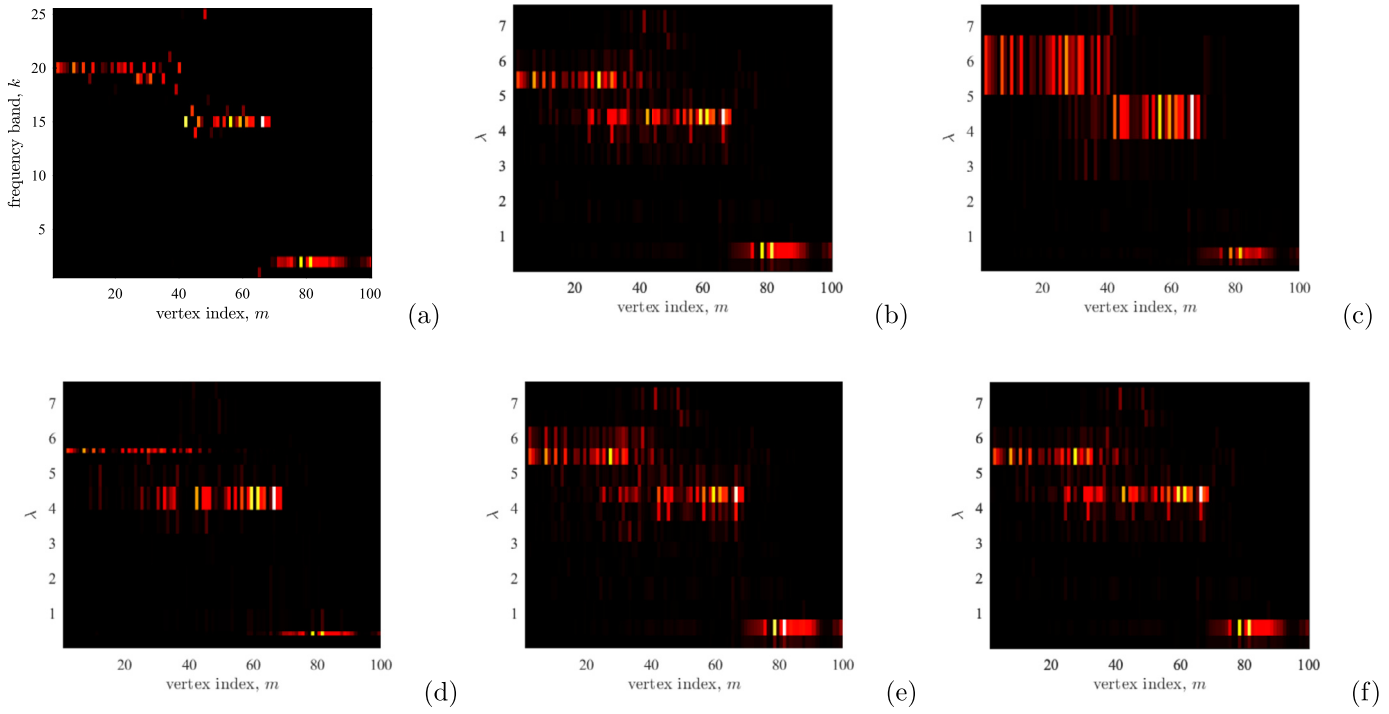

Fig. 5. Vertex-frequency representation of a three-component signal in Fig. 1(d). (a) The LGFT of the signal is calculated by using the transfer functions in Fig. 4(a) for frequency selection. The LGFT values, $S(m,k)$, were reassigned to the position of its maximum value along the frequency band index, k, for each vertex index, m. (b) The LGFT of the signal from Fig. 1(d), calculated by using the transfer functions in Fig. 4(b) for frequency selection. (c) The LGFT of the signal from Fig. 1(d), calculated using the wavelet-like transfer functions in Fig. 4(c) for frequency selection. (d) The LGFT of signal from Fig. 1(d), using signal adaptive transfer functions from Fig. 4(c) for frequency selection. (e) The LGFT of signal from Fig. 1(d), calculated based on the Chebyshev approximation of band-pass transfer functions from Fig. 4(b) with $M = 20$, shown in Fig. 4(e). (f) The LGFT of signal from Fig. 1(d), calculated based on the Chebyshev approximation of band-pass transfer functions from Fig. 4(b) with $M = 50$. LGFT from (e) and (f) illustrate the influence of the convergence of Chebyshev approximation on the final vertex-frequency representation.

The conditions for graph signal reconstruction from the LGFT will be discussed in Section 6 and both forms, (23) and (24), will be used.

Example 4. The experiment from Example 3 is repeated with the raised cosine transfer functions $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$, shown in Fig. 4(b) for $K = 15$. These functions are used for the LGFT calculation at vertex indices $m = 1, 2, \ldots, N$ in $k = 0, 1, \ldots, K-1$ bands. The LGFT values, appropriately reassigned to each eigenvalue, λ_p, are given in Fig. 5(b). The frequency resolution is defined by the transfer functions width and can be improved if narrower transfer functions are used. However, that would be at the expense of the vertex resolution.

Example 5. The experiment in Example 4 is extended by considering a spectral index-varying (wavelet-like transform) form of Meyer's transfer functions, $H_k(\lambda_p)$, $p = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$, as depicted in Fig. 4(c). The so-obtained LGFT values are shown in Fig. 5(c). In order to illustrate the change of resolution in this case, the LGFT values were reassigned to each corresponding eigenvalue, λ_p.

The *spectral graph wavelet-like transform* is just an example of varying spectral transfer functions in the LGFT, with the highest spectral resolution (the narrowest spectral wavelet functions) for small values of the smoothness index, λ_p. As expected, the low-frequency component in Fig. 5(c) is highly concentrated in the frequency domain, while the concentration is very poor for the high frequency component. Again, as compared to Fig. 4(b) the vertex resolution for the high frequency components is significantly improved, and there is no vertex overlapping of the regions, which is significant in Fig. 4(b). Notice that the transfer functions, $H_k(\lambda_p)$, in this example, satisfy the condition in (24) which will be important for the frame-based LGFT inversion.

In general, the change of resolution may be arbitrary and signal adaptive; for example, the resolution may be higher for the spectral intervals of λ which are rich in signal components and lower within the intervals where there are no signal components.

Example 6. The concept of *signal adaptive intervals for transfer functions* of the form (25) is illustrated in Fig. 4(d), where small intervals are defined around a significant spectral content of the signal, while wide intervals are used for λ corresponding to the less significant signal spectral content [30,23]. In the considered case, narrowest window intervals (corresponding to the high resolution in the vertex-frequency representation) are defined around the spectral positions of three signal components, at $\lambda = 0.36$, $\lambda = 4.30$, and $\lambda = 5.57$. Vertex-frequency representation calculated based on these transfer functions is shown in Fig. 5(d), with the assigned eigenvalue as the spectral axis. A notably high resolution in the representation of the first and the third component, at $\lambda = 0.36$ and $\lambda = 5.57$, is the result of the fine interval definition for spectral windows in Fig. 4(d). This particular choice of the intervals allowed for high spectral resolution representation with a small number of transfer functions, $K = 15$. A wider interval width for the second component resulted in a lower spectral resolution than in the case of the other two components.

3.4. Polynomial LGFT approximation

For the direct implementation of the bandpass functions, $H_k(\lambda)$, $k = 0, 1, \ldots, K-1$, for the LGFT, the spectral analysis in (21) or (25) of the whole graph should be performed in order to

obtain the basis functions. For a large graph representing big data problems, it is desirable (sometimes even necessary) to avoid calculation of any function over the whole graph. We will proceed to present an approach based on the polynomial approximation of the bandpass functions, $H_k(\lambda)$, $k = 0, 1, \ldots, K-1$, that will allow vertex-frequency analysis using the local signal and graph connections only.

The bandpass LGFT functions, $H_k(\lambda)$, $k = 0, 1, \ldots, K-1$, of the form (21) or (25) will be implemented using a polynomial approximation of order $(M-1)$, $\bar{P}_{k,M-1}(\lambda)$, $k = 0, 1, \ldots, K-1$. Since the transfer functions $H_k(\lambda)$ are used at discrete points $\lambda = \lambda_p$ and the polynomial approximation is continuous within $0 \leq \lambda \leq \lambda_{\max}$, the natural choice are the so called "min-max" Chebyshev polynomials whose maximal deviation from the desired transfer function is minimal (a review of the orthogonal polynomial approximations of the graph spectral functions can be found in [31]).

The Chebyshev polynomials are defined by

$$T_0(z) = 1,$$

$$T_1(z) = z,$$

$$\vdots$$

$$T_m(z) = 2zT_{m-1}(z) - T_{m-2}(z),$$

for $m \geq 2$ and $-1 \leq z \leq 1$.

We will use the finite $(M-1)$-order of the Chebyshev polynomials approximation

$$\bar{P}_{k,M-1}(\lambda) = \frac{c_{k,0}}{2} + \sum_{m=1}^{M-1} c_{k,m}\bar{T}_m(\lambda), \tag{28}$$

where

$$\bar{T}_m(\lambda) = T_m(2\lambda/\lambda_{\max} - 1)$$

is used in order to map the argument from the interval $0 \leq \lambda \leq \lambda_{\max}$ to the interval from -1 to 1. The polynomial coefficients are calculated using the Chebyshev polynomial inversion property, given by

$$c_{k,m} = \frac{2}{\pi}\int_{-1}^{1} H_k((z+1)\lambda_{\max}/2)T_m(z)\frac{dz}{\sqrt{1-z^2}}.$$

This leads to the vertex domain implementation of the spectral LGFT form in (19), given by

$$\mathbf{s}_k = \bar{P}_{k,M-1}(\mathbf{L})\mathbf{x},$$

for $k = 0, 1, \ldots, K-1$, with

$$\bar{P}_{k,M-1}(\mathbf{L}) = \frac{c_{k,0}}{2} + \sum_{m=1}^{M-1} c_{k,m}\bar{T}_m(\mathbf{L}), \tag{29}$$

$$= h_{0,k}\mathbf{I} + h_{1,k}\mathbf{L} + h_{2,k}\mathbf{L}^2 + \cdots + h_{(M-1),k}\mathbf{L}^{M-1}.$$

The polynomial form in (29) uses only the $(M-1)$-neigh-borhood in the calculation of the LGFT for each considered vertex, without the need for eigendecomposition analysis, thus significantly reducing the computational cost.

This implementation is of special interest when the graph represents big data. In that case, it is possible to perform vertex-frequency analysis without even storing the whole graph Laplacian into computer memory. For vertex-frequency analysis at a vertex n, the calculation of the largest power of the Laplacian $\mathbf{L}^{M-1}$ includes only the M neighborhood of the vertex n. Thus, to calculate the vertex-frequency representation at a vertex n, it is sufficient to store a very reduced set of data related to the M neighborhood of the considered vertex, both for the signal and the graph Laplacian.

Table 2
Coefficients, $h_{i,k}$, $i = 0, 1, \ldots, M-1$, $k = 0, 1, \ldots, K-1$, for the polynomial calculation of the LGFT, $\mathbf{s}_k$, of a signal, $\mathbf{x}$, in various spectral bands, k, for $(M-1) = 5$ and $K = 10$.

$\mathbf{s}_k = (h_{0,k}\mathbf{I} + h_{1,k}\mathbf{L} + h_{2,k}\mathbf{L}^2 + h_{3,k}\mathbf{L}^3 + h_{4,k}\mathbf{L}^4 + h_{5,k}\mathbf{L}^5)\mathbf{x}$						
k	$h_{0,k}$	$h_{1,k}$	$h_{2,k}$	$h_{3,k}$	$h_{4,k}$	$h_{5,k}$
0	1.079	−1.867	1.101	−0.2885	0.03458	−0.001548
1	−0.053	1.983	−1.798	0.5744	−0.07722	0.003723
2	−0.134	0.763	−0.310	0.0222	0.00422	−0.000460
3	0.050	−0.608	0.900	−0.3551	0.05348	−0.002762
4	0.096	−0.726	0.768	−0.2475	0.03172	−0.001424
5	0.016	−0.013	−0.128	0.1047	−0.02231	0.001424
6	−0.073	0.616	−0.779	0.3228	−0.05135	0.002762
7	−0.051	0.351	−0.356	0.1146	−0.01323	0.000460
8	0.084	−0.687	0.871	−0.3751	0.06409	−0.003723
9	−0.021	0.183	−0.251	0.1172	−0.02196	0.001419

Example 7. Consider the shifted transfer functions, $H_k(\lambda)$, $k = 0, 1, \ldots, K-1$, defined by (21) and (22), shown in Fig. 4(b), for $K = 15$. Functions $H_k(\lambda)$ satisfy $\sum_{k=0}^{K-1} H_k(\lambda) = 1$. Every individual transfer function, $H_k(\lambda)$, is approximated by $\bar{P}_{k,M-1}(\lambda)$, $M \geq 2$, of the form (28). Two orders of the polynomial are considered: $M = 20$ and $M = 40$. The Chebyshev polynomial approximations of the band-pass functions from Fig. 4(b) are shown in Fig. 4(e), for $M = 20$. For convenience, the summation $\sum_{k=0}^{K-1} \bar{P}_{k,M-1}(\lambda)$ is also calculated. As designated by the horizontal dotted line in Fig. 4(e), the summation values are close to 1, thus indicating the stable invertibility of the LGFT.

The approximations of transfer functions, $H_k(\lambda)$, obtained in this way, are used for the LGFT based vertex-frequency analysis of the three-component signal from Fig. 1(d). The absolute LGFT values are presented in Fig. 5(e) for $M = 20$ and in Fig. 5(f) for $M = 50$ (almost the same as in Fig. 5(b)). The lower component concentration in Fig. 5(d) than in Fig. 5(e) is related to the less precise approximation of the spectral transfer functions for $M = 20$ than for $M = 50$. Notice that high values of the polynomial order, $(M-1)$, increase calculation complexity and require a wide vertex neighborhood in the calculation of the LGFT. For a large order, M, the advantage of local neighborhood calculation is almost lost.

Note that this expansion is very sensitive to the coefficient precision (important to emphasize for possible hardware realizations with finite length registers). The number of four significant decimal places in the presented coefficients were chosen such that the approximation error with these coefficients was $MSE = -20$ dB. In hardware realization, four significant decimal places correspond to 12 bits in the register required for mantissa and its sign. Considering also the exponent, we can conclude that at least 16-bit registers should be used.

Example 8. Chebyshev polynomial approximation of order $(M-1) = 5$ is calculated for the band-pass transfer functions, $H_k(\lambda)$, $k = 0, 1, \ldots, K-1$, of the raised cosine form (21), with $K = 10$. The corresponding coefficients, $h_{i,k}$ for the vertex-domain implementation (19), are given in Table 2. The number of functions, K, is directly related to the desired frequency domain resolution, as can be seen from Fig. 4(b). An increase in the number of transfer functions, K, makes these functions more concentrated. On the other hand, more concentrated functions, $H_k(\lambda)$, require an increased number of terms M for their approximation. The study of the approximation error for a given frequency resolution, defined by K, as a function of the number of approximation terms, M, is performed and results are given in Table 3. The mean squared error (MSE) is calculated from the difference of the desired, $H_k(\lambda)$,

Table 3
Mean squared error (MSE) in the calculation of the Chebychev polynomial approximation of band-pass transfer functions $H_k(\lambda)$, $k = 0, 1, \ldots, K-1$, of the raised cosine form (21), calculated for various resolutions, K and approximation order, M. The MSE is shown in dB.

M	$K=10$	$K=15$	$K=20$	$K=30$	$K=45$	$K=60$
5	−4.59	−3.23	−2.66	−2.14	−1.82	−1.66
10	−9.34	−6.13	−4.67	−3.33	−2.55	−2.19
15	−15.79	−9.45	−6.69	−4.60	−3.36	−2.77
20	−24.89	−12.93	−9.12	−5.96	−4.14	−3.35
30	−35.37	−23.80	−15.40	−9.02	−5.89	−4.58
50	−39.65	−36.31	−32.31	−17.24	−10.07	−7.34
75	−40.13	−39.69	−38.13	−31.82	−16.99	−11.56

and approximated transfer functions, $\bar{P}_{k,M-1}(\lambda)$. Notice that for the transfer functions in Fig. 4(b) and their approximations in Fig. 4(d), calculated for $K = 15$ and $M = 20$, the MSE was equal to -12.93 [dB]. In this case, the approximation does not follow well the original transfer function. From the analysis of the original and approximated forms we concluded that the MSE of at most -15 [dB] is required. For example, for $K = 15$ the number of approximation terms (from the set presented in the table) is $M = 30$.

Note that in the case of the wavelet-like spectral domain transfer functions or signal adaptive transfer functions, the narrow transfer functions require a large number of terms, M. For example, the transfer function corresponding to the resolution for $K = 45$ requires at least $M = 75$.

Calculation complexity. If the number of nonzero elements in the graph Laplacian, $\mathbf{L}$, is $N_{\mathbf{L}}$, then the number of arithmetic operations (additions) to calculate $\mathbf{Lx}$ is of the order of $N_{\mathbf{L}}$. The same number of operations is required to calculate $\mathbf{L}^2\mathbf{x} = \mathbf{L}(\mathbf{Lx})$ using the available $\mathbf{Lx}$. This means that the total number of arithmetic operations (additions) to calculate all $\mathbf{Lx}$, $\mathbf{L}^2\mathbf{x}$,..., $\mathbf{L}^{M-1}\mathbf{x}$ is of the order $MN_{\mathbf{L}}$. Adding these terms requires additional $MN_{\mathbf{L}}$ arithmetic operations (additions), while the calculation of all terms of the form $c_m\mathbf{L}^m\mathbf{x}$ requires an order of $MN_{\mathbf{L}}$ multiplications by constants c_m, $m = 0, 1, \ldots, M-1$. Therefore, to calculate the output graph signal, $\mathbf{y} = \bar{P}_{M-1}(\mathbf{L})\mathbf{x}$, an order of $2MN_{\mathbf{L}}$ additions and $MN_{\mathbf{L}}$ multiplications is needed. Notice that the eigenanalysis of the graph Laplacian, $\mathbf{L}$, requires an order of N^3 arithmetic operations. For large graphs, the advantage in calculation complexity of the vertex domain realization with the polynomial transfer function approximation, $\mathbf{y} = \bar{P}_{M-1}(\mathbf{L})\mathbf{x}$, is obvious. Moreover, the calculation is possible for every specific vertex of interest using its M-neighborhood only. An average number of operations for each vertex is obtained if the previous total numbers of arithmetic operations are divided by N.

3.5. The LGFT and the graph wavelet transform

The classic wavelet analysis is based on defining the "mother wavelet" and using its dilatated and translated versions to create signal decomposition kernels. A direct extension of this concept is not possible on graphs as irregular signal domains, since the operations of dilatation and translation are not possible in the same way as in the case of simple regularly sampled line as the signal domain. Several attempts have been made to extend the classical wavelet analysis to general graph signals, some of which were performed on specific tree graphs, [32,33]. The most significant attempts to define the wavelet transform on general graphs have been: a lifting-based approach for multi-scale representation of graph signals [34–36], diffusion-based wavelets and diffusion based polynomial frames [37,38], and separable filter-bank wavelets [39]. The wavelet definition that can be directly related to the presented spectral domain local graph Fourier transform, and has been commonly used in the graph signal analysis, is based on the extension of the spectral domain form of the classical wavelet transform and its polynomial approximations, and was introduced in [21].

In classical signal processing theory, time-frequency analysis has many common goals with the wavelet transform (and its generalization in the form of time-scale analysis). However, these two areas are usually considered separately. The main goals of the wavelet analysis are in performing multi-resolution signal analysis, compression, and signal processing, including the wavelet domain sparsity-driven signal denoising. The main goals in classical time-frequency analysis are in the spectral and signal parameter estimation (like, for example, the instantaneous frequency), joint time-frequency domain processing, detection, and denoising of nonstationary signals.

Since the same relation between these two signal processing areas can be assumed for the graph signal processing, we shall consider only the spectral wavelet transform, which is directly related to the frequency-varying LGFT and can be considered as a special case of the frequency-varying vertex-frequency analysis, rather than a transform aimed at the graph signal compression and its wavelet-like multi-resolution analysis.

As in classical signal processing, wavelet coefficients can be defined as a *projection of a graph signal onto the wavelet kernel functions*. Assume that the basic form for the wavelet definition in the spectral domain is $H(\lambda_p)$. The wavelet in spectral domain then represents a scaled version of $H(\lambda_p)$ in the scale s_i, $i = 1, 2, \ldots, K-1$, and is denoted by [22,23,40,36,41,42]

$$H_{s_i}(\lambda_p) = H(s_i\lambda_p).$$

Additionally, a low-pass scale (father wavelet) function $G(\lambda_p)$, plays the role of low-pass function, $H_0(\lambda_p)$, in the LGFT. Therefore, a set of discrete scales for the wavelet calculation, denoted by $s \in \{s_1, s_2, \ldots, s_{K-1}\}$, is assumed with the corresponding spectral transfer functions, $H_i(\lambda_p)$ and $G(\lambda_p)$. Now, in the same way as in the case of the kernel form of the LGFT in (12), the graph wavelet transform is defined using the band-pass scaled wavelet kernel, $\psi_{m,s_i}(n)$, instead of the LGFT kernel, $\mathcal{H}_{m,k}(n)$, in (17). This yields

$$\psi_{m,s_i}(n) = \sum_{p=1}^{N} H(s_i\lambda_p)u_p(m)u_p(n), \tag{30}$$

with the wavelet coefficients given by

$$\begin{aligned} W(m, s_i) &= \sum_{n=1}^{N} \psi_{m,s_i}(n)x(n) \\ &= \sum_{n=1}^{N}\sum_{p=1}^{N} H(s_i\lambda_p)x(n)u_p(m)u_p(n) \\ &= \sum_{p=1}^{N} H(s_i\lambda_p)X(p)u_p(m) \end{aligned} \tag{31}$$

what is the counterpart of the LGFT defined by (15).

The Meyer wavelet-like transfer functions in the spectral domain, $H(s_i\lambda_p)$, are defined as in (27)

$$H(s_1\lambda) = \begin{cases} \sin\left(\frac{\pi}{2}v_x\Big(q(s_1\lambda - 1)\Big)\right), & \text{for } 1 < s_1\lambda \le M, \\ 0, & \text{elsewhere.} \end{cases} \tag{32}$$

For $2 \le i \le K-1$ the Meyer-like graph wavelet is given by

$$H(s_i\lambda)=\begin{cases}\sin\left(\frac{\pi}{2}v_x\left(q(s_i\lambda-1)\right)\right), & \text{for } 1<s_i\lambda\le M\\ \cos\left(\frac{\pi}{2}v_x\left(q(\frac{s_i\lambda}{M}-1)\right)\right), & \text{for } M<s_i\lambda\le M^2\\ 0, & \text{elsewhere,}\end{cases} \tag{33}$$

where $q=1/(M-1)$. The argument $v_x(q(s_i\lambda-1))$ is defined by (26) and the scales are related as

$$s_i=s_{i-1}M=s_1M^{i-1}=M^i/\lambda_{\max}.$$

To handle the low-pass spectral components (the interval for λ closest to $\lambda=0$), the low-pass type *scale function*, $G(\lambda)$), is added in the form

$$G(\lambda)=\begin{cases}1, & \text{for } 0\le\lambda\le M/s_K=\lambda_{\max}/M^{K-1}\\ \cos\left(\frac{\pi}{2}v_x\left(q(\frac{s_K\lambda}{M}-1)\right)\right), & \text{for } M<s_K\lambda\le M^2\\ 0, & \text{elsewhere.}\end{cases} \tag{34}$$

The form of the spectral wavelet transfer function is as in Fig. 4(c), with the vertex-frequency representation explained in Example 5.

Remark 6. The number of wavelet transfer functions, K, does not depend on the other wavelet parameters. A large value of K will only increase the number of intervals and the resolution (producing smaller width of the first interval defined by $\lambda_{\max}/M^{K-1}$) toward $\lambda\to 0$.

Remark 7. The wavelet transfer functions, $H(s_i\lambda)$, defined in (32) and (33) including the low-pass scale function, $G(\lambda)$, defined in (34), satisfy the relation

$$\sum_{i=1}^{K-1}H^2(s_i\lambda)+G^2(\lambda)=1.$$

The associated spectral wavelet transfer function, $H(s_1\lambda)=H_1(\lambda)$, corresponds to the highest frequency band. The wavelet transfer function in the scale, s_{K-1}, $H(s_{K-1}\lambda)=H_{K-1}(\lambda)$, is associated with the lowest wavelet frequency band. The notation for the spectral scale function (low-pass transfer function complementary to $H(s_{K-1}\lambda)$ within the lowest spectral interval) is $G(\lambda)$. In the LGFT, the spectral scale function, $G(\lambda)$, plays the role of low-pass transfer function with spectral index 0. Therefore, $(K-1)$ spectral wavelet transfer functions $H(s_i\lambda)$, $i=1,2,\dots,K-1$, along with the scale function $G(\lambda)$, cover exactly K spectral bands, as in the LGFT case.

According to (19), the wavelet transform (31) can be written as

$$\mathbf{w}_i=H_i(\mathbf{L})\mathbf{x}, \tag{35}$$

where $\mathbf{w}_i$ is a column vector with elements $W(m,s_i)$, $m=1,2,\dots,N$.

In implementations, we can use the vertex domain localized polynomial approximations of the spectral wavelet functions, in the same way as described in Section 3.4. If $H_i(\lambda)=H(s_i\lambda)$ is approximated by an $(M-1)$-order Chebyshev polynomial in λ,

$$H_i(\lambda)\approx\bar{P}_{i,M-1}(\lambda),$$

then the relation

$$\mathbf{w}_i\approx\bar{P}_{i,M-1}(\mathbf{L})\mathbf{x}, \tag{36}$$

follows, where $\bar{P}_{i,M-1}(\mathbf{L})$ is a polynomial in the graph Laplacian (Section 3.4).

3.6. Windows defined using the vertex neighborhood

The window, $h_m(n)$, localized at a vertex m can also be defined using the vertex neighborhood. Recall that the distance between vertices m and n, d_{mn}, is equal to the length of the shortest walk from vertex m to vertex n, and that d_{mn} are integers. Then, the window function can be defined as

$$h_m(n)=g(d_{mn}),$$

where $g(d)$ corresponds to any basic window function in classical signal processing. For example, the Hann window can be used, which is defined as

$$h_m(n)=\frac{1}{2}(1+\cos(\pi d_{mn}/D)),\ \text{for } 0\le d_{mn}<D,$$

where D is the assumed window width.

For convenience, window functions for every vertex can be calculated in a matrix form as follows:

- The vertices for which the distance is $d_{mn}=1$ are defined with an adjacency (neighborhood one) matrix $\mathbf{A}_1=\mathbf{A}$. The vertices which belong to the one-neighborhood of a vertex, m, are indicated by the unit-value elements in the mth row of the adjacency matrix $\mathbf{A}$ (in unweighted graphs). In weighed graphs, the corresponding adjacency matrix $\mathbf{A}$ can be obtained from the weighting matrix $\mathbf{W}$ as $\mathbf{A}=\text{sign}(\mathbf{W})$.
- Vertices m and n, for which the distance is $d_{mn}=2$ are defined by the following matrix

  $$\mathbf{A}_2=(\mathbf{A}\odot\mathbf{A}_1)\circ(\mathbf{1}-\mathbf{A}_1)\circ(\mathbf{1}-\mathbf{I}),$$

  where $\odot$ is the logical (Boolean) matrix product, $\circ$ is the Hadamard (element-by-element) product, and $\mathbf{1}$ is a matrix with all elements equal to 1. The mth row of matrix $\mathbf{A}\odot\mathbf{A}_1$ gives information about all vertices that are connected to the vertex m with walks of length $K=2$ or lower. It should be mentioned that the element-by-element multiplication of $\mathbf{A}\odot\mathbf{A}_1$ by matrix $(\mathbf{1}-\mathbf{A}_1)$ removes the vertices connected with walks of length 1, while the multiplication by $(\mathbf{1}-\mathbf{I})$ removes its diagonal elements.
- For $d_{mn}=d\ge 2$, we arrive at a recursive relation for the calculation of a matrix which will give the information about the vertices separated by distance d. Such a matrix has the form

  $$\mathbf{A}_d=(\mathbf{A}\odot\mathbf{A}_{d-1})\circ(\mathbf{1}-\mathbf{A}_{d-1})\circ(\mathbf{1}-\mathbf{I}). \tag{37}$$

The window matrix for an assumed graph window width, D, can now be defined as

$$\mathbf{P}_D=g(0)\mathbf{I}+g(1)\mathbf{A}_1+\cdots+g(D-1)\mathbf{A}_{D-1},$$

so that a graph signal, localized around vertex m, may be formed based on this matrix, as

$$x_m(n)=h_m(n)x(n)=P_D(n,m)x(n).$$

The LGFT representation of a graph signal, $x(n)$, then becomes

$$S(m,k)=\sum_{n=1}^{N}x(n)h_m(n)\,u_k(n)=\sum_{n=1}^{N}x(n)P_D(n,m)\,u_k(n), \tag{38}$$

with the vertex-frequency kernel given by

$$\mathcal{H}_{m,k}(n)=h_m(n)u_k(n)=P_D(n,m)u_k(n). \tag{39}$$

This allows us to arrive at the matrix form of the LGFT, given by

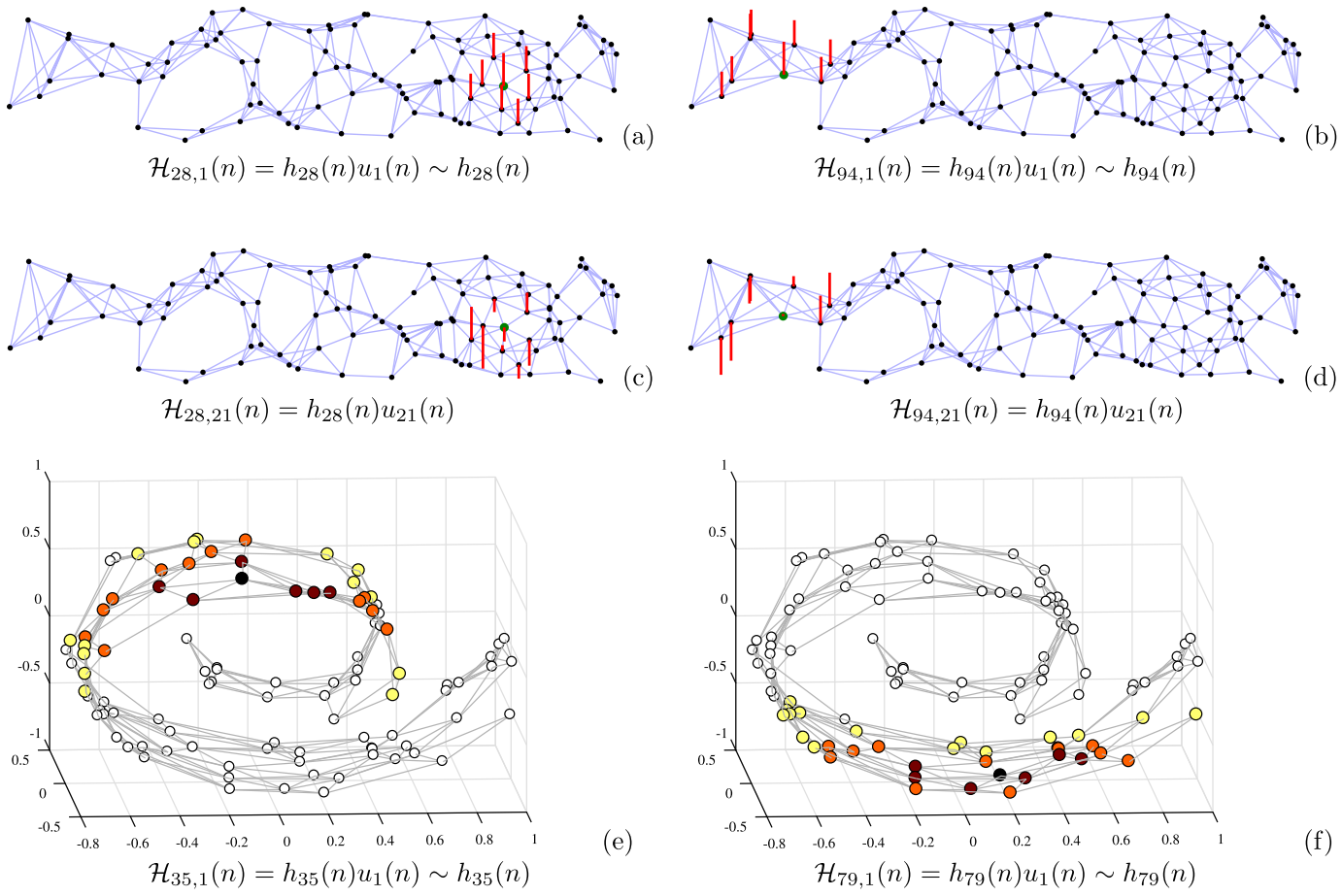

Fig. 6. Localization kernels for vertex-frequency analysis, $\mathcal{H}_{m,k}(n) = h_m(n)u_k(n)$, for the case of *vertex domain defined windows* in the local graph Fourier transform, $S(m,k) = \sum_{n=1}^{N} x(n)\mathcal{H}_{m,k}(n)$. (a) Localization kernel $\mathcal{H}_{28,1}(n) = h_{28}(n)u_1(n) \sim h_{28}(n)$, for a constant eigenvector, $u_1(n) = 1/\sqrt{N}$, centered at the vertex $m = 28$. (b) The same localization kernel as in (a), but centered at the vertex $m = 94$. (c) Localization kernel, $\mathcal{H}_{28,21}(n) = h_{28}(n)u_{21}(n)$, centered at the vertex $m = 28$ and frequency shifted by $u_{21}(n)$. Observe kernel amplitude variations, which indicate modulation of the localization window, $h_m(n)$. (d) The same localization kernel as in (c), but centered at the vertex $m = 94$. (e) Three-dimensional representation of the kernel $\mathcal{H}_{35,1}(n) = h_{35}(n)u_1(n)$. (f) Three-dimensional representation of the kernel $\mathcal{H}_{79,1}(n) = h_{79}(n)u_1(n)$.

$$\mathbf{S} = \mathbf{U}^T(\mathbf{P}_D \circ [\mathbf{x},\ \mathbf{x},\ \ldots,\ \mathbf{x}]), \tag{40}$$

where $[\mathbf{x},\ \mathbf{x},\ \ldots,\ \mathbf{x}]$ is an $N \times N$ matrix, the columns of which are the signal vector $\mathbf{x}$.

For a rectangular function, $g(d) = 1$, and for any $d < D$, the LGFT can be calculated recursively with respect to the window width, D, as

$$\mathbf{S}_D = \mathbf{S}_{D-1} + \mathbf{U}^T(\mathbf{A}_{D-1} \circ [\mathbf{x},\ \mathbf{x},\ \ldots,\ \mathbf{x}]). \tag{41}$$

Example 9. Consider the local vertex-frequency representation of the signal from Fig. 1, using vertex domain defined windows. The localization kernels, $\mathcal{H}_{m,k}(n) = h_m(n)u_k(n)$, are shown in Fig. 6 for two vertices and two spectral indices. Observe that for the spectral index $k = 1$, the localization kernel is proportional to the localization function, $h_m(n)$, given in Figs. 6(a) and 6(c) for the vertices $m = 28$ and $m = 94$. Frequency modulated forms of these localization functions are shown in Figs. 6(b) and 6(d), for the same vertices and $k = 21$.

The vertex domain window is next used to analyze the graph signal from Fig. 1. The vertex-frequency representation, $S(n,k)$, obtained with the LGFT and the vertex domain localization window is given in Fig. 7. Again, we can observe three graph signal components in three vertex regions. The marginals of $S(n,k)$ are also shown in the right and bottom panels.

4. Window parameter optimization

The concentration of local vertex spectrum representation can be measured using the normalized norm-one [43], as

$$\mathcal{M} = \frac{1}{F}\sum_{m=1}^{N}\sum_{k=1}^{N}|S(m,k)| = \frac{1}{F}\|\mathbf{S}\|_1, \tag{42}$$

where $F = \|\mathbf{S}\|_F = \sqrt{\sum_{m=1}^{N}\sum_{k=1}^{N}|S(m,k)|^2}$ is the Frobenius norm of matrix $\mathbf{S}$. Any other norm $\|\mathbf{S}\|_p^p$ with $0 \le p \le 1$ can be used instead of $\|\mathbf{S}\|_1$. Recall that norms with p close to 0 are noise sensitive, while the norm with $p = 1$ is the only convex norm, thus allowing for gradient based optimization [43].

Example 10. The concentration measure $\mathcal{M}(\tau) = \|\mathbf{S}\|_1/\|\mathbf{S}\|_F$ for the signal from Fig. 1 and the window given in (7), and over a range τ is shown in Fig. 8, along with the optimal vertex frequency representation. This representation is similar that shown in Fig. 3, where an empirical value of $\tau = 3$ was used, with the same localization window and kernel form.

The optimal τ can be obtained in only a few steps through the iteration

$$\tau_k = \tau_{k-1} - \alpha(\mathcal{M}(\tau_{k-1}) - \mathcal{M}(\tau_{k-2})),$$

with α as a step-size parameter.

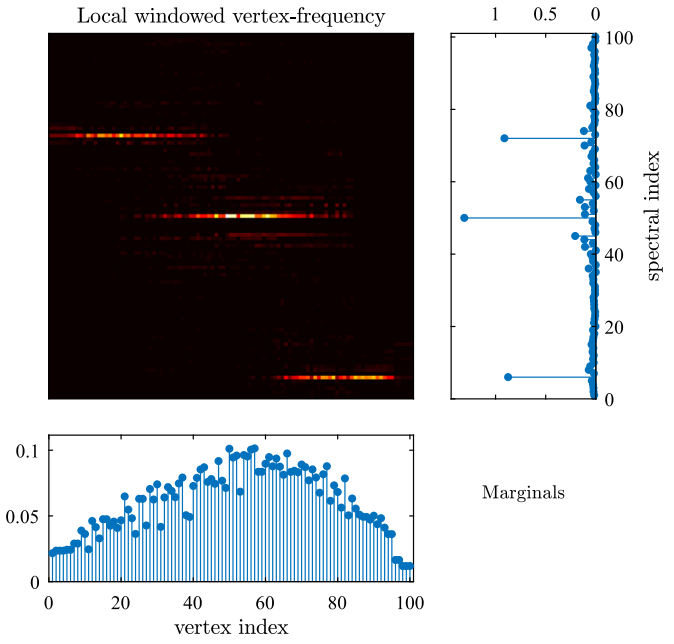

Fig. 7. Local vertex-frequency spectrum calculated using the LGFT and the vertex neighborhood windows, as in (39). This representation shows that the graph signal consists of three components located at spectral indices $k = 72$, $k = 50$, and $k = 6$, with the corresponding vertex indices in their respective vertex subsets $\mathcal{V}_1$, $\mathcal{V}_2$, and $\mathcal{V}_3$, where $\mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{V}_3 = \mathcal{V}$. Marginal properties are also given in the panels to the right and below the vertex-frequency representation, and they differ from ideal ones given respectively by $|x(n)|^2$ and $|X(k)|^2$.

The optimization of parameter τ can also be achieved using graph uncertainty principle based techniques [44–46].

5. Uncertainty principle of graph signals

In the classical signal analysis, the purpose of a window function is to enhance signal localization in the joint time-frequency domain. However, the uncertainty principle prevents an ideal localization in both time and frequency. Various forms of the uncertainty principle in the signal analysis have been defined, with surveys in [47,48]. These forms are closely related to the concentration measures in time-frequency distributions, whose review is given in [43]. While the common uncertainty principle form in time-frequency analysis is the one which establishes the lower bound for the product of effective signal widths (variances) in the time and the frequency domain [13,49] (whose quantum mechanical form is called the Robertson-Schrödinger inequality), here we will use a form of the sparsity support measure [43,47] as the one which clearly and simply shows a significant difference in classical Fourier based analysis and graph signal transforms with respect to the expected concentration in the joint vertex-frequency domain.

The sparsity support uncertainty principle for a discrete-time signal, $\mathbf{x}$, and its DFT, $\mathbf{X}$, states that

$$\|\mathbf{x}\|_0 \|\mathbf{X}\|_0 \geq N, \tag{43}$$

or in other words, that the product of the number of nonzero signal values, $\|\mathbf{x}\|_0$, and the number of its nonzero DFT coefficients, $\|\mathbf{X}\|_0$, is greater or equal than the total number of signal samples N; they cannot simultaneously assume small values.

To arrive at the *support uncertainty principle for graph signals*, consider a graph signal, $\mathbf{x}$, and its spectral transform, $\mathbf{X}$, in a domain of orthonormal basis functions, $u_k(n)$. Then, the uncertainty principle states that [47,44–46,50]

$$\|\mathbf{x}\|_0 \|\mathbf{X}\|_0 \geq \frac{1}{\max_{k,m}\{|u_k(m)|^2\}}. \tag{44}$$

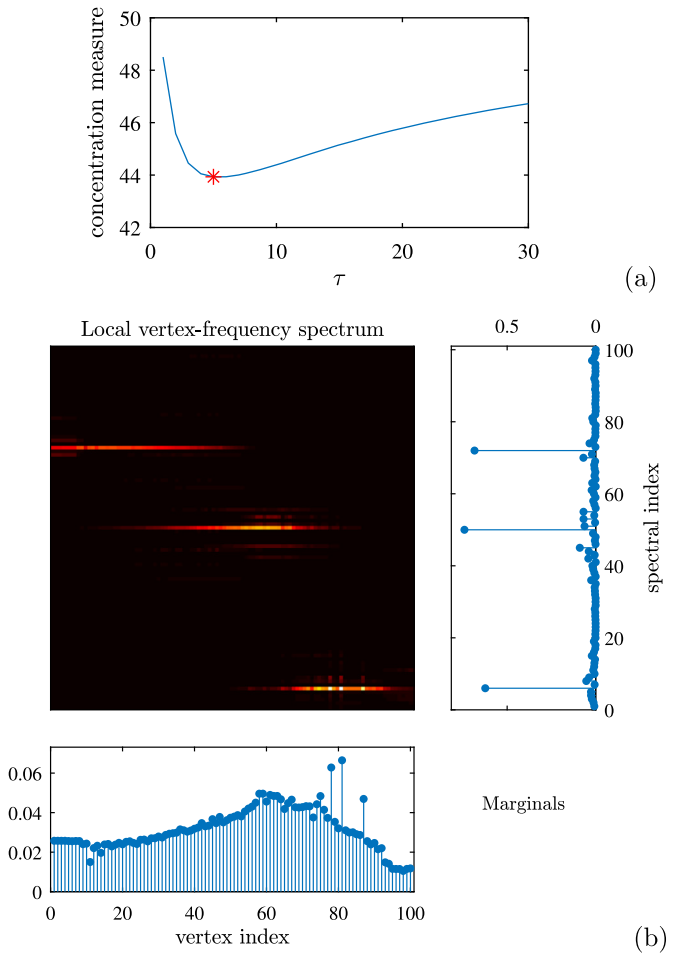

Fig. 8. Principle of the optimization of localization window. (a) Measure of the concentration of graph spectrogram for a varying spectral domain window parameter τ with the minimum at $\tau = 5$. (b) The corresponding optimal vertex-frequency representation, calculated with $\tau = 5$, together with the marginals.

This form of the support uncertainty principle is generic, and indeed when the basis functions are defined as $u_k(n) = \frac{1}{\sqrt{N}} \exp(j2\pi(n-1)(k-1)/N)$, the standard DFT uncertainty principle form (43) follows. A simple derivation of the support uncertainty principle shall be given in Section 7.3, [51].

Note that in graph signal processing, the eigenvectors/basis functions can assume quite different forms than in the DFT case. For example, when one vertex is loosely connected with other vertices, then $\max\{|u_k(m)|^2\} \to 1$, and even $\|\mathbf{x}\|_0 \|\mathbf{X}\|_0 \geq 1$ is possible for the condition in (44). This means that a graph signal can be well localized in both the vertex and the spectral domains.

Example 11. For the graph shown in Fig. 1, we have

$$\max_{k,m}\{|u_k(m)|^2\} = 0.8565$$

which indicates that even $\|\mathbf{x}\|_0 \|\mathbf{X}\|_0 \geq 1.1675$ is possible. In other words, a graph signal for which the number of nonzero samples, $x(n)$, in the vertex domain is just two, will not violate the uncertainty principle even if it has just one nonzero GFT coefficient, $X(k)$.

6. Inversion of the LGFT

The inversion relation of the LGFT, calculated using any of the presented localization (window) forms, can be considered in a uni-

fied way. Two approaches for the LGFT inversion are presented next.

6.1. Inversion by summation of the LGFT

The reconstruction of a signal, $x(n)$, from its local spectrum, $S(m,k)$, can be performed through an inverse GFT of (9), for the graph windowed signal, $x(n)h_m(n)$, as

$$x(n)h_m(n) = \sum_{k=1}^{N} S(m,k)\,u_k(n) \tag{45}$$

followed by a summation over all vertices, m, to yield constant overlap-add relation

$$x(n) = \frac{1}{\sum_{m=1}^{N} h_m(n)} \sum_{m=1}^{N}\sum_{k=1}^{N} S(m,k)u_k(n). \tag{46}$$

Remark 8. If the graph windows, $h_m(n)$, for every vertex, n, satisfy the condition

$$\sum_{m=1}^{N} h_m(n) = 1,$$

then the reconstruction does not depend on the vertex index, n, that is, the reconstruction is vertex independent. In that case

$$x(n) = \sum_{m=1}^{N}\sum_{k=1}^{N} S(m,k)u_k(n) = \sum_{k=1}^{N} X(k)u_k(n), \tag{47}$$

where $X(k) = \sum_{m=1}^{N} S(m,k)$ is a projection of the LGFT onto the spectral index axis.

For windows obtained using the generalized graph shift in (37), this condition is always satisfied since $H(1) = 1$.

The condition $\sum_{m=1}^{N} h_m(n) = 1$ can be enforced by normalizing the elements of the matrix $\mathbf{A}_d$, $d = 1, 2, \ldots, D-1$ in (37), prior to the calculation of matrix $\mathbf{P}_D$ in such a way that the sum for each column is equal to 1, to arrive at

$$\sum_{m=1}^{N} h_m(n) = \sum_{m=1}^{N} P_D(n,m) = \sum_{d=1}^{D-1} g(d) = const.$$

In general, the local vertex spectrum, $S(m,k)$, can be calculated over a reduced set of vertices, $m \in \mathcal{M} \subset \mathcal{V}$. In this case, the summation over m in the reconstruction formula should be executed over only the vertices $m \in \mathcal{M}$, while vertex-independent reconstruction is achieved if $\sum_{m\in\mathcal{M}} h_m(n) = 1$.

6.2. Inversion of the LGFT with band-pass functions

For the LGFT, defined in (19) as $\mathbf{s}_k = \sum_{p=0}^{M-1} h_{p,k}\mathbf{L}^p\mathbf{x}$, the inversion is obtained by a summation over all spectral index shifts, k, that is

$$\sum_{k=0}^{K-1} \mathbf{s}_k = \sum_{k=0}^{K-1}\sum_{p=0}^{M-1} h_{p,k}\mathbf{L}^p\mathbf{x} = \sum_{k=0}^{K-1} H_k(\mathbf{L})\mathbf{x} = \mathbf{x}, \tag{48}$$

if $\sum_{k=0}^{K-1} H_k(\mathbf{L}) = \mathbf{I}$. This condition is equivalent to the following spectral domain condition

$$\sum_{k=0}^{K-1} H_k(\mathbf{\Lambda}) = \mathbf{I}$$

since $\mathbf{U}\sum_{k=0}^{K-1} H_k(\mathbf{\Lambda})\mathbf{U}^T = \mathbf{I}$ and $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. This condition is used to define the transfer functions in Fig. 4(a) and its element-wise form is given by (23).

6.3. Kernel-based LGFT inversion

Another approach to the inversion of the local vertex spectrum, $S(m,k)$, follows the Gabor expansion frameworks [12], whereby the local vertex spectrum, $S(m,k)$, is projected back to the vertex-frequency localized kernels, $\mathcal{H}_{m,k}(n)$. The inversion for two forms of the LGFT, defined as in (10) and (15), will be next analyzed.

(a) For the LGFT defined in (10), the sum of all of its projections to the localized kernels, $\mathcal{H}_{m,k}(n)$, is

$$\begin{aligned}
&\sum_{m=1}^{N}\sum_{k=1}^{N} S(m,k)\mathcal{H}_{m,k}(n) = \sum_{m=1}^{N}\left(\sum_{k=1}^{N} S(m,k)h_m(n)u_k(n)\right)\\
&= \sum_{m=1}^{N}\left(\sum_{i=1}^{N} \underset{k\to i}{\text{IGFT}}\{S(m,k)\}\underset{k\to i}{\text{IGFT}}\{h_m(n)u_k(n)\}\right)\\
&= \sum_{m=1}^{N}\sum_{i=1}^{N}[x(i)h_m(i)][h_m(n)\delta(n-i)]\\
&= \sum_{m=1}^{N} x(n)h_m^2(n) = x(n)\sum_{m=1}^{N} h_m^2(n),
\end{aligned} \tag{49}$$

where IGFT denotes the inverse GFT transform, while Parseval's theorem for graph signals $\sum_{n=1}^{N} x(n)y(n) = \sum_{k=1}^{N} X(k)Y(k)$ was used in the derivation.

The inversion formula for the local vertex spectrum, $S(m,k)$, which yields the original graph signal, $x(n)$, then becomes

$$x(n) = \frac{1}{\sum_{m=1}^{N} h_m^2(n)} \sum_{m=1}^{N}\sum_{k=1}^{N} S(m,k)h_m(n)u_k(n). \tag{50}$$

Remark 9. This kind of inversion is vertex-invariant if the sum over all vertices m is n invariant and equal to 1, that is

$$\sum_{m=1}^{N} h_m^2(n) = 1. \tag{51}$$

If the local vertex spectrum, $S(m,k)$, is calculated over a reduced set of vertices, $m \in \mathcal{M} \subset \mathcal{V}$, then the vertex independent reconstruction condition becomes $\sum_{m\in\mathcal{M}} h_m^2(n) = 1$.

(b) For the LGFT with spectral shifted spectral windows, defined by (15) and (17), the kernel based inversion is given by

$$x(n) = \sum_{m=1}^{N}\sum_{k=0}^{K-1} S(m,k)\mathcal{H}_{m,k}(n) \tag{52}$$

if the following condition, given in (24),

$$\sum_{k=0}^{K-1} H_k^2(\lambda_p) = 1 \tag{53}$$

is satisfied for all λ_p, $p = 1, 2, \ldots, N$.

The inversion formula in (52), with the condition in (53), follows from

$$\sum_{m=1}^{N}\sum_{k=0}^{K-1} S(m,k)\mathcal{H}_{m,k}(n) \tag{54}$$

$$= \sum_{m=1}^{N} \sum_{k=0}^{K-1} \sum_{p=1}^{N} X(p) H_k(\lambda_p) u_p(m) \sum_{l=1}^{N} H_k(\lambda_l) u_l(m) u_l(n).$$

Since $\sum_{m=1}^{N} u_p(m) u_l(m) = \delta(p-l)$, the last expression reduces to the graph signal, $x(n)$, that is

$$\sum_{k=0}^{K-1} \sum_{p=1}^{N} X(p) H_k(\lambda_p) H_k(\lambda_p) u_p(n) = x(n), \tag{55}$$

if the transfer functions, $H_k(\lambda_p)$, $k = 0, 1, \ldots, K-1$, satisfy the condition in (53) for all λ_p.

6.4. The wavelet inversion

The wavelet inversion formula

$$x(n) = \sum_{n=1}^{N} \sum_{i=0}^{K-1} \psi(n, s_i) W(n, s_i) \tag{56}$$

can be derived in the same way as in (52), with the condition in (53) assuming the wavelet transform form

$$G^2(\lambda_p) + \sum_{i=1}^{K-1} H^2(s_i \lambda_p) = 1, \tag{57}$$

where a set of discrete scales for the wavelet calculation, denoted by $s \in \{s_1, s_2, \ldots, s_{K-1}\}$, is assumed with corresponding spectral transfer functions $H(s_i \lambda_p)$, $i = 1, 2, \ldots, K-1$, and a low-pass scale (father wavelet) function $G(\lambda_p)$, playing the role of the low-pass function, $H_0(\lambda_p)$, in the LGFT.

Since the number of wavelet transform coefficients, $W(n, s_i)$, is greater than the number of signal samples N, this representation is redundant, and this redundancy allows us to implement the transform through a fast algorithm, rather than using the explicit computation of all wavelet coefficients [22,23,29]. Indeed, for large graphs, it can be computationally too complex to compute the full eigendecomposition of the graph Laplacian. A common way to avoid this computational burden for large graphs and big data is to use a polynomial approximation schemes for $H(s_i \lambda)$. One such approach is through the truncated Chebyshev polynomial approximations, which admit order recursive calculation, as showed in Section 3.4 and (36). Note that this form corresponds to the LGFT form in (19) and (20).

6.5. Vertex-varying filtering

The filtering in the vertex-frequency domain can be implemented using the vertex-frequency support function $B(m, k)$. The filtered local vertex spectrum is then given by

$$S_f(m, k) = S(m, k) B(m, k)$$

and the filtered signal, $x_f(n)$, is obtained by the inversion of $S_f(m, k)$ using the above mentioned inversion methods. The filtering support function, $B(m, k)$, can be obtained, for example, by thresholding noisy values of the local vertex spectrum, $S(m, k)$.

Example 12. The inversion relation for the case of band-pass transfer functions and the vertex-varying filtering are verified for the case of signal $x(n)$, from Fig. 1(d). This graph signal is corrupted by additive white Gaussian noise, at the signal-to-noise ratio of $SNR_{in} = 5.5$ dB. The LGFT of the noisy graph signal, $S(m, k)$, is calculated based on shifted band-pass spectral transfer functions, $H_k(\lambda_p)$, $k = 0, 1, \ldots, K-1$, $p = 1, 2, \ldots, N$, of the form defined (25), so that the inversion condition in (53) holds. The total number of $K = 25$ frequency shifted transfer functions, $H_k(\lambda_p)$, of the form as in Fig. 4(b) are used. A simple thresholding-based filtering support function $B(m, k) = 1$ if $|S(m, k)| \geq T$, and $B(m, k) = 0$ elsewhere, is used as the basis for the vertex-varying filtering, $S_f(m, k) = S(m, k) B(m, k)$, for $m = 1, 2, \ldots, N$, $k = 0, 1, \ldots, K-1$. The threshold value $T = 0.078$ was set empirically. The output graph signal, $x_f(n)$, is obtained using the inversion relation in (52) for the filtered LGFT, $S_f(m, k)$. The achieved output SNR was $SNR_{out} = 8.94$ dB. (See Fig. 9.)

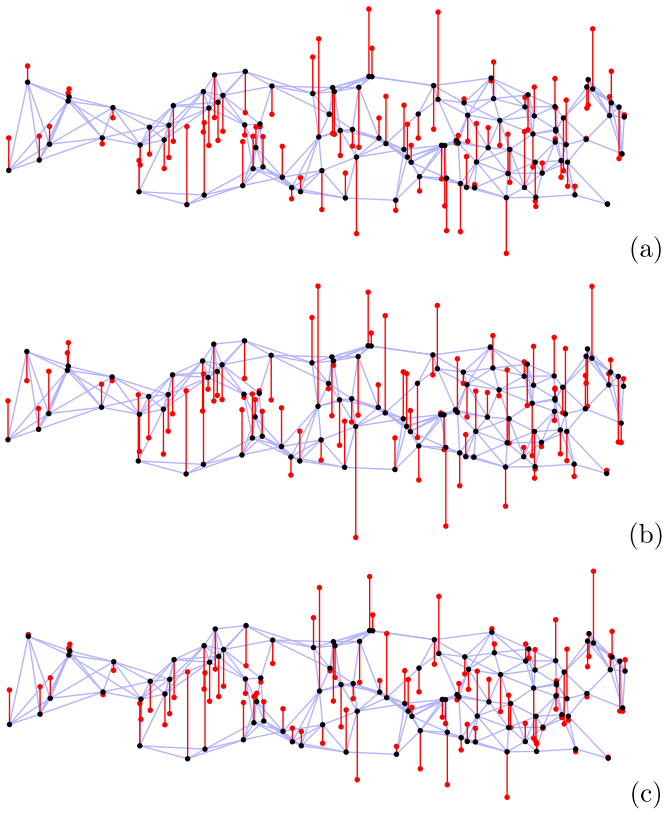

Fig. 9. Vertex-varying filtering of a graph signal. (a) The original graph signal, $x(n)$, from Fig. 1(d). (b) The graph signal, $x(n)$, corrupted by an additive white Gaussian noise, at $SNR_{in} = 5.5$ dB. (c) The resulting graph signal, $x_f(n)$, obtained based on vertex-varying filtering. Filtering is implemented by thresholding the LGFT of noisy graph signal, $S(m, k)$, to produce $SNR_{out} = 8.94$ dB.

A filtering framework for time-varying graph signals may be found in [52].

7. Vertex-frequency energy distributions

Like in time-frequency analysis, the distribution of graph signal energy, as a function of the vertex and spectral indices, is an alternative way to approach vertex-frequency analysis without localization windows. A graph form of the Rihaczek distribution is used as the basic distribution to introduce the concepts of the vertex-frequency domain energy parameters, such as local smoothness and the marginal properties. The graph Rihaczek distribution is then used to derive the support uncertainty principle and to define a class of the reduced interference vertex-frequency energy distributions which satisfy the graph signal marginal properties.

7.1. Graph Rihaczek distribution definition

The energy of a general signal is usually defined as

$$E = \sum_{n=1}^{N} x^2(n) = \sum_{n=1}^{N} x(n) \sum_{k=1}^{N} X(k) u_k(n).$$

This expression can be rearranged as

$$E=\sum_{n=1}^{N}\sum_{k=1}^{N}x(n)X(k)u_k(n)=\sum_{n=1}^{N}\sum_{k=1}^{N}E(n,k),$$

where for each vertex, the vertex-frequency energy distribution, $E(n,k)$, is defined by [53,54]

$$E(n,k)=x(n)X(k)u_k(n)=\sum_{m=1}^{N}x(n)x(m)u_k(m)u_k(n). \tag{58}$$

Remark 10. The definition in (58) corresponds to the Rihaczek distribution in classical time-frequency analysis [12–14]. Observe that based on the Rihaczek distribution and the expression in (58), we may obtain a vertex-frequency representation even without a localization window. This very important property is also the main advantage (along with the concentration improvement) of classical time-frequency distributions with respect to the spectrogram and STFT based time-frequency representations.

The marginal properties of the vertex-frequency energy distribution, $E(n,k)$, are defined as its projections onto the spectral index axis, k, and the vertex index axis, m, to give

$$\sum_{n=1}^{N}E(n,k)=|X(k)|^2 \quad \text{and} \quad \sum_{k=1}^{N}E(n,k)=x^2(n).$$

These correspond respectively to the squared spectra, $|X(k)|^2$, and the signal power, $x^2(n)$, of the graph signal, $x(n)$.

The Rihaczek energy condition is

$$\sum_{n=1}^{N}\sum_{k=1}^{N}E(n,k)=\sum_{k=1}^{N}|X(k)|^2=\sum_{n=1}^{N}x^2(n)=E_x=1, \tag{59}$$

where the unit energy of the signal, $E_x=1$, may be assumed without loss of generality,

Example 13. Fig. 10 shows the vertex-frequency distribution, $E(n,k)$, of the graph signal from Fig. 1, together with its marginal properties. The marginal properties are satisfied up to the computer precision, and show that the localization of energy is better than in the cases obtained with the localization windows in Figs. 3, 7, and 8. Importantly, the distribution, $E(n,k)$, does not use a localization window.

7.2. Smoothness index and local smoothness

The *smoothness index* in graph signal processing plays the role of *frequency* in classical spectral analysis, and is defined as the *Rayleigh quotient* of matrix $\mathbf{L}$ and vector $\mathbf{x}$, that is

$$l=\frac{\mathbf{x}^T\mathbf{L}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}\geq 0. \tag{60}$$

Remark 11. The expression in (60) indicates that the smoothness index can be considered as a measure of the rate of change of a graph signal. *Faster changing signals (corresponding to high-frequency signals) have larger values of the smoothness index.* Therefore, the maximally smooth graph signal is a constant signal, $x(n)=c$, for which the smoothness index is $l=0$.

In the mathematics literature, the inverse of the smoothness index is known as the *curvature* (curvature $\sim 1/l$). While larger values of the smoothness index correspond to graph signals with larges rates of change (less smooth graph signals), the larger values of curvature would indicate smoother graph signals.

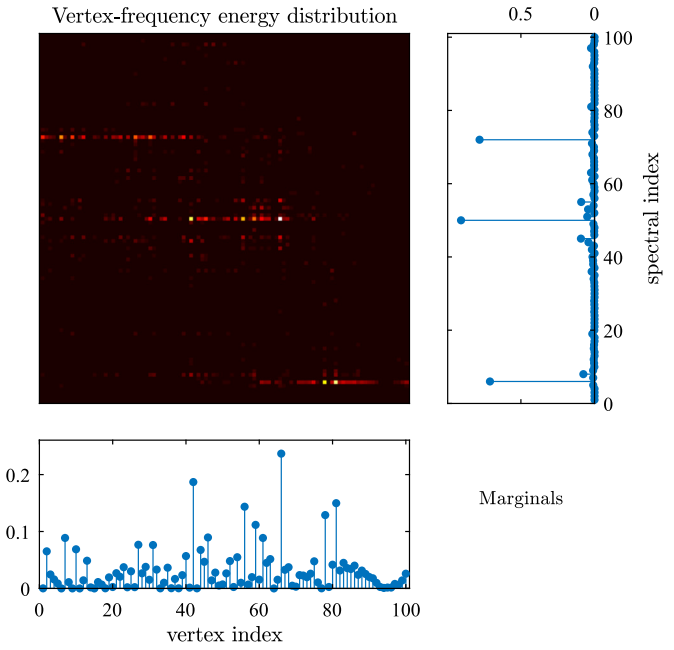

Fig. 10. Vertex-frequency energy distribution for a signal whose vertex-frequency representation is given in Fig. 3. No localization window was used here.

The smoothness index for an eigenvector, $\mathbf{u}_k$, of the graph Laplacian, $\mathbf{L}$, is equal to its corresponding eigenvalue, λ_k, that is

$$\frac{\mathbf{u}_k^T\mathbf{L}\mathbf{u}_k}{\mathbf{u}_k^T\mathbf{u}_k}=\lambda_k, \tag{61}$$

since by definition $\mathbf{L}\mathbf{u}_k=\lambda_k\mathbf{u}_k$.

Remark 12. If the above eigenvectors are the classical Fourier transform basis functions, then the smoothness index corresponds to the squared frequency of the considered basis function, $\lambda_k\sim\omega_k^2$, while the curvature corresponds to the squared period in harmonic signals.

This makes it possible to define the local smoothness index, $\lambda(n)$, for a vertex n, in analogy with the standard instantaneous frequency, $\omega(t)$, at an instant t, as [55]

$$\lambda(n)=\frac{\mathcal{L}_x(n)}{x(n)}, \tag{62}$$

where it was assumed that $x(n)\neq 0$ and $\mathcal{L}_x(n)$ are the elements of the vector $\mathbf{Lx}$.

The properties of the local smoothness index include:

1. The local smoothness index, $\lambda(n)$, for a monocomponent signal

   $$x(n)=\alpha u_k(n),$$

   is vertex independent, and is equal to the global smoothness index, λ_k, since

   $$\mathcal{L}_x(n)=\alpha\mathcal{L}_{u_k}(n)=\alpha\lambda_k u_k(n).$$

   In the time-domain signal analysis, this property means that the instantaneous frequency of a sinusoidal signal is equal to its global frequency.
2. Assume a piece-wise monocomponent signal

   $$x(n)=\alpha_i u_{k_i}(n) \text{ for } n\in\mathcal{V}_i, \quad i=1,2,\ldots,M,$$

where $\mathcal{V}_i \subset \mathcal{V}$ are the subsets of the vertices such that $\mathcal{V}_i \cap \mathcal{V}_j = \emptyset$ for $i \neq j$, $\mathcal{V}_1 \cup \mathcal{V}_2 \cup \cdots \cup \mathcal{V}_M = \mathcal{V}$, that is, every vertex belongs to only one subset $\mathcal{V}_i$. Given the monocomponent nature of the signal, within each subset, the considered signal is proportional to the eigenvector $u_{k_i}(n)$.

Then, for each interior vertex $n \in \mathcal{V}_i$, i.e., a vertex whose neighborhood lies in the same set $\mathcal{V}_i$, the local smoothness index is given by

$$\lambda(n) = \frac{\alpha_i \mathcal{L}_{u_{k_i}}(n)}{\alpha_i u_{k_i}(n)} = \lambda_{k_i}. \tag{63}$$

3. An ideally concentrated vertex-frequency distribution (ideal distribution) can be defined as

$$I(n,k) \sim |x(n)|^2 \delta\Big(\lambda_k - [\lambda(n)]\Big),$$

whereby it was assumed that the local smoothness index is rounded to the nearest eigenvalue.

This distribution can also be used as a local smoothness estimator, since for each vertex, n, the maximum of $I(n,k)$ is positioned at the nearest eigenvalue, $\lambda_k = [\lambda(n)]$. The index of the eigenvalue, $\hat{k}$, that corresponds to the local smoothness index is then obtained as

$$\hat{k}(n) = \arg\max_k \{I(n,k)\},$$

so that the estimated local smoothness becomes $\hat{\lambda}(n) = \lambda_{\hat{k}(n)}$. This estimator is quite common and is widely used in classic time-frequency analysis [12–14].

4. The vertex-frequency distribution, $E(n,k)$, satisfies the *local smoothness property* if

$$\frac{\sum_{k=1}^{N} \lambda_k E(n,k)}{\sum_{k=1}^{N} E(n,k)} = \lambda(n). \tag{64}$$

In that case, the center of masses of vertex-frequency distribution along the spectral index axis, k, should be exactly at $\lambda = \lambda(n)$, and it can be used as an unbiased estimator of this graph signal parameter.

Example 14. The vertex-frequency distribution defined by $E(n,k) = x(n)X(k)u_k(n)$ satisfies the local smoothness property in (64), since

$$\frac{\sum_{k=1}^{N} \lambda_k E(n,k)}{\sum_{k=1}^{N} E(n,k)} = \frac{\sum_{k=1}^{N} \lambda_k x(n)X(k)u_k(n)}{\sum_{k=1}^{N} x(n)X(k)u_k(n)} = \frac{\mathcal{L}_x(n)}{x(n)} = \lambda(n).$$

The above relation follows from the fact that $\sum_{k=1}^{N} \lambda_k X(k)u_k(n)$ are the elements of the IGFT of $\lambda_k X(k)$. Upon employing the matrix form of the IGFT of $\mathbf{\Lambda X}$, we get

$$\mathbf{U\Lambda X} = \mathbf{U\Lambda}(\mathbf{U}^T\mathbf{U})\mathbf{X} = (\mathbf{U\Lambda U}^T)(\mathbf{UX}) = \mathbf{Lx}.$$

With the notation, $\mathcal{L}_x(n)$, for the elements of $\mathbf{Lx}$, we obtain

$$\sum_{k=1}^{N} \lambda_k X(k)u_k(n) = \mathcal{L}_x(n).$$

The local smoothness index for the graph signal from Fig. 1 is presented in Fig. 11.

7.3. Support uncertainty principle derivation [51]

From the Rihaczek distribution energy condition (59), with the unit energy, follows

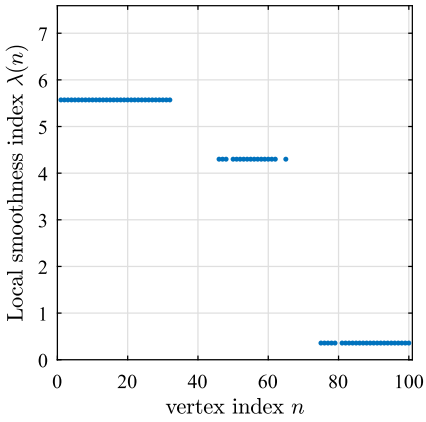

Fig. 11. Local smoothness index, $\lambda(n)$, of graph signal from Fig. 1.

$$1 \leq \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} |E(n,k). \tag{65}$$

Assume, as in [50], that the support $\mathbb{M}$ of the signal $x(n)$ is finite, $\mathbb{M} = \{n_1, n_2, \ldots, n_M\}$, meaning that $x(n) \neq 0$ for $n \in \mathbb{M}$ and $x(n) = 0$ for $n \notin \mathbb{M}$, while the support of the graph Fourier transform $X(k)$ is $\mathbb{K} = \{k_1, k_2, \ldots, k_K\}$, where $X(k) \neq 0$ for $k \in \mathbb{K}$ and $X(k) = 0$ for $k \notin \mathbb{K}$. By definition, we can write the relations

$$||\mathbf{x}||_0 = \text{card}\{\mathbb{M}\} = M \ \text{ and } \ ||\mathbf{X}||_0 = \text{card}\{\mathbb{K}\} = K. \tag{66}$$

Applying the Schwartz inequality to (65) squared, we get

$$1 = \Big(\sum_{n\in\mathbb{M}} \sum_{k\in\mathbb{K}} E(n,k)\Big)^2 \leq \Big(\sum_{n\in\mathbb{M}} \sum_{k\in\mathbb{K}} |x(n)|\,|X(k)|\,|u_k(n)|\Big)^2$$

$$= \Big(\sum_{n\in\mathbb{M}} \sum_{k\in\mathbb{K}} (\sqrt{|u_k(n)|}|x(n)|)\,(\sqrt{|u_k(n)|}|X(k)|)\Big)^2 \tag{67}$$

$$\leq \sum_{n\in\mathbb{M}} \sum_{k\in\mathbb{K}} |u_k(n)||x(n)|^2 \sum_{n\in\mathbb{M}} \sum_{k\in\mathbb{K}} |u_k(n)||X(k)|^2 \tag{68}$$

$$\leq \max_{n,k}\{|u_k(n)|^2\} KM = \max_{n,k}\{|u_k(n)|^2\} ||\mathbf{x}||_0 ||\mathbf{X}||_0, \tag{69}$$

since the unit energy of the graph signal is assumed, that is, $\sum_{n\in\mathbb{M}} |x(n)|^2 = \sum_{k\in\mathbb{K}} |X(k)|^2 = 1$.

The inequality in (69) results in the support uncertainty principle [50]

$$||\mathbf{x}||_0 ||\mathbf{X}||_0 \geq \frac{1}{\max_{n,k}\{|u_k(n)|^2\}}. \tag{70}$$

An improved bound of the support uncertainty principle is recently derived in [51] using the same relations.

7.4. Reduced interference distributions (RID) on graphs

In order to emphasize the relations and the resemblance to the classical time-frequency analysis, in this subsection we will use the complex-sensitive notation for eigenvectors and spectral vectors. The frequency domain definition of the energy distribution in (58) is given by

$$E(n,k) = x(n)X^*(k)u_k^*(n) = \sum_{p=1}^{N} X(p)X^*(k)u_p(n)u_k^*(n).$$

Remark 13. In classical time-frequency analysis the Wigner distribution is used to derive the generalized Cohen class of distributions. Since the Wigner distribution is not appropriate in graph

spectral analysis, the Cohen class of distribution with the Rihaczek distribution,

$$R(t,\omega)=x(t)X^*(\omega)e^{-j\omega t},$$

as the basic distribution will be used [12–14]. Its ambiguity domain form (a two-dimensional Fourier transform of $R(t,\omega)$ over t and ω) is

$$A(\theta,\tau)=\frac{1}{2\pi}\int_u X(u)X^*(u-\theta)e^{j(u-\theta)\tau}du.$$

The Cohen class of distributions, with the Rihaczek distribution as the basic distribution, is defined by

$$C(t,\omega)=\frac{1}{2\pi}\int_\theta\int_\tau A(\theta,\tau)c(\theta,\tau)e^{-j\omega\tau}e^{j\theta t}d\tau d\theta,$$

where $c(\theta,\tau)$ is the kernel function. Using the defined ambiguity domain form of the Rihaczek distribution $A(\theta,\tau)$ we get

$$C(t,\omega)=\frac{1}{4\pi^2}\int_u\int_v X(u)X^*(v)e^{jut}e^{-jvt}$$
$$\times\int_\tau c(u-v,\tau)e^{-j\tau\omega}e^{j\tau v}d\tau du dv. \quad (71)$$

The frequency-frequency domain form of the Cohen class of distributions is

$$C(t,\omega)=\int_u\int_v X(u)X^*(v)e^{jut}e^{-jvt}\phi(u-v,\omega-u)\frac{du dv}{4\pi^2}, \quad (72)$$

where

$$\phi(u-v,\omega-u)=\int_\tau c(u-v,\tau)e^{-j\tau\omega}e^{j\tau v}d\tau.$$

The Rihaczek distribution is then obtained with $\phi(u-v,\omega-u)=2\pi\delta(v-\omega)$ or $c(u-v,\tau)=1$.

The marginal properties are met if the kernel, $c(\theta,\tau)$, satisfies the conditions $c(\theta,0)=1$ and $c(0,\tau)=1$.

In analogy with the classical Cohen class of distributions in (72), the general form of graph distribution can be defined through introducing a kernel, $\phi(p,k,q)$, as [56]

$$G(n,k)=\sum_{p=1}^{N}\sum_{q=1}^{N}X(p)X^*(q)u_p(n)u_q^*(n)\phi(p,k,q). \quad (73)$$

Observe that for $\phi(p,k,q)=\delta(q-k)$, the graph Rihaczek distribution in (58) follows.

The so obtained distribution $G(n,k)$ may also satisfy the vertex and frequency marginal properties, as elaborated next.

- The *vertex marginal property* is satisfied if

  $$\sum_{k=1}^{N}\phi(p,k,q)=1.$$

  This is obvious from

  $$\sum_{k=1}^{N}G(n,k)=\sum_{p=1}^{N}\sum_{q=1}^{N}X(p)X^*(q)u_p(n)u_q^*(n)=|x(n)|^2.$$

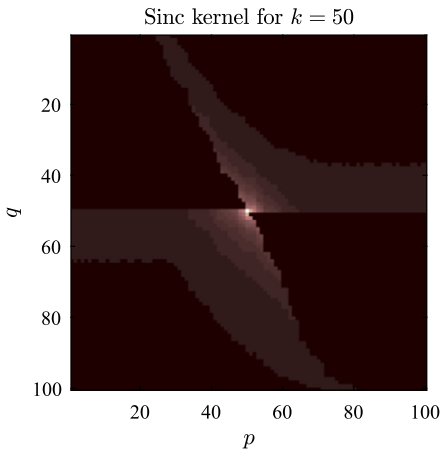

Fig. 12. The sinc kernel of the reduced interference vertex-frequency distribution in the frequency domain.

- The *frequency marginal property* is satisfied if

  $$\phi(p,k,p)=\delta(p-k).$$

  Then, the sum over vertex indices produces

  $$\sum_{n=1}^{N}G(n,k)=\sum_{p=1}^{N}|X(p)|^2\phi(p,k,p)=|X(k)|^2,$$

  since $\sum_{n=1}^{N}u_p(n)u_q^*(n)=\delta(p-q)$, that is, the eigenvectors are orthonormal.

7.5. Reduced interference distribution kernels

A straightforward extension of classical time-frequency kernels to graph signal processing would naturally be based on exploiting the relation $\lambda\sim\omega^2$, together with an appropriate exponential kernel normalization.

The simplest reduced interference kernel in the frequency-frequency shift domain, which would satisfy the marginal properties, is the *sinc kernel*, given by

$$\phi(p,k,q)=\begin{cases}\frac{1}{1+2|p-q|}, & \text{for } |k-p|\le|p-q|,\\ 0, & \text{otherwise}\end{cases}$$

and shown in Fig. 12 at the frequency shift corresponding to $k=50$.

Example 15. The sinc kernel was used for a vertex-frequency representation of the signal from Fig. 1(d), with the results shown in Fig. 13. This representation is a smoothed version of the energy vertex-frequency distribution in Fig. 10, whereby both (vertex and frequency) marginals are preserved.

Remark 14. Graph spectrogram and marginal properties. The general vertex-frequency distribution can be written for the vertex-vertex shift domain as a dual form of (73), and has the form

$$G(n,k)=\sum_{m=1}^{N}\sum_{l=1}^{N}x(m)x^*(l)u_k(m)u_k^*(l)\varphi(m,n,l), \quad (74)$$

where $\varphi(m,n,l)$ is the kernel in this domain (the same mathematical form as for the frequency-frequency shift domain kernel). The frequency marginal is satisfied if $\sum_{n=1}^{N}\varphi(m,n,l)=1$ holds, while the vertex marginal is met if $\varphi(m,n,m)=\delta(m-n)$. The relation of

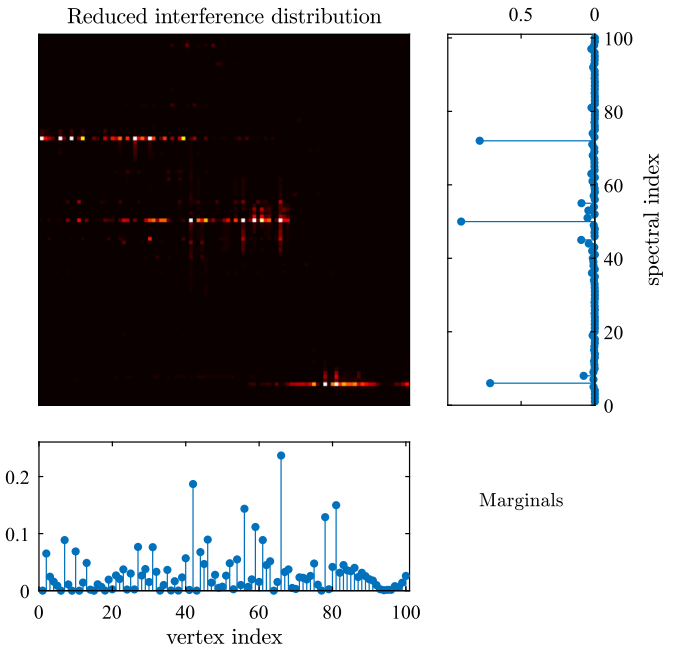

Fig. 13. Reduced interference vertex-frequency distribution of a signal whose vertex-frequency representation is given in Fig. 3. The marginal properties are given in the panels to the right and below the vertex-frequency representation, and are equal to their corresponding ideal forms given by $|x(n)|^2$ and $|X(k)|^2$.

this distribution with the vertex domain spectrogram (3) is simple, and given by

$$\varphi(m,n,l) = h_n(m)h_n^*(l).$$

However, this kernel cannot satisfy both marginal properties, while the unbiased energy condition $\sum_{n=1}^{N}\varphi(m,n,m) = 1$ reduces to (51).

Remark 15. Classical time-frequency analysis follows as a special case from the general form of graph distributions in (73), if the considered graph is a directed circular graph. This becomes obvious upon recalling that eigendecomposition of the adjacency matrix produces complex-valued eigenvectors of the form $u_k(n) = \exp(j2\pi(n-1)(k-1)/N)/\sqrt{N}$ in (73), and the classical (Rihaczek based) Cohen class of distributions follows.

8. Comparison of the presented methods

A summary of all the presented methods, together with their main properties, is as follows.

- The LGFT with a spectrum domain window function, presented in Section 3.1, is defined using the graph shift and modulation based on the basis functions. This is a pure graph form, with a direct link to the classical time-frequency analysis. Both shifts of the kernel are performed by multiplying a predefined frequency domain low-pass function by the basis functions in frequency domain (vertex shift) and in the vertex domain (frequency shift). Although this kernel definition is the most general, it does not guarantee that the kernel is strictly bounded in either the vertex or the spectral domain. In addition, the complete set of basis functions is required for the implementation. This form cannot satisfy the tight frame condition for the kernel-based inversion.
- A spectral shifted form of the kernel using a band-pass type function in the spectral domain is defined in Section 3.2. Instead of the spectral shift by modulation in the vertex domain, the shift is here performed by directly shifting the spectral domain function. Notice that direct shifts do not account for the nonlinear nature of the eigenvalues, while for a high resolution a very narrow spectral functions are required. This form can satisfy both reconstruction conditions using the constant overlap-add relation and weighted overlap-add relation, meaning that it can be a tight frame. This form also allows for polynomial approximations and realizations using polynomials in vertex domain, without basis function calculation. However, for high frequency resolution, the order of polynomials becomes rather high so that a wide neighborhood is used in the vertex domain calculation and the accuracy of the coefficients is crucial.
- The spectral varying forms of the spectral shifted kernels, as in the spectral graph wavelet transform or a signal adaptive kernel, are reviewed in Section 3.5. The property of the previous class of vertex-frequency representations that they do not follow the nonlinear nature of eigenvalues can be alleviated by using signal adaptive functions, including frequency-varying forms that correspond to the spectral graph wavelet transform. This form can provide a good signal-tailored representation, however it is highly dependent on the expected signal form.
- Pure vertex domain forms and graph shifts of the window are elaborated in Section 3.6. This approach relies on the classical window definition using the adjacency matrix. This realization is vertex limited and uses only the considered vertex neighborhood. However, it requires partial basis functions calculation, although only around the considered vertex and frequency index, and can be defined in such a way that the constant overlap-add inversion condition is satisfied.
- Quadratic vertex-frequency distribution of the Rihaczek type, addressed in Section 7, does not use a localization window, and as such has big advantage and high resolution. However, in this case inversion is not straightforward like in the case of the linear signal vertex-frequency representations. Cross-terms may also appear in the case of quadratic distributions. This distribution satisfies the vertex and frequency marginal properties, as well as the local smoothness property.
- Reduced interference quadratic vertex-frequency distributions, in Section 7.4, can keep high resolution as in the case of the Rihaczek type distribution, but with smoothed interferences in the spectral domain using kernels, like in the standard time-frequency domain. These kernels can be defined is a such way that the vertex and the frequency marginal property are preserved. Here, the inversion is even more difficult than in the Rihaczek type distribution and these kinds of distributions are used mainly for the analysis purposes, including graph signal parameters estimation, like in classical analysis.

9. Conclusion

This article has reviewed vertex-frequency analysis, as an approach to the localized analysis of graph signals. Traditional approaches for graph analysis, clustering and segmentation are based only on the graph topology and spectral properties of graphs themselves. However, when dealing with signals on graphs, localized analyzes should be focused on data on graphs, while incorporating the graph topology. This unified approach to define and implement graph signal localization methods, which takes into account both the data on graph and the corresponding graph topology, is at the core of vertex-frequency analysis. Like in classical time-frequency analysis, the main research efforts have been devoted to the graph signal linear representations which include a localization window. Several methods for the definition of localization widows in the spectral and vertex domain have been presented in this review. Optimization of the window parameters, uncertainty principle, and

inversion methods have also been discussed. Following the classical time-frequency analysis, energy forms of vertex-frequency energy and reduced interference distributions, which do not use localization windows, have been considered in the second part of the paper. Their role as local smoothness index estimator is presented.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Classical signal processing and graph signal processing framework

Consider a simple classical discrete-time finite impulse response (FIR) system, whose input-output relation is

$$y(n) = h_0 x(n) + h_1 x(n-1) + h_2 x(n-2) + \cdots + h_M x(n-M).$$

In a classical system, the signal values are given at a well-ordered signal domain, defined by the instants $n = 1$, $n = 2$, ..., $n = N$. If we assume a periodic form of the signal (necessary for the DFT analysis) and $N = 8$ then the signal domain and an arbitrary signal are shown in Fig. 14(a).

Notice that the previous input-output relation can be written in the matrix form as

$$\mathbf{y} = h_0\mathbf{x} + h_1\mathbf{A}\mathbf{x} + h_2\mathbf{A}^2\mathbf{x} + \cdots + h_M\mathbf{A}^M\mathbf{x} = H(\mathbf{A})\mathbf{x}. \tag{75}$$

where

$$\mathbf{y}=\begin{bmatrix} y(1)\\ y(2)\\ y(3)\\ \vdots\\ y(N)\end{bmatrix},\ \mathbf{x}=\begin{bmatrix} x(1)\\ x(2)\\ x(3)\\ \vdots\\ x(N)\end{bmatrix},\ \mathbf{A}=\begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 1\\ 1 & 0 & 0 & \dots & 0 & 0\\ 0 & 1 & 0 & \dots & 0 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & 0 & \dots & 1 & 0\end{bmatrix}.$$

The adjacency matrix $\mathbf{A}$ is an instant (vertex) connectivity matrix for the signal domain, with elements $A_{mn} = 1$ if the vertex m is connected (predecessor, in this case) to the vertex n, and $A_{mn} = 0$ otherwise, as shown in Fig. 14. The element-wise form of $\mathbf{y} = \mathbf{A}\mathbf{x}$ is $y(n) = x(n-1)$, as expected.

If the eigendecomposition of the square matrix $\mathbf{A}$ is performed according to the definition

$$\mathbf{A}\mathbf{u}_k = \lambda_k \mathbf{u}_k$$

or in the matrix form

$$\mathbf{A}\mathbf{U} = \mathbf{U}\mathbf{\Lambda} \ \text{ or } \ \mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$$

where $\mathbf{U}$ is the matrix of eigenvectors, $\mathbf{u}_k$, as its columns and $\mathbf{\Lambda}$ is a diagonal matrix with the eigenvalues λ_k on its diagonal, then we can write the input-output relation (75) in the form

$$\mathbf{y} = h_0\mathbf{x} + h_1\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}\mathbf{x} + \cdots + h_M\mathbf{U}\mathbf{\Lambda}^M\mathbf{U}^{-1}\mathbf{x},$$

where the eigendecomposition property

$$\mathbf{A}^M = \mathbf{U}\mathbf{\Lambda}^M\mathbf{U}^{-1}$$

is used. Now, by left-multiplication by $\mathbf{U}^{-1}$ we can write

$$\mathbf{U}^{-1}\mathbf{y} = h_0\mathbf{U}^{-1}\mathbf{x} + h_1\mathbf{\Lambda}\mathbf{U}^{-1}\mathbf{x} + \cdots + h_M\mathbf{\Lambda}^M\mathbf{U}^{-1}\mathbf{x},$$

or

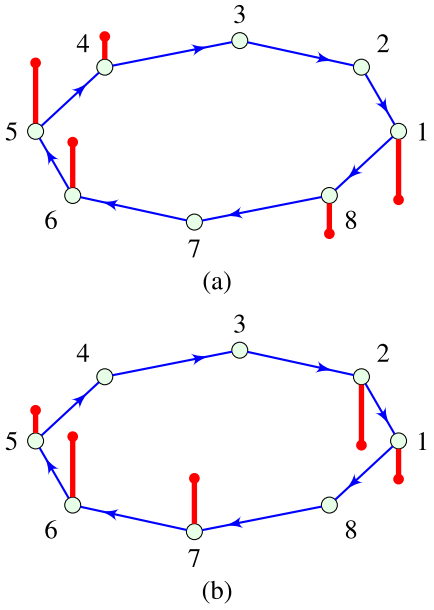

Fig. 14. Graph, graph signal, and a shift operator on a directed graph (classical circular shift). (a) Elements of a signal, $\mathbf{x}$, shown as red lines on a directed circular graph. (b) The shifted version, $\mathbf{Ax}$, of the graph signal from (a).

$$\mathbf{Y} = (h_0 + h_1\mathbf{\Lambda} + h_2\mathbf{\Lambda}^2 + \cdots + h_M\mathbf{\Lambda}^M)\mathbf{X} = H(\mathbf{\Lambda})\mathbf{X},$$

where

$$\mathbf{Y} = \mathbf{U}^{-1}\mathbf{y} \ \text{ and } \ \mathbf{X} = \mathbf{U}^{-1}\mathbf{x}$$

are the DFTs of the input and output signal and $H(\mathbf{\Lambda})$ is a diagonal transfer function. Indeed, this is the case since the eigenvalue relation for the adjacency matrix $\mathbf{A}$ is

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0,$$

and $(\mathbf{A} - \lambda\mathbf{I})$ is a super-diagonal matrix for which we get

$$\det(\mathbf{A} - \lambda\mathbf{I}) = \lambda^N - 1 = 0.$$

Since $1 = e^{-j2\pi(k-1)}$, the solutions are

$$\lambda_k = e^{-j2\pi(k-1)/N} \ \text{ with } \ u_k(n) = \tfrac{1}{\sqrt{N}} e^{j2\pi(k-1)(n-1)/N},$$

for $k = 1, 2, \dots, N$, producing exactly the DFT transformation matrix $\mathbf{U}^{-1} = \mathbf{U}^H$, with normalized columns (so that their energy is unity) and indexing $n = 1, 2, \dots, N$. We can easily see that

$$H(\mathbf{\Lambda}) = (h_0 + h_1\mathbf{\Lambda} + h_2\mathbf{\Lambda}^2 + \cdots + h_M\mathbf{\Lambda}^M)$$

is the DFT of the FIR system coefficients, producing the well-known element-wise relation

$$Y(k) = H(k)X(k)$$
$$H(k) = h_0 + h_1\lambda_k + h_2\lambda_k^2 + \cdots + h_M\lambda_k^M.$$

For an irregular graph domain of a general form, the *graph shift* is not so simple, and several graph operators (listed in Table 1 and denoted by using a unified operator $\mathbf{S}$ instead of $\mathbf{A}$) are used to produce the system on the graph

$$\mathbf{y} = h_0\mathbf{x} + h_1\mathbf{S}\mathbf{x} + h_2\mathbf{S}^2\mathbf{x} + \cdots + h_M\mathbf{S}^M\mathbf{x} = H(\mathbf{S})\mathbf{x},$$

with the corresponding GFT form

Table 4
Time-frequency to vertex-frequency analysis correspondences.

Time-frequency analysis	Vertex-frequency analysis
$S(m,k)=\frac{1}{\sqrt{N}}\sum_{n=1}^{N}x(n)h(n-m)\,e^{-j2\pi(k-1)(n-1)/N}$	$S(m,k)=\sum_{n=1}^{N}x(n)h_m(n)\,u_k^*(n)$
$h_m(n)=h(n-m)=\frac{1}{\sqrt{N}}\sum_{k=1}^{N}H(k)e^{-j\frac{2\pi}{N}(m-1)(k-1)}e^{j\frac{2\pi}{N}(n-1)(k-1)}$	$h_m(n)=\sum_{k=1}^{N}H(k)u_k^*(m)u_k(n)$
$\mathcal{H}_{m,k}(n)=\frac{1}{\sqrt{N}}h(n-m)e^{-j\frac{2\pi}{N}(n-1)(k-1)}$	$\mathcal{H}_{m,k}(n)=h_m(n)u_k^*(n)$
$S(m,k)=\langle\mathcal{H}_{m,k}(n),x(n)\rangle=\sum_{n=1}^{N}\mathcal{H}_{m,k}(n)x(n)$	$S(m,k)=\langle\mathcal{H}_{m,k}(n),x(n)\rangle=\sum_{n=1}^{N}\mathcal{H}_{m,k}(n)x(n)$
$S(m,k)=\frac{1}{\sqrt{N}}\sum_{p=1}^{N}X(p)H(p-k)e^{j\frac{2\pi}{N}(m-1)(p-1)}$	$S(m,k)=\sum_{p=1}^{N}X(p)H_k(p)\,u_p(m)$
Polynomial approximation and implementation of $H(p-k)$ is not widely used in classical time-frequency analysis	$\mathbf{s}_k=\mathbf{U}H_k(\mathbf{\Lambda})\mathbf{X}=\mathbf{U}H_k(\mathbf{\Lambda})\mathbf{U}^{-1}\mathbf{x}=H_k(\mathbf{L})\mathbf{x}\approx\bar{P}_{k,M-1}(\mathbf{L})\mathbf{x}$ $\bar{P}_{k,M-1}(\mathbf{L})=h_{0,k}\mathbf{I}+h_{1,k}\mathbf{L}+\cdots+h_{(M-1),k}\mathbf{L}^{M-1}$
$RD(n,k)=\frac{1}{\sqrt{N}}x(n)X^*(k)e^{-j\frac{2\pi}{N}(n-1)(k-1)}$	$E(n,k)=x(n)X^*(k)u_k^*(n)$
$\|\mathbf{x}\|_0\|\mathbf{X}\|_0\geq N$	$\|\mathbf{x}\|_0\|\mathbf{X}\|_0\geq\frac{1}{\max_{k,m}\{\|u_k(m)\|^2\}}$
Frequency $\omega_k=2\pi(k-1)/N$, frequency index, k of the basis function $\exp(j2\pi(k-1)(n-1))/\sqrt{N}$	Smoothness, λ_k, smoothness index, k of the Laplacian basis function $u_k(n)$
$\frac{\int_t\omega RD(t,\omega)d\omega}{\int_t RD(t,\omega)d\omega}=\phi'(t)=\omega_i(t)$	$\frac{\sum_{k=1}^{N}\lambda_k E(n,k)}{\sum_{k=1}^{N}E(n,k)}=\frac{\mathcal{L}_x(n)}{x(n)}=\lambda(n)$
$C(t,\omega)=\int_u\int_v X(u)X^*(v)e^{jut}e^{-jvt}\phi(u-v,\omega-u)\frac{dudv}{4\pi^2}$	$G(n,k)=\sum_{p=1}^{N}\sum_{q=1}^{N}X(p)X^*(q)u_p(n)u_q^*(n)\phi(p,k,q)$
Time marginal condition, $c(\theta,0)=1$, $\frac{1}{2\pi}\int_u\phi(u-v,\omega-u)d\omega=1$	Vertex marginal condition, $\sum_{k=1}^{N}\phi(p,k,q)=1$
Frequency marginal, $c(0,\tau)=1$, $\phi(u-v,\omega-u)=\delta(u-\omega)$	Frequency marginal condition, $\phi(p,k,p)=\delta(p-k)$

It has been assumed that $\mathbf{U}^{-1}=\mathbf{U}^H$. For symmetric matrices (like, the commonly used graph Laplacian) $\mathbf{U}^{-1}=\mathbf{U}^T$ and $u_k^*(n)=u_k(n)$.

$$\mathbf{Y}=(h_0+h_1\mathbf{\Lambda}+h_2\mathbf{\Lambda}^2+\cdots+h_M\mathbf{\Lambda}^M)\mathbf{X}=H(\mathbf{\Lambda})\mathbf{X},$$

$$H(\mathbf{\Lambda})=(h_0+h_1\mathbf{\Lambda}+h_2\mathbf{\Lambda}^2+\cdots+h_M\mathbf{\Lambda}^M)$$

where

$$\mathbf{Y}=\mathbf{U}^{-1}\mathbf{y}\quad\text{and}\quad\mathbf{X}=\mathbf{U}^{-1}\mathbf{x}$$

are the GFTs with the transformation matrix, $\mathbf{U}$, and eigenvalue matrix, $\mathbf{\Lambda}$, is obtained using the corresponding definitions in Table 1. The element-wise form of the GFT is given in (1).

It is important to note that in the same way as the calculation of $\mathbf{A}^M\mathbf{x}$ on directed circular graph requires the signal neighbors at a distance M from the considered instant, n, defined by $(n-M)$, the calculation of the term, $\mathbf{S}^M\mathbf{x}$, in the presented graph shift operators from Table 1, requires only M-neighborhood of the considered vertex, n. This fact is of great importance for large graphs and processing of big data on these graphs.

Undirected circular graph. If the circular graph in Fig. 14 were not directed, then every vertex n would be connected to both neighboring vertices, $n-1$ and $n+1$, to produce the adjacency matrix, $\mathbf{A}=\mathbf{W}$, and the graph Laplacian, $\mathbf{L}=\mathbf{D}-\mathbf{W}$, of the form

$$\mathbf{W}=\begin{bmatrix}0&1&0&0&0&0&0&1\\1&0&1&0&0&0&0&0\\0&1&0&1&0&0&0&0\\0&0&1&0&1&0&0&0\\0&0&0&1&0&1&0&0\\0&0&0&0&1&0&1&0\\0&0&0&0&0&1&0&1\\1&0&0&0&0&0&1&0\end{bmatrix},$$

$$\mathbf{L}=\begin{bmatrix}2&-1&0&0&0&0&0&-1\\-1&2&-1&0&0&0&0&0\\0&-1&2&-1&0&0&0&0\\0&0&-1&2&-1&0&0&0\\0&0&0&-1&2&-1&0&0\\0&0&0&0&-1&2&-1&0\\0&0&0&0&0&-1&2&-1\\-1&0&0&0&0&0&-1&2\end{bmatrix}.$$

The eigendecomposition relation for the graph Laplacian of this graph, $\mathbf{L}\mathbf{u}_k=\lambda_k\mathbf{u}_k$, admits a simple element-wise form

$$-u_k(n-1)+2u_k(n)-u_k(n+1)=\lambda_k u_k(n). \tag{76}$$

The solution to the second-order difference equation in (76) is

$$u_k(n)=\cos\left(\frac{2\pi(k-1)(n-1)}{N}+\phi_k\right)$$

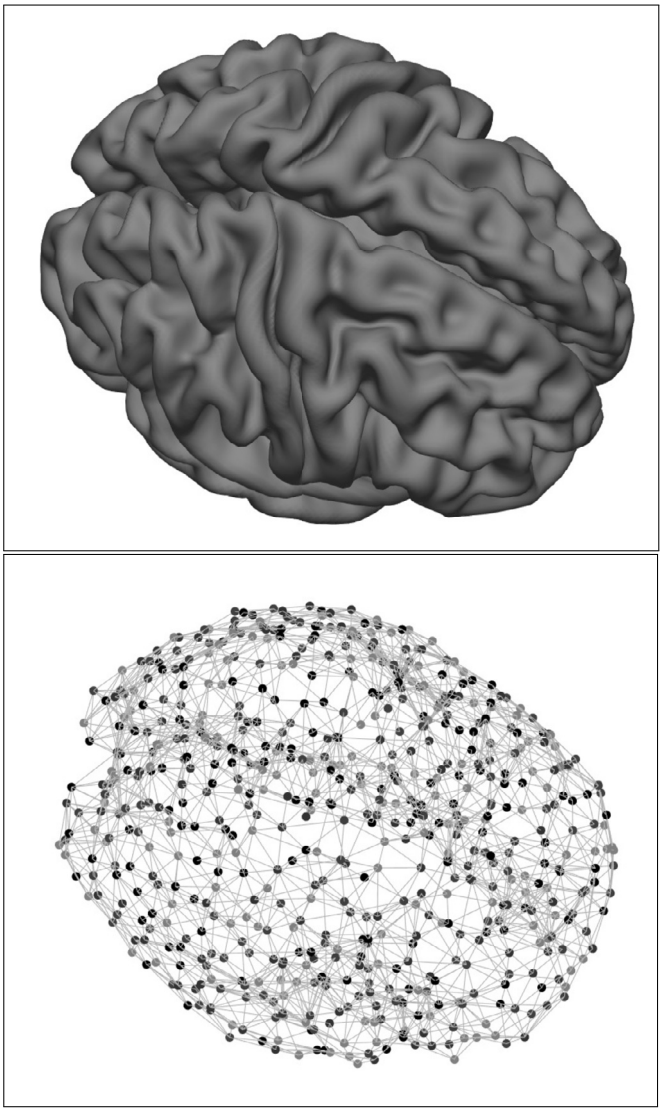

Fig. 15. Brain atlas (top) and its graph, with vertex coloring based on three smoothest generalized eigenvectors, $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$, of graph Laplacian, with the spectral vector, $\mathbf{q}_n = [u_1(n), u_2(n), u_3(n)]$, employed as coordinates for the RGB coloring scheme (bottom), [57,58].

with

$$\lambda_k = 2\Big(1 - \cos\Big(\frac{2\pi(k-1)}{N}\Big)\Big)$$
$$= 4\sin^2\Big(\frac{1}{2}\frac{2\pi(k-1)}{N}\Big) = 4\sin^2\Big(\frac{1}{2}\omega_k\Big).$$

Obviously, for every eigenvalue, λ_k (except for λ_1 and for the last eigenvalue, λ_N, for an even N), there exist two orthogonal eigenvectors with, for example, $\phi_k = 0$ and $\phi_k = \pi/2$. The individual eigenvectors, $\mathbf{u}_k$, correspond to the standard harmonic basis functions, $\cos(\frac{2\pi(k-1)(n-1)}{N})$ and $\sin(\frac{2\pi(k-1)(n-1)}{N})$, and the standard Fourier series analysis of real-valued signals follows. Notice that for small frequency, ω_k^2, we can write

$$\lambda_k = 4\sin^2(\frac{1}{2}\omega_k) \approx \omega_k^2$$

meaning that the smoothness index, λ_k, of an graph Laplacian eigenvector, $\mathbf{u}_k$, is related to the squared frequency, ω_k^2, of the corresponding harmonic in classical analysis (see also Table 4).

Appendix B. Brain connectivity graph

The human brain activity signals can be mapped on a graph where each vertex corresponds to a brain region. The edge weights are considered to be known *a priori* and represent the structural connectivity or the functional coherence between brain regions. The graph signal processing and vertex-frequency analysis can be used to analyze the brain activity signal. It is known that, for example, slow-varying spectral content in the graph brain signal represents similar activities in regions that are highly connected in the functional brain graphs, while high frequencies denote very different activities in such brain regions [2].

Fig. 15 shows the benchmark Brain Atlas connectivity graph [57,58], where the data is given in two matrices: "Coactivation matrix", $\hat{\mathbf{W}}$, and "Coordinate matrix". The "Coordinate matrix" contains the vertex coordinates in a three-dimensional Euclidean space, whereby the coordinate of a vertex n is defined by the n-th row of the "Coordinate matrix", that is, $[x_n, y_n, z_n]$.

In our analysis, the graph weight matrix, $\mathbf{W}$, was formed by:

(i) Thresholding the "Coactivation matrix", $\hat{\mathbf{W}}$, to preserve only the strongest connections within this brain atlas, for example, those greater than $0.1\max\{\hat{W}_{mn}\}$, as recommended in [58];

(ii) Only the edges between the vertices m and n, whose Euclidean distance satisfies $d_{mn} \leq 20$ are kept in the graph representation.

The elements, W_{mn}, of the brain graph weight matrix, $\mathbf{W}$, are therefore obtained from the corresponding elements, $\hat{W}_{mn}$, of the "Coactivation matrix" as

$$W_{mn} = \begin{cases} \hat{W}_{mn}, & \text{if } \hat{W}_{mn} > 0.1\max\{\hat{W}_{mn}\} \text{ and } d_{mn} \leq 20 \\ 0, & \text{elsewhere.} \end{cases} \tag{77}$$

The brain connectivity graph with the so defined weight matrix, $\mathbf{W}$, is shown in Fig. 15 (bottom).

Vertex-frequency kernels for the brain graph analysis are presented for two vertices, $n = 300$ and $n = 100$. Two frequency indices are also considered, $k = 1$ and $k = 31$. Kernels at the vertex $n = 300$ and frequency index $k = 1$ and $k = 31$ are presented with wider and narrower vertex domain support function, as shown in Fig. 16. These kernels enable localized vertex-frequency analysis of brain signals, using different resolutions.

Appendix C. Sensor network graphs

One of the most natural and straightforward applications of graph signal processing is in the context of sensor networks, for example a temperature sensor network, described in [1]. A graph in these examples represents relative positions of sensors in the environment, and the application areas include denoising, reconstruction, or distributed processing of sensor data.

To this end, consider a multi-sensor setup, shown in Fig. 17, for measuring temperature field in a known geographical region. The temperature sensing locations are chosen according to the significance of a particular geographic area to local users. The weight W_{nm} indicates the strength of the coupling between signal values at the sensing points n and m; it has the value $W_{nm} = 0$ if the points n and m are not related or if $n = m$. Each edge has an associated weight, W_{nm}, which adds a "mutual sensor relevance" information to the already established "spatial awareness" modeled by the edges. This equips graph signal models with additional flexibility.

Spectral analysis of the graph is performed using the graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$. The kernels for the vertex-frequency analysis presented in this paper may be used for denoising or compression. The kernels for the vertex $n = 39$ and the frequency indices $k = 1$ and $k = 7$ are shown in Fig. 17, for two different kernel widths.

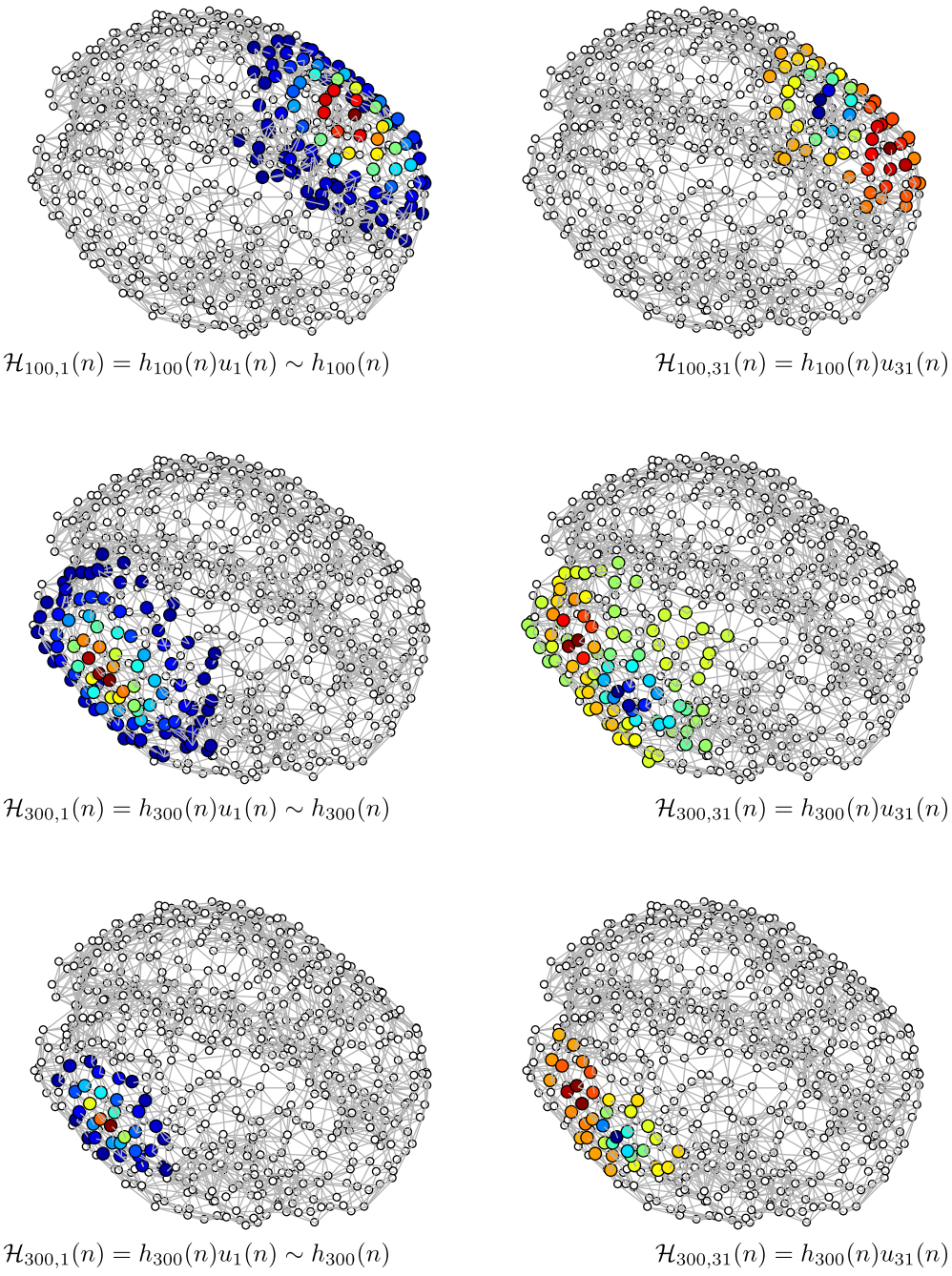

Fig. 16. Vertex-frequency kernels for the brain graph analysis. (Top) Kernels at the vertex $n = 300$ and frequency index $k = 1$ and $k = 31$ and its graph. (Middle) Kernels at the vertex $n = 100$ and frequency index $k = 1$ and $k = 31$ and its graph. (Bottom) Kernels at the vertex $n = 300$ and frequency index $k = 1$ and $k = 31$ and its graph with a narrower vertex domain support function.

The sensing network for the benchmark *Minnesota road graph* is shown in Fig. 18 for the kernels obtained by shifting the spectral domain band-pass functions of the raised cosine form. Two widths of these functions are considered: (i) one when the whole spectral domain from λ_0 to $\lambda_{\max}$ is covered by $K = 21$ overlapping band-pass functions, as in Fig. 4(b), and (ii) the other when there are $K = 121$ functions within the whole spectral range. In general, wider spectral domain functions ($K = 21$) correspond to higher vertex domain resolution, while the narrower band-pass functions ($K = 121$) produce higher spectral domain resolution. The vertex $m = 800$ is used as the central position for all considered vertex-frequency kernels. In this example, the generalized graph Laplacian eigenvectors, $\mathbf{L}\mathbf{u}_k = \lambda_k \mathbf{D}\mathbf{u}_k$, are used as the basis functions, $\mathbf{u}_k$, in the kernel calculation. These functions provide basis for localized graph signal analysis and processing, with the conditions for signal reconstruction over the entire graph.

Appendix D. Graph spectrogram and frames

The *graph spectrogram* is defined, based on (9), as

$$|S(m,k)|^2 = \left| \sum_{n=1}^{N} x(n) h_m(n)\, u_k(n) \right|^2. \tag{78}$$

Then, according to Parseval's theorem, the *vertex marginal property*, which is a projection of $|S(m,k)|^2$ onto the vertex index axis, is given by

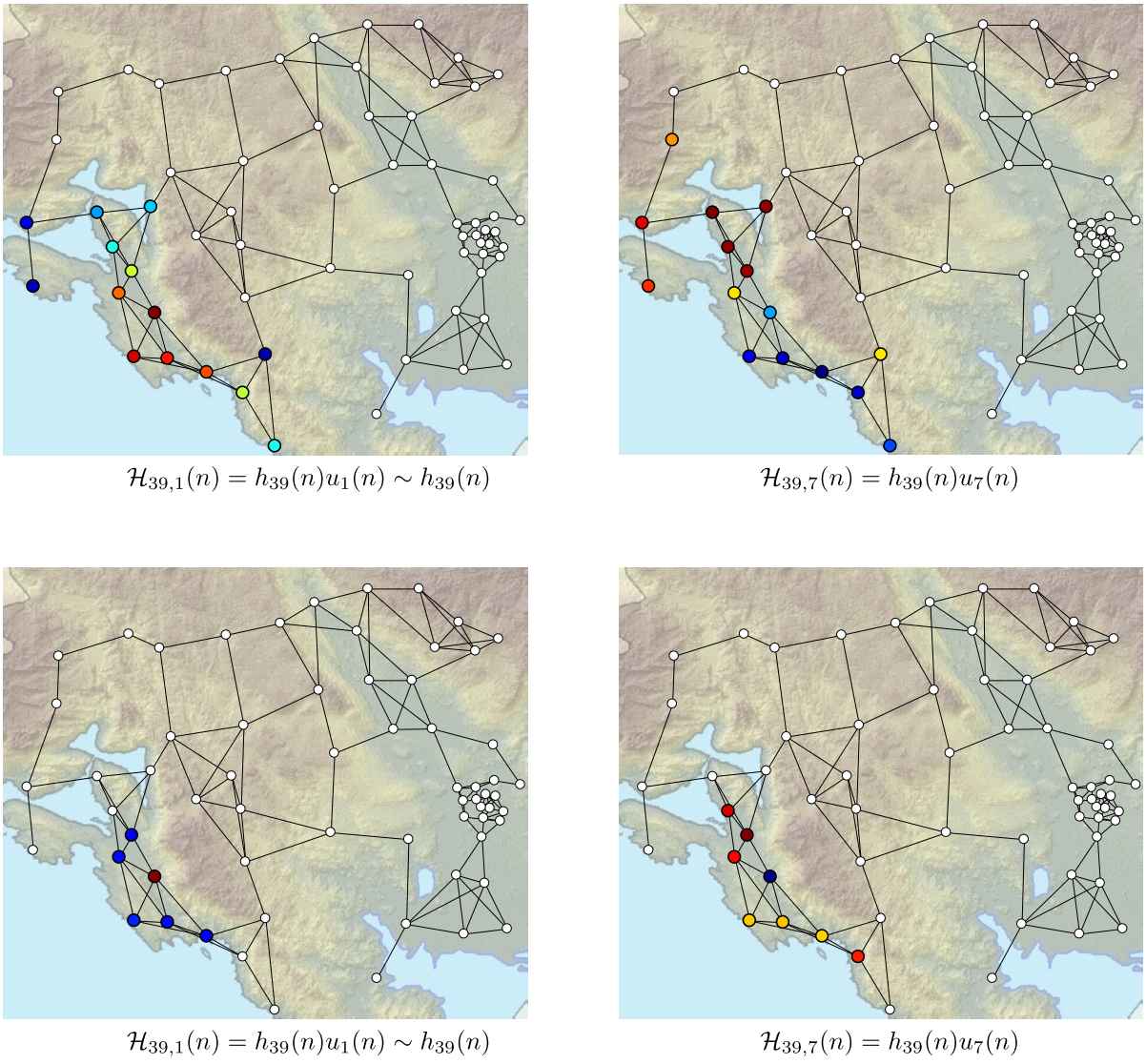

Fig. 17. Vertex-frequency kernels for the sensing network analysis. (Top) Kernels at the vertex $n = 300$ and frequency index $k = 1$ and $k = 31$ and its graph. (Bottom) Kernels at the vertex $n = 300$ and frequency index $k = 1$ and $k = 31$ and its graph with a narrower vertex domain support function.

$$\sum_{k=1}^{N} |S(m,k)|^2 = \sum_{k=1}^{N} S(m,k) \sum_{n=1}^{N} x(n)h_m(n)\, u_k(n)$$

$$= \sum_{n=1}^{N} |x(n)h_m(n)|^2,$$

which would be equal to the signal power, $|x(m)|^2$, at the vertex m, if $h_m(n) = \delta(m-n)$. Since this is not the case, the vertex marginal property of the graph spectrogram is equal to the power of the graph signal in hand, smoothed by the window, $h_m(n)$.

Energy of graph spectrogram. For the total energy of vertex spectrogram, we consequently have

$$\sum_{m=1}^{N}\sum_{k=1}^{N} |S(m,k)|^2 = \sum_{n=1}^{N} \left(|x(n)|^2 \sum_{m=1}^{N} |h_m(n)|^2 \right). \tag{79}$$

If $\sum_{m=1}^{N} |h_m(n)|^2 = 1$ for all n, then the spectrogram on the graph is *energy unbiased* (statistically consistent with respect to the energy), that is

$$\sum_{m=1}^{N}\sum_{k=1}^{N} |S(m,k)|^2 = \sum_{n=1}^{N} |x(n)|^2 = ||\mathbf{x}||^2 = E_x. \tag{80}$$

The LGFT viewed as a frame. A set of functions, $S(m,k)$, is called *a frame* for the expansion of a graph signal, $\mathbf{x}$, if [30]

$$A||\mathbf{x}||^2 \leq \sum_{m=1}^{N} |S(m,k)|^2 \leq B||\mathbf{x}||^2,$$

where A and B are positive constants. The constants A and B govern the numerical stability of recovering the original signal, $\mathbf{x}$, from the coefficients, $S(m,k)$.

If $A = B$, the frame is termed *Parseval's tight frame* and the signal can be recovered as

$$x(n) = \frac{1}{A} \sum_{m=1}^{N}\sum_{k=1}^{N} S(m,k)h_m(n)u_k(n).$$

The LGFT defined in (15) is Parseval's tight frame if

$$\sum_{k=0}^{K-1}\sum_{m=1}^{N} |S(m,k)|^2 = \sum_{k=0}^{K-1}\sum_{p=1}^{N} |X(p)H_k(\lambda_p)|^2 = E_x, \tag{81}$$

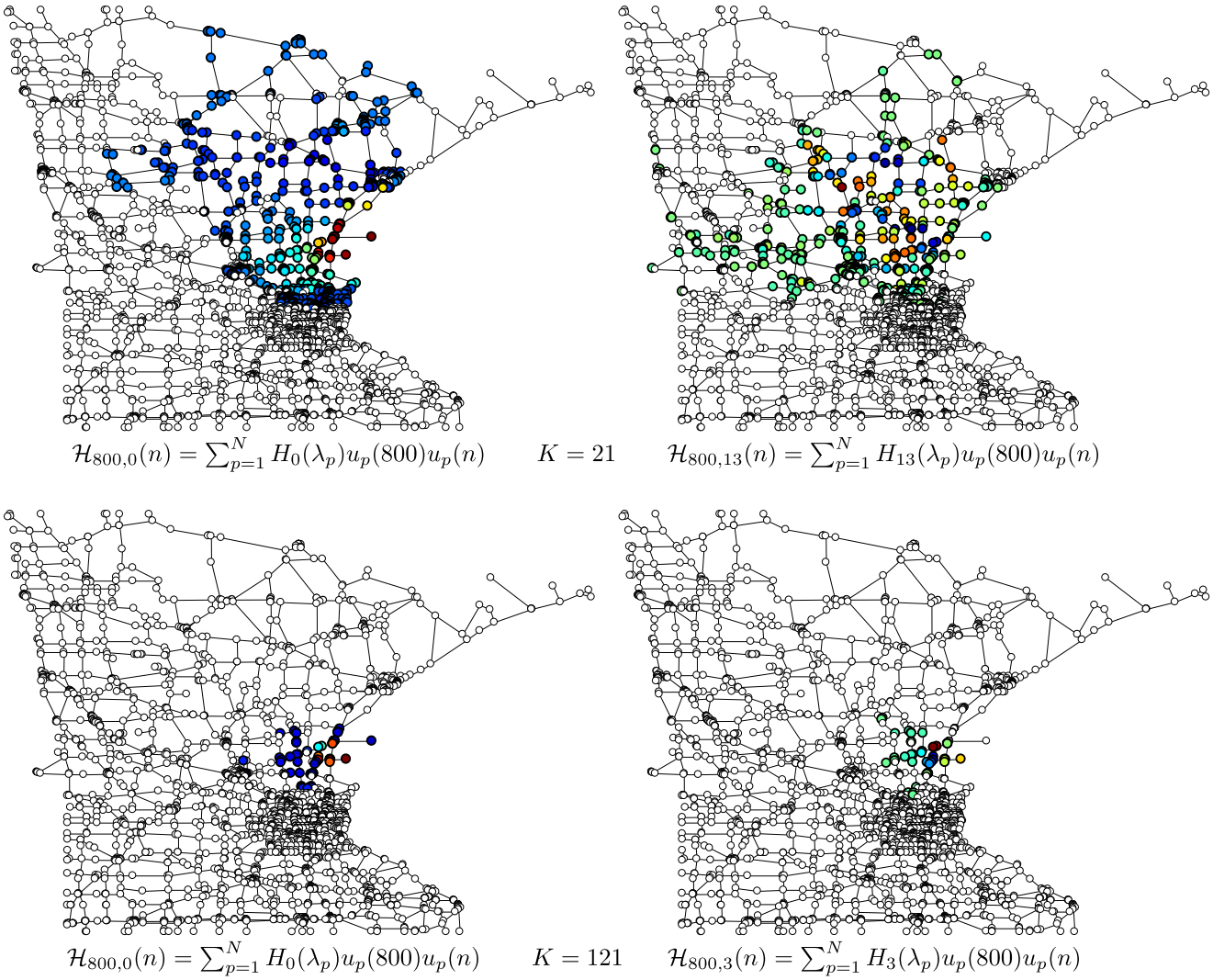

Fig. 18. Vertex-frequency kernels for the Minnesota road map graph as the sensing network domain. (Top) Kernels at the vertex $n = 800$ and the spectral band positions at $k = 0$ and $k = 13$, with the whole spectral domain covered by $K = 21$ overlapping band-pass functions of the raised cosine type. (Bottom) Kernels at the vertex $n = 800$ and the spectral band positions at $k = 0$ and $k = 3$, with the whole spectral domain covered by $K = 121$ overlapping band-pass functions of the raised cosine type.

where Parseval's theorem for the $S(m,k)$ as the GFT of $X(p)H_k(\lambda_p)$ is used to yield

$$\sum_{m=1}^{N} |S(m,k)|^2 = \sum_{p=1}^{N} |X(p)H_k(\lambda_p)|^2.$$

This means that the LGFT in (15) is a tight frame if the condition in (53) holds. This condition is used to define transfer functions in Fig. 4(b) and 4(c).

In general, form (81), it is easy to conclude that the graph spectrogram energy is bounded with

$$AE_x \leq \sum_{k=0}^{K-1} \sum_{m=1}^{N} |S(m,k)|^2 \leq BE_x, \tag{82}$$

where A and B are respectively the minimum and the maximum value of

$$g(\lambda_p) = \sum_{k=0}^{K-1} |H_k(\lambda_p)|^2.$$

In the same way as in the LGFT case, it can be shown that the wavelet transform also represents a frame with [22,23]

$$A||\mathbf{x}||^2 \leq \sum_{n=1}^{N} \sum_{i=0}^{K-1} |W(n,s_i)|^2 \leq B||\mathbf{x}||^2, \tag{83}$$

where $i = 0$ stands for the coefficient obtained for the scale function and

$$A = \min_{\lambda_p} g(\lambda_p),$$

$$B = \max_{\lambda_p} g(\lambda_p), \quad \text{and}$$

$$g(\lambda_p) = G^2(\lambda_p) + \sum_{i=1}^{K-1} |H(s_i\lambda_p)|^2. \tag{84}$$

The low-pass scale function, $G(\lambda_p)$, is added in the reconstruction formula, $\psi(n,s_0) \to \psi(n,s_0) + \phi(n,s_0)$, since all $H(s_i\lambda) = 0$ for $\lambda = 0$. It should be mentioned that the spectral functions of the wavelet transform, $H(s_i\lambda_p)$, form Parseval's frame if $g(\lambda_p) = 1$, that is relation (57) holds, when $A = B = 1$.

If a continuous (polynomial) approximation of the transfer functions is used, $G^2(\lambda) \approx \bar{P}_{0,M-1}(\lambda)$ and $H(s_i\lambda) \approx \bar{P}_{i,M-1}(\lambda)$, $i = 1, 2, \ldots, K-1$, then an approximation of the constants A and B is obtained by finding the respective minimum and maximum

of the continuous approximation $g(\lambda)$ in (84), within the interval $0 \leq \lambda \leq \lambda_{max}$.

References

[1] L. Stankovic, D.P. Mandic, M. Dakovic, I. Kisil, E. Sejdic, A.G. Constantinides, Understanding the basis of graph signal processing via an intuitive example-driven approach [lecture notes], IEEE Signal Process. Mag. 36 (6) (2019) 133–145.

[2] A. Ortega, P. Frossard, J. Kovačević, J.M. Moura, P. Vandergheynst, Graph signal processing: overview, challenges, and applications, Proc. IEEE 106 (5) (2018) 808–828.

[3] A. Sandryhaila, J.M. Moura, Discrete signal processing on graphs, IEEE Trans. Signal Process. 61 (7) (2013) 1644–1656.

[4] A. Marques, A. Ribeiro, S. Segarra, Graph signal processing: fundamentals and applications to diffusion processes, in: Proc. Int. Conf. Accoustic, Speech and Signal Processing, (ICASSP), 2017, IEEE, 2017.

[5] L. Stankovic, D. Mandic, M. Dakovic, M. Brajovic, B. Scalzo, T. Constantinides, Graph signal processing–parts I, II, III, arXiv preprint, arXiv:1907.03467, 2019.

[6] A. Sandryhaila, J.M. Moura, Discrete signal processing on graphs: frequency analysis, IEEE Trans. Signal Process. 62 (12) (2014) 3042–3054.

[7] B. Ricaud, P. Borgnat, N. Tremblay, P. Gonçalves, P. Vandergheynst, Fourier could be a data scientist: from graph Fourier transform to signal processing on graphs, C. R. Phys. 20 (5) (2019) 474–488.

[8] S. Chen, R. Varma, A. Sandryhaila, J. Kovačević, Discrete signal processing on graphs: sampling theory, IEEE Trans. Signal Process. 63 (24) (2015) 6510–6523.

[9] R. Hamon, P. Borgnat, P. Flandrin, C. Robardet, Extraction of temporal network structures from graph-based signals, IEEE Trans. Signal Inf. Process. Netw. 2 (2) (2016) 215–226.

[10] R. Hamon, P. Borgnat, P. Flandrin, C. Robardet, Transformation from graphs to signals and back, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 111–139.

[11] A. Sandryhaila, 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. 31 (5) (2014) 80–90.

[12] L. Stanković, M. Daković, T. Thayaparan, Time-Frequency Signal Analysis with Applications, Artech House, 2014.

[13] L. Cohen, Time-Frequency Analysis, Prentice Hall PTR, 1995.

[14] B. Boashash, Time-Frequency Signal Analysis and Processing: A Comprehensive Reference, Academic Press, 2015.

[15] D.I. Shuman, B. Ricaud, P. Vandergheynst, Vertex-frequency analysis on graphs, Appl. Comput. Harmon. Anal. 40 (2) (2016) 260–291.

[16] D.I. Shuman, B. Ricaud, P. Vandergheynst, A windowed graph Fourier transform, in: Proc. IEEE Statistical Signal Processing Workshop (SSP), 2012, pp. 133–136.

[17] X.-W. Zheng, Y.Y. Tang, J.-T. Zhou, H.-L. Yuan, Y.-L. Wang, L.-N. Yang, J.-J. Pan, Multi-windowed graph Fourier frames, in: Proc. IEEE International Conference on Machine Learning and Cybernetics, vol. 2, 2016, pp. 1042–1048.

[18] M. Tepper, G. Sapiro, A short-graph Fourier transform via personalized pagerank vectors, in: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016, pp. 4806–4810.

[19] L. Stanković, M. Daković, E. Sejdić, Vertex-frequency analysis: a way to localize graph spectral components [lecture notes], IEEE Signal Process. Mag. 34 (4) (2017) 176–182.

[20] T. Cioacă, B. Dumitrescu, M.-S. Stupariu, Graph-based wavelet multiresolution modeling of multivariate terrain data, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 479–507.

[21] D. Hammond, P. Vandergheynst, R. Gribonval, Wavelets on graphs via spectral graph theory, Appl. Comput. Harmon. Anal. 30 (2) (2011) 129–150.

[22] D.K. Hammond, P. Vandergheynst, R. Gribonval, The spectral graph wavelet transform: fundamental theory and fast computation, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 141–175.

[23] H. Behjat, D. Van De Ville, Spectral design of signal-adapted tight frames on graphs, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 177–206.

[24] D.I. Shuman, S.K. Narang, P. Frossard, A. Ortega, 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. 30 (3) (2013) 83–98.

[25] S. Sardellitti, S. Barbarossa, P. Di Lorenzo, On the graph Fourier transform for directed graphs, IEEE J. Sel. Top. Signal Process. 11 (6) (2017) 796–811.

[26] C.J. Quinn, N. Kiyavash, T.P. Coleman, Directed information graphs, IEEE Trans. Inf. Theory 61 (12) (2015) 6887–6909.

[27] L. Stanković, On the STFT inversion redundancy, IEEE Trans. Circuits Syst. II, Express Briefs 63 (3) (2015) 284–288.

[28] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, arXiv preprint, arXiv:1609.02907, 2016.

[29] N. Leonardi, D. Van De Ville, Tight wavelet frames on multislice graphs, IEEE Trans. Signal Process. 61 (13) (2013) 3357–3367.

[30] H. Behjat, U. Richter, D. Van De Ville, L. Sörnmo, Signal-adapted tight frames on graphs, IEEE Trans. Signal Process. 64 (22) (2016) 6017–6029.

[31] M. Brajović, L. Stanković, M. Daković, On polynomial approximations of spectral windows in vertex-frequency representations, in: 2020 24th International Conference on Information Technology (IT), IEEE, 2020, pp. 1–4.

[32] F. Murtagh, The Haar wavelet transform of a dendrogram, J. Classif. 24 (1) (2007) 3–32.

[33] B.N. Ann B. Lee, L. Wasserman, Treelets: an adaptive multiscale basis for sparse unordered data, Ann. Appl. Stat. 2 (2) (2008) 435–471.

[34] M. Jansen, G.P. Nason, B.W. Silverman, Multiscale methods for data on graphs and irregular multidimensional situations, J. R. Stat. Soc., Ser. B, Stat. Methodol. 71 (1) (2009) 97–125.

[35] S.K. Narang, A. Ortega, Lifting based wavelet transforms on graphs, in: Proceedings: APSIPA ASC 2009: Asia-Pacific Signal and Information Processing Association, 2009 Annual Summit and Conference, 2009, pp. 441–444.

[36] R. Rustamov, L.J. Guibas, Wavelets on graphs via deep learning, in: Advances in Neural Information Processing Systems, 2013, pp. 998–1006.

[37] R.R. Coifman, S. Lafon, Diffusion maps, Appl. Comput. Harmon. Anal. 21 (1) (2006) 5–30.

[38] M. Maggioni, H. Mhaskar, Diffusion polynomial frames on metric measure spaces, Appl. Comput. Harmon. Anal. 24 (3) (2008) 329–353.

[39] S.K. Narang, A. Ortega, Perfect reconstruction two-channel wavelet filter banks for graph structured data, IEEE Trans. Signal Process. 60 (6) (2012) 2786–2799.

[40] H. Behjat, N. Leonardi, L. Sörnmo, D. Van De Ville, Anatomically-adapted graph wavelets for improved group-level fMRI activation mapping, NeuroImage 123 (2015) 185–199.

[41] I. Jestrović, J.L. Coyle, E. Sejdić, A fast algorithm for vertex-frequency representations of signals on graphs, Signal Process. 131 (2017) 483–491.

[42] M. Masoumi, M. Rezaei, A.B. Hamza, Shape analysis of carpal bones using spectral graph wavelets, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 419–436.

[43] L. Stanković, A measure of some time–frequency distributions concentration, Signal Process. 81 (3) (2001) 621–631.

[44] B. Pasdeloup, V. Gripon, R. Alami, M.G. Rabbat, Uncertainty principle on graphs, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 317–340.

[45] M. Tsitsvero, S. Barbarossa, P. Di Lorenzo, Signals on graphs: uncertainty principle and sampling, IEEE Trans. Signal Process. 64 (18) (2016) 539–554.

[46] A. Agaskar, Y.M. Lu, A spectral graph uncertainty principle, IEEE Trans. Inf. Theory 59 (7) (2013) 4338–4356.

[47] B. Ricaud, B. Torrésani, A survey of uncertainty principles and some signal processing applications, Adv. Comput. Math. 40 (3) (2014) 629–650.

[48] N. Perraudin, B. Ricaud, D.I. Shuman, P. Vandergheynst, Global and local uncertainty principles for signals on graphs, APSIPA Trans. Signal Inf. Process. 7 (e3) (2018) 1–26.

[49] L. Stankovic, Highly concentrated time-frequency distributions: pseudo quantum signal representation, IEEE Trans. Signal Process. 45 (3) (1997) 543–551.

[50] M. Elad, A.M. Bruckstein, Generalized uncertainty principle and sparse representation in pairs of bases, IEEE Trans. Inf. Theory 48 (9) (2002) 2558–2567.

[51] L. Stankovic, The support uncertainty principle and the graph Rihaczek distribution: revisited and improved, IEEE Signal Process. Lett. (2020), https://doi.org/10.1109/LSP.2020.3000686, early access.

[52] A.W. Bohannon, B.M. Sadler, R.V. Balan, A filtering framework for time-varying graph signals, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 341–376.

[53] L. Stanković, E. Sejdić, M. Daković, Vertex-frequency energy distributions, IEEE Signal Process. Lett. 25 (3) (2018) 358–362.

[54] L. Stanković, M. Daković, E. Sejdić, Vertex-frequency energy distributions, in: Vertex-Frequency Analysis of Graph Signals, Springer, 2019, pp. 377–415.

[55] M. Daković, L. Stanković, E. Sejdić, Local smoothness of graph signals, Math. Probl. Eng. 2019 (2019).

[56] L. Stanković, E. Sejdić, M. Daković, Reduced interference vertex-frequency distributions, IEEE Signal Process. Lett. 25 (9) (2018) 1393–1397.

[57] M. Mijalkov, E. Kakaei, J.B. Pereira, E. Westman, G. Volpe, BRAPH: a graph theory software for the analysis of brain connectivity, PLoS ONE 12 (8) (2017) e0178798.

[58] Complex Network Measures of Brain Connectivity: Uses and Interpretations, NeuroImage 52 (3) (2010) 1059–1069.

Ljubiša Stanković, (ljubisa@ucg.ac.me) is professor at the University of Montenegro. His research interests include time-frequency analysis, compressive sensing, and graph signal processing. He is a vice-president of the National Academy of Sciences and Arts of Montenegro (CANU) and is a recipient of the 2017 European Association for Signal Processing Best Journal Paper Award. He is a Fellow of the IEEE.

Danilo P. Mandic, (d.mandic@imperial.ac.uk) is a professor of signal processing at Imperial College London, United Kingdom. He has a keen interest in signal processing education, is a member of the IEEE Signal Processing Society Education Technical Committee, and his contributions were

recognized with the President's Award for Excellence in Post-graduate Supervision at Imperial College in 2014. He is a Fellow of the IEEE.

Miloš Daković, (milos@ucg.ac.me) is professor at the University of Montenegro. His research interests include graph signal processing, compressive sensing, and time–frequency analysis. He is a Member of the IEEE.

Bruno Scalzo Dees (bruno.scalzo-dees12@imperial.ac.uk) received the M.Eng. degree in aeronautical engineering from Imperial College London, where he is currently pursuing the Ph.D. His current research interests include graph signal processing, dynamical systems, and tensor decompositions.

Miloš Brajović, (milosb@ucg.ac.me) received the Ph.D. in EE from the University of Montenegro. He is currently working as a postdoctoral teaching assistant and his research interests include graph signal processing, time–frequency signal analysis, and compressive sensing.

Ervin Sejdić, (esejdic@ieee.org) is an associate professor at the University of Pittsburgh, Pennsylvania. His research interests include biomedical signal processing, rehabilitation engineering, and neuroscience. He received the U.S. Presidential Early Career Award for Scientists and Engineers in 2016. He is a Senior Member of the IEEE.

Anthony G. Constantinides, (a.constantinides@imperial.ac.uk) is emeritus professor of signal processing in the Department of Electrical and Electronic Engineering at Imperial College London, United Kingdom. He is a Life Fellow of the IEEE.