

Vertex-Frequency Analysis on Graphs

David I Shuman, Benjamin Ricaud, Pierre Vandergheynst^{1,2}

Signal Processing Laboratory (LTS2), Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Abstract

One of the key challenges in the area of signal processing on graphs is to design dictionaries and transform methods to identify and exploit structure in signals on weighted graphs. To do so, we need to account for the intrinsic geometric structure of the underlying graph data domain. In this paper, we generalize one of the most important signal processing tools - windowed Fourier analysis - to the graph setting. Our approach is to first define generalized convolution, translation, and modulation operators for signals on graphs, and explore related properties such as the localization of translated and modulated graph kernels. We then use these operators to define a windowed graph Fourier transform, enabling vertex-frequency analysis. When we apply this transform to a signal with frequency components that vary along a path graph, the resulting spectrogram matches our intuition from classical discrete-time signal processing. Yet, our construction is fully generalized and can be applied to analyze signals on any undirected, connected, weighted graph.

Keywords: Signal processing on graphs; time-frequency analysis; generalized translation and modulation; spectral graph theory; localization; clustering

1. Introduction

In applications such as social networks, electricity networks, transportation networks, and sensor networks, data naturally reside on the vertices of weighted graphs. Moreover, weighted graphs are a flexible tool that can be used to describe similarities between data points in statistical learning problems, functional connectivities between different regions of the brain, and the geometric structures of countless other topologically-complex data domains.

In order to reveal relevant structural properties of such data on graphs and/or sparsely represent different classes of signals on graphs, we can construct dictionaries of atoms, and represent graph signals as linear combinations of the dictionary atoms. The design of such dictionaries is one of the fundamental problems of signal processing, and the literature is filled with a wide range of dictionaries, including, e.g., Fourier, time-frequency, curvelet, shearlet, and bandlet dictionaries (see, e.g., [1] for an excellent historical overview of dictionary design methods and signal transforms).

Of course, the dictionary needs to be tailored to a given class of signals under consideration. Specifically, as exemplified in [2, Example 1], in order to identify and exploit structure in signals on weighted graphs, we need to account for the intrinsic geometric structure of the underlying data domain when designing dictionaries and signal transforms. When we construct dictionaries of features on weighted graphs, it is also desirable to (i) explicitly control how these features change from vertex to vertex, and (ii) ensure that we treat vertices in a homogeneous way (i.e., the resulting dictionaries are invariant to permutations in the vertex labeling). Unfortunately, weighted graphs are irregular structures that lack a shift-invariant notion

Email addresses: david.shuman@epfl.ch (David I Shuman), benjamin.ricaud@epfl.ch (Benjamin Ricaud), pierre.vandergheynst@epfl.ch (Pierre Vandergheynst)

¹This work was supported by FET-Open grant number 255931 UNLocX.

²Part of the work reported here was presented at the *IEEE Statistical Signal Processing Workshop, August 2011, Ann Arbor, MI*.

of translation, a key component in many signal processing techniques for data on regular Euclidean spaces. Thus, many of the existing dictionary design techniques cannot be directly applied to signals on graphs in a meaningful manner, and an important challenge is to design new localized transform methods that account for the structure of the data domain.

Accordingly, a number of new multiscale wavelet transforms for signals on graphs have been introduced recently (see [2] and references therein for a review of wavelet transforms for signals on graphs). Although the field of signal processing on graphs is still young, the hope is that such transforms can be used to efficiently extract information from high-dimensional data on graphs (either statistically or visually), as well as to regularize ill-posed inverse problems.

Windowed Fourier transforms, also called short-time Fourier transforms, are another important class of time-frequency analysis tools in classical signal processing. They are particularly useful in extracting information from signals with oscillations that are localized in time or space. Such signals appear frequently in applications such as audio and speech processing, vibration analysis, and radar detection. Our aim here is to generalize windowed Fourier analysis to the graph setting.

Underlying the classical windowed Fourier transform are the translation and modulation operators. While these fundamental operations seem simple in the classical setting, they become significantly more challenging when we deal with signals on graphs. For example, when we want to translate the blue Mexican hat wavelet on the real line in Figure 1(a) to the right by 5, the result is the dashed red signal. However, it is not immediately clear what it means to translate the blue signal in Figure 1(c) on the weighted graph in Figure 1(b) “to vertex 1000.” Modulating a signal on the real line by a complex exponential corresponds to translation in the Fourier domain. However, the analogous spectrum in the graph setting is discrete and bounded, and therefore it is difficult to define a modulation in the vertex domain that corresponds to translation in the graph spectral domain.

In this paper, an extended version of the short workshop proceeding [3], we define generalized convolution, translation, and modulation operators for signals on graphs, analyze properties of these operators, and then use them to adapt the classical windowed Fourier transform to the graph setting. The result is a method to construct windowed Fourier frames, dictionaries of atoms adapted to the underlying graph structure that enable vertex-frequency analysis, a generalization of time-frequency analysis to the graph setting. After a brief review of the classical windowed Fourier transform in the next section and some spectral graph theory background in Section 3, we introduce and study generalized convolution and translation operators in Section 4 and generalized modulation operators in Section 5. We then define and explore the properties of windowed graph Fourier frames in Section 6, where we also present illustrative examples of a graph spectrogram analysis tool and signal-adapted graph clustering. We conclude in Section 7 with some comments on open issues.

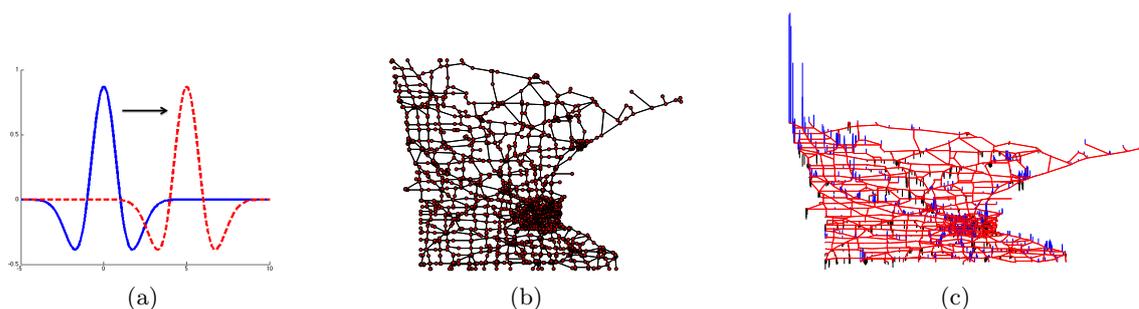


Figure 1: (a) Classical translation. (b) The Minnesota road graph [4], whose edge weights are all equal to 1. (c) What does it mean to “translate” this signal on the vertices of the Minnesota road graph? The blue and black lines represent the magnitudes of the positive and negative components of the signal, respectively.

2. The Classical Windowed Fourier Transform

For any $f \in L^2(\mathbb{R})$ and $u \in \mathbb{R}$, the translation operator $T_u : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R})$ is defined by

$$(T_u f)(t) := f(t - u), \quad (1)$$

and for any $\xi \in \mathbb{R}$, the modulation operator $M_\xi : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R})$ is defined by

$$(M_\xi f)(t) := e^{2\pi i \xi t} f(t). \quad (2)$$

Now let $g \in L^2(\mathbb{R})$ be a window (i.e., a smooth, localized function) with $\|g\|_2 = 1$. Then a windowed Fourier atom (see, e.g., [5], [6], [7, Chapter 4.2]) is given by

$$g_{u,\xi}(t) := (M_\xi T_u g)(t) = g(t - u) e^{2\pi i \xi t}, \quad (3)$$

and the windowed Fourier transform (WFT) of a function $f \in L^2(\mathbb{R})$ is

$$Sf(u, \xi) := \langle f, g_{u,\xi} \rangle = \int_{-\infty}^{\infty} f(t) [g(t - u)]^* e^{-2\pi i \xi t} dt. \quad (4)$$

An example of a windowed Fourier atom is shown in Figure 2.

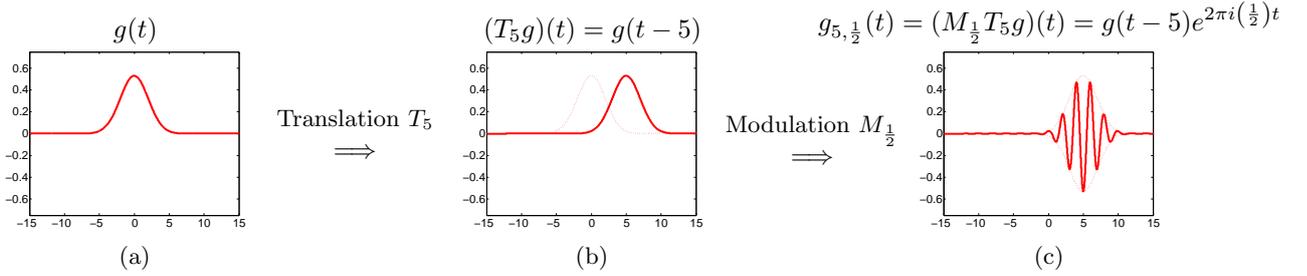


Figure 2: A classical windowed Fourier atom. In this example, the window is a Gaussian with standard deviation equal to 2 and scaled so that $\|g\|_2 = 1$. The real part of the atom $g_{5, \frac{1}{2}}$ is shown in (c).

A second, perhaps more intuitive, way to interpret $Sf(u, \xi)$ is as the Fourier transform of $f(T_u g)^*$, evaluated at frequency ξ . That is, we multiply the signal f by (the complex conjugate of) a translated window $T_u g$ in order to localize the signal to a specific area of interest in time, and then perform Fourier analysis on this localized, windowed signal. This interpretation is illustrated in Figure 3.

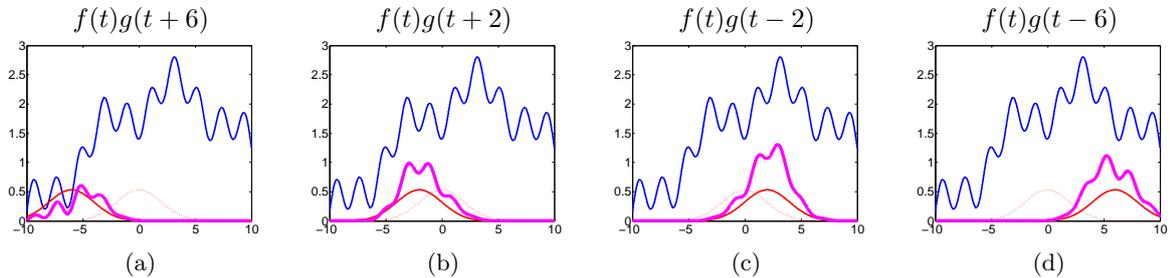


Figure 3: Second interpretation of the windowed Fourier transform: multiply the signal f (shown in blue) by a sliding window (shown in solid red) to obtain a windowed signal (shown in magenta); then take Fourier transforms of the windowed signals.

As mentioned in Section 1, our plan for the rest of the paper is to generalize the translation and modulation operators to the graph setting, and then mimic the classical windowed Fourier transform construction of (3) and (4).

3. Spectral Graph Theory Notation and Background

We consider undirected, connected, weighted graphs $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$, where \mathcal{V} is a finite set of vertices \mathcal{V} with $|\mathcal{V}| = N$, \mathcal{E} is a set of edges, and W is a weighted adjacency matrix (see, e.g., [8] for all definitions in this section). A signal $f : \mathcal{V} \rightarrow \mathbb{R}$ defined on the vertices of the graph may be represented as a vector $f \in \mathbb{R}^N$, where the n^{th} component of the vector f represents the signal value at the n^{th} vertex in \mathcal{V} . The non-normalized graph Laplacian is defined as $\mathcal{L} := D - W$, where D is the diagonal degree matrix. We denote by d the vector of degrees (i.e., the diagonal elements of D), so that $d_n = \sum_{m \neq n} W_{mn}$ is the degree of vertex n . Then $d_{\min} := \min_n \{d_n\}$ and $d_{\max} := \max_n \{d_n\}$.

As the graph Laplacian \mathcal{L} is a real symmetric matrix, it has a complete set of orthonormal eigenvectors, which we denote by $\{\chi_\ell\}_{\ell=0,1,\dots,N-1}$. Without loss of generality, we assume that the associated real, non-negative Laplacian eigenvalues are ordered as $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_{N-1} := \lambda_{\max}$, and we denote the graph Laplacian spectrum by $\sigma(\mathcal{L}) := \{\lambda_0, \lambda_1, \dots, \lambda_{N-1}\}$.

3.1. The Graph Fourier Transform and the Graph Spectral Domain

The classical Fourier transform is the expansion of a function f in terms of the eigenfunctions of the Laplace operator, i.e., $\hat{f}(\xi) = \langle f, e^{2\pi i \xi t} \rangle$. Analogously, the *graph Fourier transform* \hat{f} of a function $f \in \mathbb{R}^N$ on the vertices of \mathcal{G} is the expansion of f in terms of the eigenfunctions of the graph Laplacian. It is defined by

$$\hat{f}(\lambda_\ell) := \langle f, \chi_\ell \rangle = \sum_{n=1}^N f(n) \chi_\ell^*(n), \quad (5)$$

where we adopt the convention that the inner product be conjugate-linear in the second argument. The *inverse graph Fourier transform* is then given by

$$f(n) = \sum_{\ell=0}^{N-1} \hat{f}(\lambda_\ell) \chi_\ell(n). \quad (6)$$

With this definition of the graph Fourier transform, the Parseval relation holds; i.e., for any $f, g \in \mathbb{R}^N$,

$$\langle f, g \rangle = \langle \hat{f}, \hat{g} \rangle,$$

and thus

$$\sum_{n=1}^N |f(n)|^2 = \|f\|_2^2 = \langle f, f \rangle = \langle \hat{f}, \hat{f} \rangle = \|\hat{f}\|_2^2 = \sum_{\ell=0}^{N-1} |\hat{f}(\lambda_\ell)|^2.$$

Note that the definitions of the graph Fourier transform and its inverse in (5) and (6) depend on the choice of graph Laplacian eigenvectors, which is not necessarily unique. Throughout this paper, we do not specify how to choose these eigenvectors, but assume they are fixed. The ideal choice of the eigenvectors in order to optimize the theoretical analysis conducted here and elsewhere remains an interesting open question; however, in most applications with extremely large graphs, the explicit computation of a full eigendecomposition is not practical anyhow, and methods that only utilize the graph Laplacian through sparse matrix-vector multiplication are preferred. We discuss these computational issues further in Section 6.7.1.

It is also possible to use other bases to define the forward and inverse graph Fourier transforms. The eigenvectors of the normalized graph Laplacian $\tilde{\mathcal{L}} := D^{-\frac{1}{2}} \mathcal{L} D^{-\frac{1}{2}}$ comprise one such basis that is prominent in the graph signal processing literature. While we use the non-normalized graph Laplacian eigenvectors as the Fourier basis throughout this paper, the normalized graph Laplacian eigenvectors can also be used to define generalized translation and modulation operators, and we comment briefly on the resulting differences in the Appendix.

We consider signals' representations in both the vertex domain (analogous to the time/space domains in classical Euclidean settings) and the graph spectral domain (analogous to the frequency domain in classical settings). As an example, in Figure 4, we show these two different representations of the signal from Figure 1(c). In this case, we actually generate the signal in the graph spectral domain, starting with a continuous kernel, $\hat{f} : [0, \lambda_{\max}] \rightarrow \mathbb{R}$, given by $\hat{f}(\lambda_\ell) := Ce^{-\tau\lambda_\ell}$, where $\tau = 5$. We then form the discrete signal \hat{f} by evaluating the continuous kernel $\hat{f}(\cdot)$ at each of the graph Laplacian eigenvalues in $\sigma(\mathcal{L})$. The constant C is chosen so that $\|f\|_2 = 1$, and the kernel $\hat{f}(\cdot)$ is referred to as a normalized heat kernel (see, e.g., [8, Chapter 10]). Finally, we generate the signal shown in both Figure 1(c) and 4(a) by taking the inverse graph Fourier transform (6) of \hat{f} .

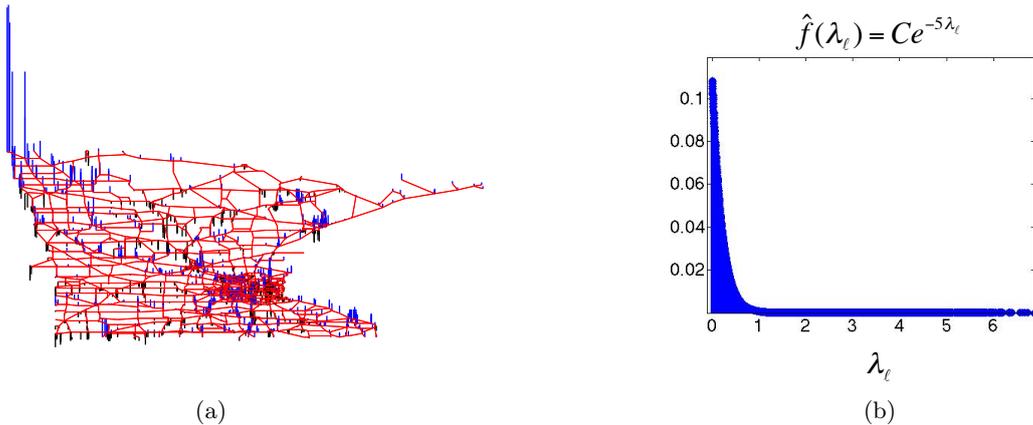


Figure 4: A signal f represented in two domains. (a) The vertex domain. (b) The graph spectral domain.

Some intuition about the graph spectrum can also be carried over from the classical setting to the graph setting. In the classical setting, the Laplacian eigenfunctions (complex exponentials) associated with lower eigenvalues (frequencies) are relatively smooth, whereas those associated with higher eigenvalues oscillate more rapidly. The graph Laplacian eigenvalues and associated eigenvectors satisfy

$$\lambda_\ell = \chi_\ell^\top \mathcal{L} \chi_\ell = \sum_{(m,n) \in \mathcal{E}} W_{mn} [\chi_\ell(m) - \chi_\ell(n)]^2.$$

Therefore, since each term in the summation of the right-hand side is non-negative, the eigenvectors associated with smaller eigenvalues are smoother; i.e., the component differences between neighboring vertices are small (see, e.g., [2, Figure 2]). As the eigenvalue or “frequency” increases, larger differences in neighboring components of the graph Laplacian eigenvectors may be present (see [2, 9] for further discussions of notions of frequency for the graph Laplacian eigenvalues). This well-known property has been extensively utilized in a wide range of problems, including spectral clustering [10], machine learning [11, Section III], and ill-posed inverse problems in image processing [12].

3.2. Localization of Graph Laplacian Eigenvectors and Coherence

There have recently been a number of interesting research results concerning the localization properties of graph Laplacian eigenvectors. For different classes of random graphs, [13, 14, 15] show that with high probability for graphs of sufficiently large size, the eigenvectors of the graph Laplacian (or in some cases, the graph adjacency operator), are delocalized; i.e., the restriction of the eigenvector to a large set must have substantial energy, or in even stronger statements, the element of the matrix $\chi := [\chi_0, \chi_1, \dots, \chi_{N-1}]$ with the largest absolute value is small. We refer to this latter value as the *mutual coherence* (or simply

coherence) between the basis of Kronecker deltas on the graph and the basis of graph Laplacian eigenvectors:

$$\mu := \max_{\substack{\ell \in \{0, 1, \dots, N-1\} \\ i \in \{1, 2, \dots, N\}}} |\langle \chi_\ell, \delta_i \rangle| \in \left[\frac{1}{\sqrt{N}}, 1 \right], \quad (7)$$

where

$$\delta_i(n) = \begin{cases} 1, & \text{if } i = n \\ 0, & \text{otherwise} \end{cases}.$$

While the previously mentioned non-localization results rely on estimates from random matrix theory, Brooks and Lindenstrauss [16] also show that for sufficiently large, unweighted, non-random, regular graphs that do not have too many short cycles through the same vertex, in order for $\sum_{i \in \mathcal{S}} |\chi_\ell(i)|^2 > \epsilon$ for any ℓ , the subset $\mathcal{S} \subset \mathcal{V}$ must satisfy $|\mathcal{S}| \geq N^\delta$, where the constant δ depends on both ϵ and structural restrictions placed on the graph.

These non-localization results are consistent with the intuition one might gain from considering the eigenvectors of the Laplacian for the unweighted path and ring graphs shown in Figure 5. The eigenvalues of the graph Laplacian of the unweighted path graph with N vertices are given by

$$\lambda_\ell = 2 - 2 \cos \left(\frac{\pi \ell}{N} \right), \quad \forall \ell \in \{0, 1, \dots, N-1\},$$

and one possible choice of associated orthonormal eigenvectors is

$$\begin{aligned} \chi_0(n) &= \frac{1}{\sqrt{N}}, \quad \forall n \in \{1, 2, \dots, N\}, \text{ and} \\ \chi_\ell(n) &= \sqrt{\frac{2}{N}} \cos \left(\frac{\pi \ell (n - 0.5)}{N} \right) \text{ for } \ell = 1, 2, \dots, N-1. \end{aligned} \quad (8)$$

These graph Laplacian eigenvectors, which are shown in Figure 5(b), are the basis vectors in the Discrete Cosine Transform (DCT-II) transform [17] used in JPEG image compression. Like the continuous complex exponentials, they are non-localized, globally oscillating functions. The *unordered* eigenvalues of the graph Laplacian of the unweighted ring graph with N vertices are given by (see, e.g., [18, Chapter 3], [19, Proposition 1])

$$\bar{\lambda}_\ell = 2 - 2 \cos \left(\frac{2\pi \ell}{N} \right), \quad \forall \ell \in \{0, 1, \dots, N-1\},$$

and one possible choice of associated orthonormal eigenvectors is

$$\chi_\ell = \frac{1}{\sqrt{N}} \left[1, \omega^\ell, \omega^{2\ell}, \dots, \omega^{(N-1)\ell} \right]^\top, \text{ where } \omega = e^{\frac{2\pi j}{N}}. \quad (9)$$

These eigenvectors correspond to the columns of the Discrete Fourier Transform (DFT) matrix. With this choice of eigenvectors, the coherence of the unweighted ring graph with N vertices is $\frac{1}{\sqrt{N}}$, the smallest possible coherence of any graph. In this case, the basis of DFT columns and the basis of Kronecker deltas on vertices of the graph are said to be *mutually unbiased bases*.

However, empirical studies such as [20] show that certain graph Laplacian eigenvectors may in fact be highly localized, especially when the graph features one or more vertices whose degrees are significantly higher or lower than the average degree, or when the graph features a high degree of clustering (many triangles in the graph). Moreover, Saito and Woei [21] identify a class of starlike trees with highly-localized eigenvectors, some of which are even close to a delta on the vertices of the graph. The following example shows two less structured graphs that also have high coherence.

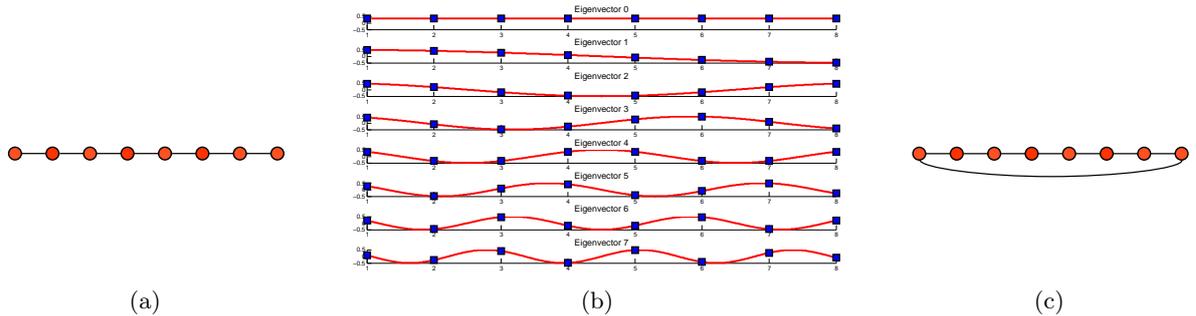


Figure 5: (a) The path graph. (b) One possible choice of the graph Laplacian eigenvectors of the path graph is the discrete cosines. (c) The ring graph.

Example 1: In Figure 6, we show two weighted, undirected graphs with coherences of 0.96 and 0.94. The sensor network is generated by randomly placing 500 nodes in the $[0, 1] \times [0, 1]$ plane. The Swiss roll graph is generated by randomly sampling 1000 points from the two-dimensional Swiss roll manifold embedded in \mathbb{R}^3 . In both cases, the weights are assigned with a thresholded Gaussian kernel weighting function based on the Euclidean distance between nodes:

$$W_{ij} = \begin{cases} \exp\left(-\frac{[d(i,j)]^2}{2\sigma_1^2}\right) & \text{if } d(i,j) \leq \sigma_2 \\ 0 & \text{otherwise} \end{cases}. \quad (10)$$

For the sensor network, $\sigma_1 = 0.074$ and $\sigma_2 = 0.075$. For the Swiss roll graph, $\sigma_1 = 0.100$ and $\sigma_2 = 0.215$. The coherences are based on the orthonormal Laplacian eigenvectors computed by MATLAB's svd function.

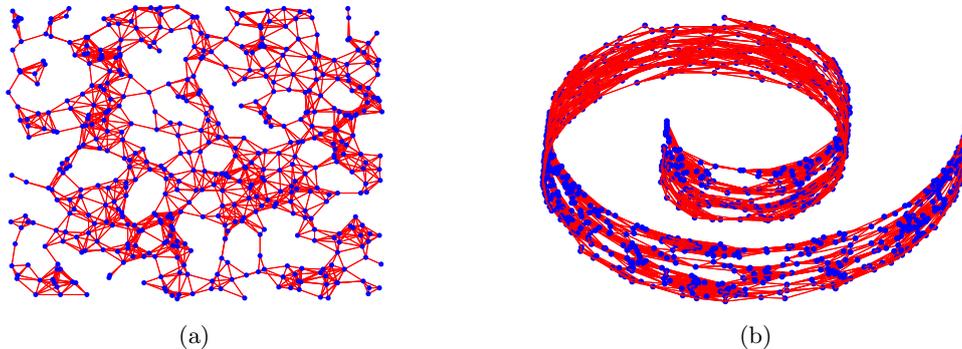


Figure 6: Two graphs with high coherence. (a) The coherence of this random sensor network is 0.96. (b) The coherence of this graph on a cloud of 1000 points randomly sampled on the Swiss roll manifold is 0.94.

The existence of localized eigenvectors can limit the degree to which our intuition from classical time-frequency analysis extends to localized vertex-frequency analysis of signals on graphs. We discuss this point in more detail in Sections 4.2, 6.6, and 6.7.

Finally, we define some quantities that are closely related to the coherence μ and will also be useful in our analysis. We denote the largest absolute value of the elements of a given graph Laplacian eigenvector by

$$\mu_\ell := \|\chi_\ell\|_\infty = \max_{i \in \{1, 2, \dots, N\}} |\chi_\ell(i)|, \quad (11)$$

and the largest absolute value of a given row of χ by

$$\nu_i := \max_{\ell \in \{0, 1, \dots, N-1\}} |\chi_\ell(i)|. \quad (12)$$

Note that

$$\mu = \max_{\ell \in \{0,1,\dots,N-1\}} \{\mu_\ell\} = \max_{i \in \{1,2,\dots,N\}} \{\nu_i\}. \quad (13)$$

4. Generalized Convolution and Translation Operators

The main objective of this section is to define a generalized translation operator that allows us to shift a window around the vertex domain so that it is localized around any given vertex, just as we shift a window along the real line to any center point in the classical windowed Fourier transform for signals on the real line. We use a generalized notion of translation that is – aside from a constant factor that depends on the number of vertices in the graph – the same notion used as one component of the spectral graph wavelet transform (SGWT) in [22, Section 4]. However, in order to leverage intuition from classical time-frequency analysis, we motivate its definition differently here by first defining a generalized convolution operator for signals on graphs. We then discuss and analyze a number of properties of the generalized translation as a standalone operator, including the localization of translated kernels.

4.1. Generalized Convolution of Signals on Graphs

For signals $f, g \in L^2(\mathbb{R})$, the classical convolution product $h = f * g$ is defined as

$$h(t) = (f * g)(t) := \int_{\mathbb{R}} f(\tau)g(t - \tau)d\tau. \quad (14)$$

Since the simple translation $g(t - \tau)$ cannot be directly extended to the graph setting, we cannot directly generalize (14). However, the classical convolution product also satisfies

$$h(t) = (f * g)(t) = \int_{\mathbb{R}} \hat{h}(k)\psi_k(t)dk = \int_{\mathbb{R}} \hat{f}(k)\hat{g}(k)\psi_k(t)dk, \quad (15)$$

where $\psi_k(t) = e^{2\pi ikt}$. This important property that convolution in the time domain is equivalent to multiplication in the Fourier domain is the notion we generalize instead. Specifically, by replacing the complex exponentials in (15) with the graph Laplacian eigenvectors, we define a *generalized convolution* of signals $f, g \in \mathbb{R}^N$ on a graph by

$$(f * g)(n) := \sum_{\ell=0}^{N-1} \hat{f}(\lambda_\ell)\hat{g}(\lambda_\ell)\chi_\ell(n). \quad (16)$$

Using notation from the theory of matrix functions [23], we can also write the generalized convolution as

$$f * g = \hat{g}(\mathcal{L})f = \chi \begin{bmatrix} \hat{g}(\lambda_0) & & 0 \\ & \ddots & \\ 0 & & \hat{g}(\lambda_{N-1}) \end{bmatrix} \chi^* f.$$

Proposition 1: *The generalized convolution product defined in (16) satisfies the following properties:*

1. *Generalized convolution in the vertex domain is multiplication in the graph spectral domain:*

$$\widehat{f * g} = \hat{f}\hat{g}. \quad (17)$$

2. *Let $\alpha \in \mathbb{R}$ be arbitrary. Then*

$$\alpha(f * g) = (\alpha f) * g = f * (\alpha g). \quad (18)$$

3. *Commutativity:*

$$f * g = g * f. \quad (19)$$

4. *Distributivity:*

$$f * (g + h) = f * g + f * h. \quad (20)$$

5. *Associativity:*

$$(f * g) * h = f * (g * h). \quad (21)$$

6. *Define a function $g_0 \in \mathbb{R}^N$ by $g_0(n) := \sum_{\ell=0}^{N-1} \chi_\ell(n)$. Then g_0 is an identity for the generalized convolution product:*

$$f * g_0 = f. \quad (22)$$

7. *An invariance property with respect to the graph Laplacian (a difference operator):*

$$\mathcal{L}(f * g) = (\mathcal{L}f) * g = f * (\mathcal{L}g). \quad (23)$$

8. *The sum of the generalized convolution product of two signals is a constant times the product of the sums of the two signals:*

$$\sum_{n=1}^N (f * g)(n) = \sqrt{N} \hat{f}(0) \hat{g}(0) = \frac{1}{\sqrt{N}} \left[\sum_{n=1}^N f(n) \right] \left[\sum_{n=1}^N g(n) \right]. \quad (24)$$

4.2. Generalized Translation on Graphs

The application of the classical translation operator T_u defined in (1) to a function $f \in L^2(\mathbb{R})$ can be seen as a convolution with δ_u :

$$(T_u f)(t) := f(t - u) = (f * \delta_u)(t) \stackrel{(15)}{=} \int_{\mathbb{R}} \hat{f}(k) \widehat{\delta_u}(k) \psi_k(t) dk = \int_{\mathbb{R}} \hat{f}(k) \psi_k^*(u) \psi_k(t) dk,$$

where the equalities are in the weak sense. Thus, for any signal $f \in \mathbb{R}^N$ defined on the the graph \mathcal{G} and any $i \in \{1, 2, \dots, N\}$, we also define a *generalized translation operator* $T_i : \mathbb{R}^N \rightarrow \mathbb{R}^N$ via generalized convolution with a delta centered at vertex i :

$$(T_i f)(n) := \sqrt{N} (f * \delta_i)(n) \stackrel{(16)}{=} \sqrt{N} \sum_{\ell=0}^{N-1} \hat{f}(\lambda_\ell) \chi_\ell^*(i) \chi_\ell(n). \quad (25)$$

The translation (25) is a kernelized operator. The window to be shifted around the graph is defined in the graph spectral domain via the kernel $\hat{f}(\cdot)$. To translate this window to vertex i , the ℓ^{th} component of the kernel is multiplied by $\chi_\ell^*(i)$, and then an inverse graph Fourier transform is applied. As an example, in Figure 7, we apply generalized translation operators to the normalized heat kernel from Figure 1(c). We can see that doing so has the desired effect of shifting a window around the graph, centering it at any given vertex i .

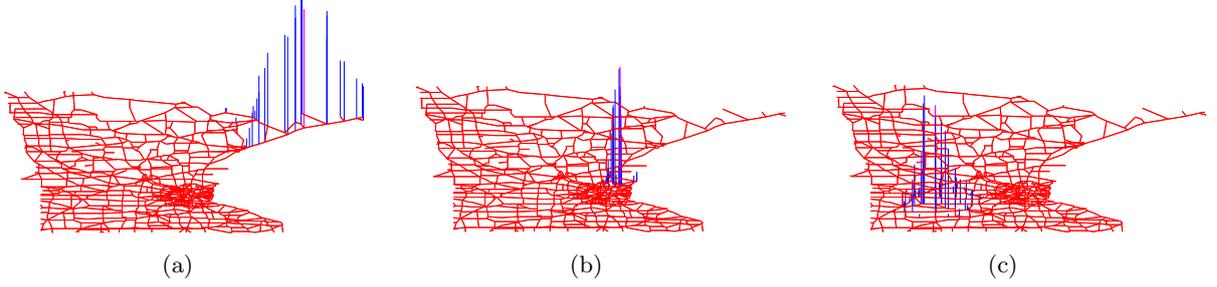


Figure 7: The translated signals (a) $T_{200}f$, (b) $T_{1000}f$, and (c) $T_{2000}f$, where f , the signal shown in Figure 1(c), is a normalized heat kernel satisfying $\hat{f}(\lambda_\ell) = Ce^{-5\lambda_\ell}$. The component of the translated signal at the center vertex is highlighted in magenta.

4.3. Properties of the Generalized Translation Operator

Some expected properties of the generalized translation operator follow immediately from the generalized convolution properties of Proposition 1.

Corollary 1: For any $f, g \in \mathbb{R}^N$ and $i, j \in \{1, 2, \dots, N\}$,

1. $T_i(f * g) = (T_i f) * g = f * (T_i g)$.
2. $T_i T_j f = T_j T_i f$.
3. $\sum_{n=1}^N (T_i f)(n) = \sqrt{N} \hat{f}(0) = \sum_{n=1}^N f(n)$.

However, the niceties end there, and we should also point out some properties that are true for the classical translation operator, but not for the generalized translation operator for signals on graphs. First, unlike the classical case, the set of translation operators $\{T_i\}_{i \in \{1, 2, \dots, N\}}$ do not form a mathematical group; i.e., $T_i T_j \neq T_{i+j}$. In the very special case of *shift-invariant* graphs [24, p. 158], which are graphs for which the DFT basis vectors (9) are graph Laplacian eigenvectors (the unweighted ring graph shown in Figure 5(c) is one such graph), we have

$$T_i T_j = T_{[(i-1)+(j-1) \bmod N] + 1}, \quad \forall i, j \in \{1, 2, \dots, N\}. \quad (26)$$

However, (26) is not true in general for arbitrary graphs. Moreover, while the idea of successive translations $T_i T_j$ carries a clear meaning in the classical case, it is not a particularly meaningful concept in the graph setting due to our definition of generalized translation as a kernelized operator.

Second, unlike the classical translation operator, the generalized translation operator is not an isometric operator; i.e., $\|T_i f\|_2 \neq \|f\|_2$ for all indices i and signals f . Rather, we have

Lemma 1: For any $f \in \mathbb{R}^N$,

$$|\hat{f}(0)| \leq \|T_i f\|_2 \leq \sqrt{N} \nu_i \|f\|_2 \leq \sqrt{N} \mu \|f\|_2. \quad (27)$$

Proof.

$$\begin{aligned} \|T_i f\|_2^2 &= \sum_{n=1}^N \left(\sqrt{N} \sum_{\ell=0}^{N-1} \hat{f}(\lambda_\ell) \chi_\ell^*(i) \chi_\ell(n) \right)^2 \\ &= N \sum_{\ell=0}^{N-1} \sum_{\ell'=0}^{N-1} \hat{f}(\lambda_\ell) \hat{f}(\lambda_{\ell'}) \chi_\ell^*(i) \chi_{\ell'}^*(i) \sum_{n=1}^N \chi_\ell(n) \chi_{\ell'}(n) \\ &= N \sum_{\ell=0}^{N-1} |\hat{f}(\lambda_\ell)|^2 |\chi_\ell^*(i)|^2 \end{aligned} \quad (28)$$

$$\leq N \nu_i^2 \|f\|_2^2. \quad (29)$$

Substituting $\chi_0(i) = \frac{1}{\sqrt{N}}$ into (28) yields the first inequality in (27). \square

If $\mu = \frac{1}{\sqrt{N}}$, as is the case for the ring graph with the DFT graph Laplacian eigenvectors, then $|\chi_\ell^*(i)|^2 = \frac{1}{N}$ for all i and ℓ , and (28) becomes $\|T_i f\|_2 = \|f\|_2$. However, for general graphs, $|\chi_\ell^*(i)|$ may be small or even zero for some i and ℓ , and thus $\|T_i f\|_2$ may be significantly smaller than $\|f\|_2$, and in fact may even be zero. Meanwhile, if μ is close to 1, then we may also have the case that $\|T_i f\|_2 \approx \sqrt{N}\|f\|_2$. Figures 8 and 9 show examples of $\{\|T_i f\|_2\}_{i=1,2,\dots,N}$ for different graphs and different kernels.

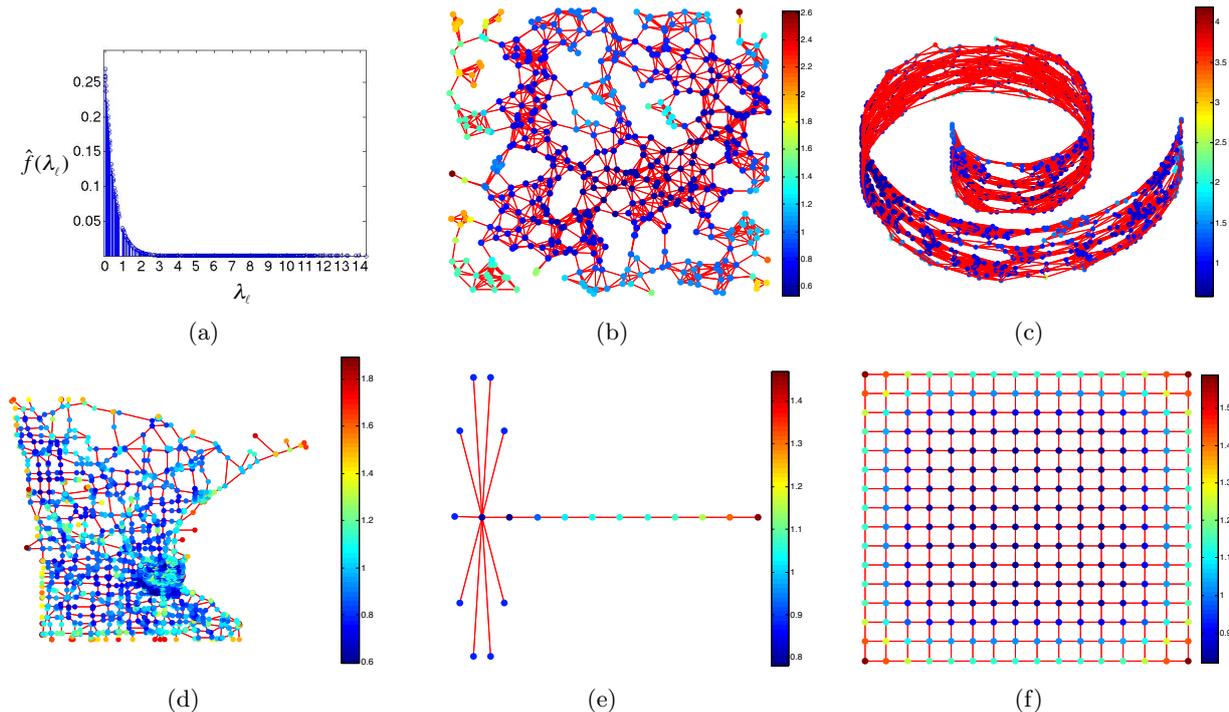


Figure 8: Norms of translated normalized heat kernels with $\tau = 2$. (a) A normalized heat kernel $\hat{f}(\lambda_\ell) = Ce^{-2\lambda_\ell}$ on the sensor network graph shown in (b). (b)-(f) The value at each vertex i represents $\|T_i f\|_2$. The edges of the graphs in (b) and (c) are weighted by a thresholded Gaussian kernel weighting function based on the physical distance between nodes (10), whereas the edges of the graphs in (d)-(f) all have weights equal to one. In all cases, the norms of the translated windows are not too close to zero, and the larger norms tend to be located at the “boundary” vertices in the graph. The lower bound $|\hat{f}(0)|$ and upper bound $\sqrt{N}\mu\|f\|_2$ of Lemma 1 are (b) [0.27,21.38]; (c) [0.20,29.22]; (d) [0.07,42.88]; (e) [0.62,4.26]; (f) [0.36,3.25].

4.4. Localization of Translated Kernels in the Vertex Domain

We now examine to what extent translated kernels are localized in the vertex domain. First, we note that a polynomial kernel with degree K that is translated to a given center vertex is strictly localized in a ball of radius K around the center vertex, where the distance $d_G(\cdot, \cdot)$ used to define the ball is the *geodesic* or *shortest path* distance (i.e., the distance between two vertices is the minimum number of edges in any path connecting them). Note that this choice of distance measure ignores the weights of the edges and only depends on the unweighted adjacency matrix of the graph.

Lemma 2: Let \widehat{p}_K be a polynomial kernel with degree K ; i.e.,

$$\widehat{p}_K(\ell) = \sum_{k=0}^K a_k \lambda_\ell^k \quad (30)$$

for some coefficients $\{a_k\}_{k=0,1,\dots,K}$. If $d_G(i, n) > K$, then $(T_i p_K)(n) = 0$.

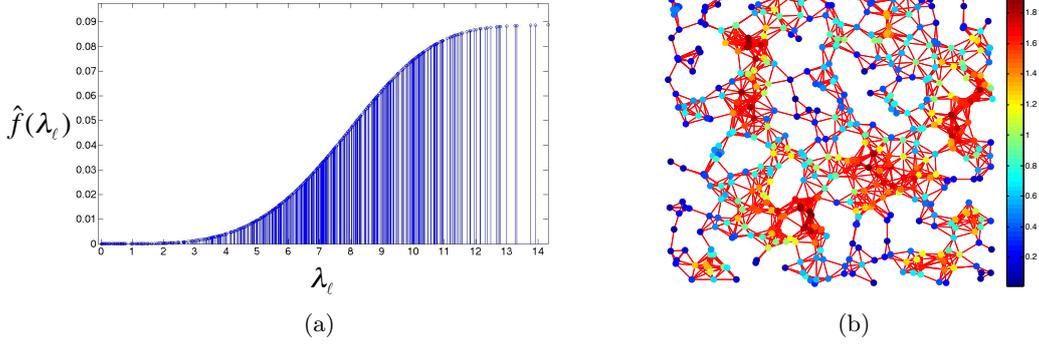


Figure 9: (a) A smooth kernel with its energy concentrated on the higher frequencies of the sensor network's spectrum. (b) The values of $\|T_i f\|_2$. Unlike the normalized heat kernels of Figure 8, the translated kernels may have norms close to zero. In this example, $\|f\|_2 = 1$ and the minimum norm of a translated window is 0.013.

Proof. By [22, Lemma 5.2], $d_G(i, n) > K$ implies $(\mathcal{L}^K)_{i, n} = 0$. Combining this with the fact that

$$(\mathcal{L}^k)_{i, n} = \sum_{\ell=0}^{N-1} \lambda_\ell^k \chi_\ell^*(i) \chi_\ell(n),$$

and with the definitions (25) and (30) of the generalized translation and the polynomial kernel, we have

$$\begin{aligned} (T_i p_K)(n) &= \sqrt{N} \sum_{\ell=0}^{N-1} \widehat{p}_K(\lambda_\ell) \chi_\ell^*(i) \chi_\ell(n) \\ &= \sqrt{N} \sum_{\ell=0}^{N-1} \sum_{k=0}^K a_k \lambda_\ell^k \chi_\ell^*(i) \chi_\ell(n) \\ &= \sqrt{N} \sum_{k=0}^K a_k (\mathcal{L}^k)_{i, n} = 0, \quad \forall i, n \in \mathcal{V} \text{ s.t. } d_G(i, n) > K. \end{aligned}$$

□

More generally, as seen in Figure 7, if we translate a smooth kernel to a given center vertex i , the magnitude of the translated kernel at another vertex n decays as the distance between i and n increases. In Theorem 1, we provide one estimate of this localization by combining the strict localization of polynomial kernels with the following upper bound on the minimax polynomial approximation error.

Lemma 3 ([25, Equation (4.6.10)]): *If a function $f(x)$ is $(K + 1)$ -times continuously differentiable on an interval $[a, b]$, then*

$$\inf_{q_K} \{\|f - q_K\|_\infty\} \leq \frac{\lceil \frac{b-a}{2} \rceil^{K+1}}{(K+1)! 2^K} \|f^{(K+1)}\|_\infty, \quad (31)$$

where the infimum in (31) is taken over all polynomials q_K of degree K .

Theorem 1: *Let $\hat{g} : [0, \lambda_{\max}] \rightarrow \mathbb{R}$ be a kernel, and define $d_{in} := d_G(i, n)$ and $K_{in} := d_{in} - 1$. Then*

$$|(T_i g)(n)| \leq \sqrt{N} \inf_{p_{K_{in}}} \left\{ \sup_{\lambda \in [0, \lambda_{\max}]} |\hat{g}(\lambda) - \widehat{p}_{K_{in}}(\lambda)| \right\} = \sqrt{N} \inf_{p_{K_{in}}} \{\|\hat{g} - \widehat{p}_{K_{in}}\|_\infty\}, \quad (32)$$

where the infimum is taken over all polynomial kernels of degree K_{in} , as defined in (30). More generally, for $p, q \geq 1$ such that $\frac{1}{p} + \frac{1}{q} = 1$,

$$|(T_i g)(n)| \leq \sqrt{N} \mu^{\frac{2(q-1)}{q}} \inf_{\widehat{p_{K_{in}}}} \{ \|\hat{g} - \widehat{p_{K_{in}}}\|_p \}. \quad (33)$$

Proof. From Lemma 2, $(T_i p_{K_{in}})(n) = 0$ for all polynomial kernels $\widehat{p_{K_{in}}}$ of degree K_{in} . Thus, we have

$$\begin{aligned} |(T_i g)(n)| &= \inf_{\widehat{p_{K_{in}}}} |(T_i g)(n) - (T_i p_{K_{in}})(n)| = \inf_{\widehat{p_{K_{in}}}} \left| \sqrt{N} \sum_{\ell=0}^{N-1} [\hat{g}(\lambda_\ell) - \widehat{p_{K_{in}}}(\lambda_\ell)] \chi_\ell^*(i) \chi_\ell(n) \right| \\ &\leq \sqrt{N} \inf_{\widehat{p_{K_{in}}}} \left\{ \sum_{\ell=0}^{N-1} \left| [\hat{g}(\lambda_\ell) - \widehat{p_{K_{in}}}(\lambda_\ell)] \chi_\ell^*(i) \chi_\ell(n) \right| \right\} \\ &\leq \sqrt{N} \inf_{\widehat{p_{K_{in}}}} \left\{ \left(\sum_{\ell=0}^{N-1} |\hat{g}(\lambda_\ell) - \widehat{p_{K_{in}}}(\lambda_\ell)|^p \right)^{\frac{1}{p}} \left(\sum_{\ell=0}^{N-1} |\chi_\ell^*(i) \chi_\ell(n)|^q \right)^{\frac{1}{q}} \right\} \end{aligned} \quad (34)$$

$$\leq \sqrt{N} \mu^{\frac{2(q-1)}{q}} \inf_{\widehat{p_{K_{in}}}} \{ \|\hat{g} - \widehat{p_{K_{in}}}\|_p \}, \quad (35)$$

where (34) follows from Hölder's inequality, and (35) follows from

$$\sum_{\ell=0}^{N-1} |\chi_\ell^*(i) \chi_\ell(n)|^q \leq \mu^{2(q-1)} \sum_{\ell=0}^{N-1} |\chi_\ell^*(i) \chi_\ell(n)| \leq \mu^{2(q-1)} \sqrt{\sum_{\ell=0}^{N-1} |\chi_\ell(i)|^2} \sqrt{\sum_{\ell=0}^{N-1} |\chi_\ell(n)|^2} = \mu^{2(q-1)}.$$

□

Substituting (31) into (35) yields the following corollary to Theorem 1.

Corollary 2: *If $\hat{g}(\cdot)$ is d_{in} -times continuously differentiable on $[0, \lambda_{\max}]$, then*

$$|(T_i g)(n)| \leq \left[\frac{2\sqrt{N}}{d_{in}!} \left(\frac{\lambda_{\max}}{4} \right)^{d_{in}} \right] \sup_{\lambda \in [0, \lambda_{\max}]} |\hat{g}^{(d_{in})}(\lambda)|. \quad (36)$$

When the kernel $\hat{g}(\cdot)$ has a significant DC component, as is the case for the most logical candidate window functions, such as the heat kernel, then we can combine Theorem 1 and Corollary 2 with the lower bound on the norm of a translated kernel from Lemma 1 to show that the energy of the translated kernel is localized around the center vertex i .

Corollary 3: *If $\hat{g}(0) \neq 0$, then*

$$\frac{|(T_i g)(n)|}{\|T_i g\|_2} \leq \frac{\sqrt{N}}{|\hat{g}(0)|} \inf_{\widehat{p_{K_{in}}}} \left\{ \sup_{\lambda \in [0, \lambda_{\max}]} |\hat{g}(\lambda) - \widehat{p_{K_{in}}}(\lambda)| \right\}. \quad (37)$$

Moreover, if $\hat{g}(\cdot)$ is d_{in} -times continuously differentiable on $[0, \lambda_{\max}]$, then

$$\frac{|(T_i g)(n)|}{\|T_i g\|_2} \leq \left[\frac{2\sqrt{N}}{d_{in}! |\hat{g}(0)|} \left(\frac{\lambda_{\max}}{4} \right)^{d_{in}} \right] \sup_{\lambda \in [0, \lambda_{\max}]} |\hat{g}^{(d_{in})}(\lambda)|. \quad (38)$$

Example 2: *If $\hat{g}(\lambda) = e^{-\tau\lambda}$, then $\hat{g}(0) = 1$, $\sup_{\lambda \in [0, \lambda_{\max}]} |\hat{g}^{(d_{in})}(\lambda)| = \tau^{d_{in}}$, and (38) becomes*

$$\frac{|(T_i g)(n)|}{\|T_i g\|_2} \leq \frac{2\sqrt{N}}{d_{in}!} \left(\frac{\tau \lambda_{\max}}{4} \right)^{d_{in}} \leq \sqrt{\frac{2N}{d_{in}\pi}} e^{-\frac{1}{12d_{in}+1}} \left(\frac{\tau \lambda_{\max} e}{4d_{in}} \right)^{d_{in}}, \quad (39)$$

where the second inequality follows from Stirling's approximation: $m! \geq \sqrt{2\pi m} \left(\frac{m}{e}\right)^m e^{\frac{1}{12m+1}}$ (see, e.g. [26, p. 257]). Interestingly, a term of the form $\left(\frac{C\tau\epsilon}{d_{in}}\right)^{d_{in}}$ also appears in the upper bound of [27, Theorem 1], which is specifically tailored to the heat kernel and derived via a rather different approach.

Agaskar and Lu [28, 29] define the graph spread of a signal f around a given vertex i as

$$\Delta_i^2(f) := \frac{1}{\|f\|_2^2} \sum_{n=1}^N [dg(i, n)]^2 [f(n)]^2.$$

We can now use (39) to bound the spread of a translated heat kernel around the center vertex i in terms of the diffusion parameter τ as follows:

$$\begin{aligned} \Delta_i^2(T_i g) &= \frac{1}{\|T_i g\|_2^2} \sum_{n=1}^N d_{in}^2 |T_i g(n)|^2 \\ &= \sum_{n \neq i} d_{in}^2 \left[\frac{|T_i g(n)|}{\|T_i g\|_2} \right]^2 \\ &\stackrel{(39)}{\leq} 4N \sum_{n \neq i} \frac{1}{[(d_{in} - 1)!]^2} \left(\frac{\tau^2 \lambda_{\max}^2}{16} \right)^{d_{in}} \\ &= \frac{N\tau^2 \lambda_{\max}^2}{4} \sum_{r=1}^{\text{diam}(\mathcal{G})} |\mathcal{R}(i, r)| \frac{1}{[(r-1)!]^2} \left(\frac{\tau^2 \lambda_{\max}^2}{16} \right)^{r-1}, \end{aligned} \quad (40)$$

where $|\mathcal{R}(i, r)|$ is the number of vertices whose distance from vertex i is exactly r . Note that

$$|\mathcal{R}(i, r)| \leq d_i (d_{\max} - 1)^{r-1}, \quad (41)$$

so, substituting (41) into (40), we have

$$\begin{aligned} \Delta_i^2(T_i g) &\leq \frac{N\tau^2 \lambda_{\max}^2}{4} \sum_{r=1}^{\text{diam}(\mathcal{G})} \frac{d_i}{[(r-1)!]^2} \left(\frac{\tau^2 \lambda_{\max}^2}{16(d_{\max} - 1)} \right)^{r-1} \\ &= \frac{N\tau^2 \lambda_{\max}^2 d_i}{4} \sum_{r=0}^{\text{diam}(\mathcal{G})-1} \frac{1}{[(r)!]^2} \left(\frac{\tau^2 \lambda_{\max}^2}{16(d_{\max} - 1)} \right)^r \\ &\leq \frac{N\tau^2 \lambda_{\max}^2 d_i}{4} \sum_{r=0}^{\infty} \frac{1}{r!} \left(\frac{\tau^2 \lambda_{\max}^2}{16(d_{\max} - 1)} \right)^r \\ &= \frac{N\tau^2 \lambda_{\max}^2 d_i}{4} \exp \left(\frac{\tau^2 \lambda_{\max}^2}{16(d_{\max} - 1)} \right), \end{aligned} \quad (42)$$

where the final equality in (42) follows from the Taylor series expansion of the exponential function around zero. While the above bounds can be quite loose (in particular for weighted graphs as we have not incorporated the graph weights into the bounds), the analysis nonetheless shows that we can control the spread of translated heat kernels around their center vertices through the diffusion parameter τ . For any $\epsilon > 0$, in order to ensure $\Delta_i^2(T_i g) \leq \epsilon$, it suffices to take

$$\tau \leq \frac{4}{\lambda_{\max}} \sqrt{(d_{\max} - 1) \Omega \left(\frac{\epsilon}{4N d_i (d_{\max} - 1)} \right)}, \quad (43)$$

where $\Omega(\cdot)$ is the Lambert W function. Note that the right-hand side of (43) is increasing in ϵ .

In the limit, as $\tau \rightarrow 0$, $T_i g \rightarrow \delta_i$, and the spread $\Delta_i^2(T_i g) \rightarrow 0$. On the other hand, as $\tau \rightarrow \infty$, $T_i g(n) \rightarrow \frac{1}{\sqrt{N}}$ for all n , $\|T_i g\|_2 \rightarrow 1$, and $\Delta_i^2(T_i g) \rightarrow \frac{1}{N} \sum_{n=1}^N d_{in}^2$.

5. Generalized Modulation of Signals on Graphs

Motivated by the fact that the classical modulation (2) is a multiplication by a Laplacian eigenfunction, we define, for any $k \in \{0, 1, \dots, N-1\}$, a *generalized modulation operator* $M_k : \mathbb{R}^N \rightarrow \mathbb{R}^N$ by

$$(M_k f)(n) := \sqrt{N} f(n) \chi_k(n). \quad (44)$$

5.1. Localization of Modulated Kernels in the Graph Spectral Domain

Note first that, for connected graphs, M_0 is the identity operator, as $\chi_0(n) = \frac{1}{\sqrt{N}}$ for all n . In the classical case, the modulation operator represents a translation in the Fourier domain:

$$\widehat{M_\xi f}(\omega) = \hat{f}(\omega - \xi), \forall \omega \in \mathbb{R}.$$

This property is not true in general for our modulation operator on graphs due to the discrete nature of the graph. However, we do have the nice property that if $\hat{g}(\ell) = \delta_0(\lambda_\ell)$, then

$$\begin{aligned} \widehat{M_k g}(\lambda_\ell) &= \sum_{n=1}^N \chi_\ell^*(n) (M_k g)(n) \\ &= \sum_{n=1}^N \chi_\ell^*(n) \sqrt{N} \chi_k(n) \frac{1}{\sqrt{N}} = \delta_0(\lambda_\ell - \lambda_k) = \begin{cases} 1, & \text{if } \lambda_\ell = \lambda_k \\ 0, & \text{otherwise} \end{cases}, \end{aligned}$$

so M_k maps the DC component of any signal $f \in \mathbb{R}^N$ to $\hat{f}(0)\chi_k$. Moreover, if we start with a function f that is localized around the eigenvalue 0 in the graph spectral domain, as in Figure 10, then $M_k f$ will be localized around the eigenvalue λ_k in the graph spectral domain.

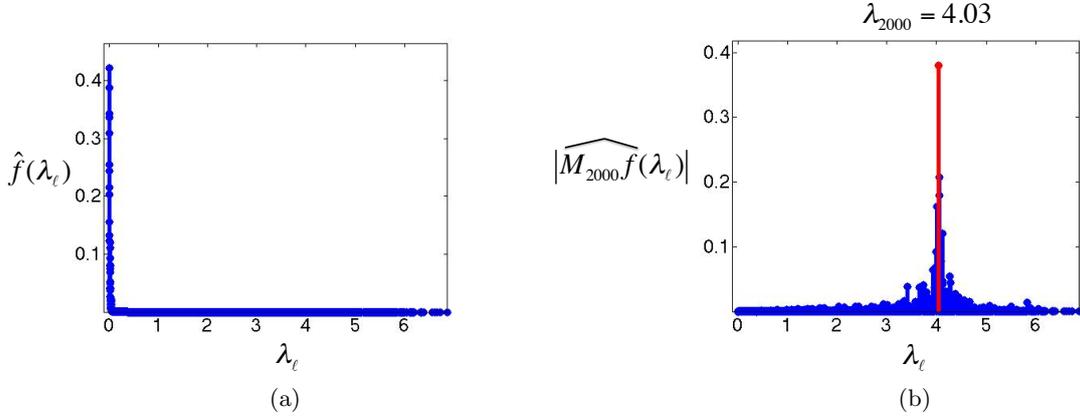


Figure 10: (a) The graph spectral representation of a signal f with $\hat{f}(\ell) = Ce^{-100\lambda_\ell}$ on the Minnesota graph, where the constant C is chosen such that $\|f\|_2 = 1$. (b) The graph spectral representation $\widehat{M_{2000}f}$ of the modulated signal $M_{2000}f$. Note that the modulated signal is localized around $\lambda_{2000} = 4.03$ in the graph spectral domain.

We quantify this localization in the next theorem, which is an improved version of [3, Theorem 1].

Theorem 2: *Given a weighted graph \mathcal{G} with N vertices, if for some $\gamma > 0$, a kernel \hat{f} satisfies*

$$\sqrt{N} \sum_{\ell=1}^{N-1} \mu_\ell |\hat{f}(\lambda_\ell)| \leq \frac{|\hat{f}(0)|}{1 + \gamma}, \quad (45)$$

then

$$|\widehat{M_k f}(\lambda_k)| \geq \gamma |\widehat{M_k f}(\lambda_\ell)| \text{ for all } \ell \neq k. \quad (46)$$

Proof.

$$\begin{aligned}
\widehat{M_k f}(\lambda_{\ell'}) &= \sum_{n=1}^N \sqrt{N} \chi_{\ell'}^*(n) \chi_k(n) f(n) = \sum_{n=1}^N \sqrt{N} \chi_{\ell'}^*(n) \chi_k(n) \sum_{\ell''=0}^{N-1} \chi_{\ell''}(n) \hat{f}(\lambda_{\ell''}) \\
&= \sum_{n=1}^N \sqrt{N} \chi_{\ell'}^*(n) \chi_k(n) \left[\frac{\hat{f}(0)}{\sqrt{N}} + \sum_{\ell''=1}^{N-1} \chi_{\ell''}(n) \hat{f}(\lambda_{\ell''}) \right] \\
&= \hat{f}(0) \delta_{\ell'k} + \sum_{n=1}^N \sqrt{N} \chi_{\ell'}^*(n) \chi_k(n) \sum_{\ell''=1}^{N-1} \chi_{\ell''}(n) \hat{f}(\lambda_{\ell''}). \tag{47}
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
|\widehat{M_k f}(\lambda_k)| &= \left| \hat{f}(0) + \sum_{n=1}^N \sqrt{N} |\chi_k(n)|^2 \sum_{\ell''=1}^{N-1} \chi_{\ell''}(n) \hat{f}(\lambda_{\ell''}) \right| \tag{48} \\
&\geq |\hat{f}(0)| - \left| \sum_{n=1}^N \sqrt{N} |\chi_k(n)|^2 \sum_{\ell''=1}^{N-1} \chi_{\ell''}(n) \hat{f}(\lambda_{\ell''}) \right| \\
&\geq |\hat{f}(0)| - \sum_{n=1}^N \sqrt{N} |\chi_k(n)|^2 \sum_{\ell''=1}^{N-1} |\chi_{\ell''}(n)| |\hat{f}(\lambda_{\ell''})| \\
&\geq |\hat{f}(0)| - \sqrt{N} \sum_{\ell''=1}^{N-1} \mu_{\ell''} |\hat{f}(\lambda_{\ell''})| \\
&\geq |\hat{f}(0)| \left(1 - \frac{1}{1+\gamma} \right) \tag{49}
\end{aligned}$$

where the last two inequalities follow from (11) and (60), respectively. Returning to (47) for $\ell \neq k$, we have

$$\begin{aligned}
\gamma |\widehat{M_k f}(\lambda_{\ell})| &= \gamma \left| \sum_{n=1}^N \sqrt{N} \chi_{\ell}^*(n) \chi_k(n) \sum_{\ell''=1}^{N-1} \chi_{\ell''}(n) \hat{f}(\lambda_{\ell''}) \right| \\
&\leq \gamma \sum_{n=1}^N \left| \sqrt{N} \chi_{\ell}^*(n) \chi_k(n) \right| \sum_{\ell''=1}^{N-1} |\chi_{\ell''}(n)| |\hat{f}(\lambda_{\ell''})| \\
&\leq \gamma \sum_{n=1}^N \left| \sqrt{N} \chi_{\ell}^*(n) \chi_k(n) \right| \sum_{\ell''=1}^{N-1} \mu_{\ell''} |\hat{f}(\lambda_{\ell''})| \\
&\leq \gamma \sqrt{N} \sum_{\ell''=1}^{N-1} \mu_{\ell''} |\hat{f}(\lambda_{\ell''})| \\
&\leq |\hat{f}(0)| \left(\frac{\gamma}{1+\gamma} \right) \tag{50}
\end{aligned}$$

where the last three inequalities follow from (11), Hölder's inequality, and (60), respectively. Combining (49) and (50) yields (61). \square

Corollary 4: *Given a weighted graph \mathcal{G} with N vertices, if for some $\gamma > 0$, a kernel \hat{f} satisfies (60), then*

$$\frac{|\widehat{M_k f}(\lambda_k)|^2}{\|M_k f\|_2^2} \geq \frac{\gamma^2}{N + 3 + 4\gamma + \gamma^2}.$$

Proof. We lower bound the numerator by squaring (49). We upper bound the denominator as follows:

$$\|M_k f\|_2^2 = |\widehat{M_k f}(\lambda_k)|^2 + \sum_{\ell \neq k} |\widehat{M_k f}(\lambda_\ell)|^2 \leq |\hat{f}(0)|^2 \left(\frac{2+\gamma}{1+\gamma}\right)^2 + (N-1)|\hat{f}(0)|^2 \left(\frac{1}{1+\gamma}\right)^2$$

which follows from squaring (50) and from applying the triangle inequality at (48), following the same steps until (49), and squaring the result. \square

Remark 1: *It would also be interesting to find conditions on \hat{f} such that*

$$\frac{\sum_{\ell: \lambda_\ell \in \mathcal{B}(\lambda_k, r)} |\widehat{M_k f}(\lambda_\ell)|^2}{\|M_k f\|_2^2} \geq \gamma_2(r),$$

or such that the spread of $\widehat{M_k f}$ around λ_k , defined as either [2, p. 93]

$$\frac{\sum_{\ell=0}^{N-1} (\lambda_\ell - \lambda_k)^2 |\widehat{M_k f}(\lambda_\ell)|^2}{\|M_k f\|_2^2} \quad \text{or} \quad \frac{\sum_{\ell=0}^{N-1} (\sqrt{\lambda_\ell} - \sqrt{\lambda_k})^2 |\widehat{M_k f}(\lambda_\ell)|^2}{\|M_k f\|_2^2} \quad (51)$$

is upper bounded.³ In Section 8.2 of the Appendix, we present an alternative definition of the generalized modulation and localization results regarding that modulation operator that contain a spread form similar to (51).

6. Windowed Graph Fourier Frames

Equipped with these generalized notions of translation and modulation of signals on graphs, we can now define windowed graph Fourier atoms and a windowed graph Fourier transform analogously to (3) and (4) in the classical case.

6.1. Windowed Graph Fourier Atoms and a Windowed Graph Fourier Transform

For a window $g \in \mathbb{R}^N$, we define a windowed graph Fourier atom by⁴

$$g_{i,k}(n) := (M_k T_i g)(n) = N \chi_k(n) \sum_{\ell=0}^{N-1} \hat{g}(\lambda_\ell) \chi_\ell^*(i) \chi_\ell(n), \quad (52)$$

and the windowed graph Fourier transform of a function $f \in \mathbb{R}^N$ by

$$Sf(i, k) := \langle f, g_{i,k} \rangle. \quad (53)$$

Note that as discussed in Section 4.2, we usually define the window directly in the graph spectral domain, as $\hat{g}(\lambda) : [0, \lambda_{\max}] \rightarrow \mathbb{R}$.

³Agaskar and Lu [28, 29] suggest to define the spread of \hat{f} in the graph spectral domain as $\frac{1}{\|f\|_2^2} \sum_{\ell=0}^{N-1} \lambda_\ell |\hat{f}(\lambda_\ell)|^2$. With this definition, the “spread” is always taken around the eigenvalue 0, as opposed to the mean of the signal, and, as a result, the signal with the highest possible spread is actually a (completely localized) Kronecker delta at λ_{\max} in the graph spectral domain, or, equivalently, χ_{\max} in the vertex domain.

⁴An alternative definition of an atom is $g_{i,k} := T_i M_k g$. In the classical setting, the translation and modulation operators do not commute, but the difference between the two definitions of a windowed Fourier atom is a phase factor [6, p. 6]. In the graph setting, it is more difficult to characterize the difference between the two definitions. In our numerical experiments, defining the atoms as $g_{i,k} := M_k T_i g$ tended to lead to more informative analysis when using the modulation definition (44), but defining the atoms as $g_{i,k} := T_i M_k g$ tended to lead to more informative analysis when using the alternative modulation definition presented Section 8.2. In this paper, we always use $g_{i,k} := M_k T_i g$.

Example 3: We consider a random sensor network graph with $N = 64$ vertices, and thresholded Gaussian kernel edge weights (10) with $\sigma_1 = \sigma_2 = 0.2$. We consider the signal f shown in Figure 11(a), and wish to compute the windowed graph Fourier transform coefficient $Sf(27, 11) = \langle f, g_{27,11} \rangle$ using a window kernel $\hat{g}(\lambda_\ell) = Ce^{-\tau\lambda_\ell}$, where $\tau = 3$ and $C = 1.45$ is chosen such that $\|g\|_2 = 1$. The windowed graph Fourier atom $g_{27,11}$ is shown in the vertex and graph spectral domains in Figure 11(b) and 11(c), respectively.

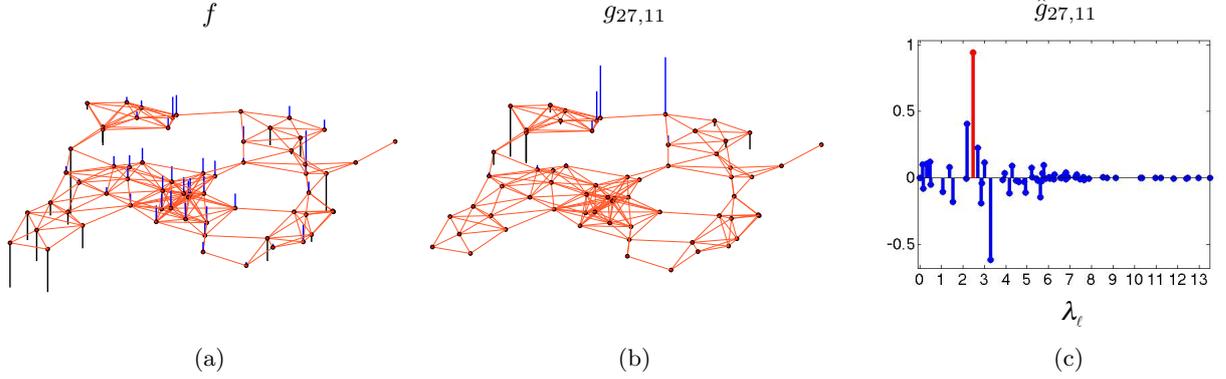


Figure 11: Windowed graph Fourier transform example. (a) A signal f on a random sensor network with 64 vertices. (b) A windowed graph Fourier atom $g_{27,11}$ centered at vertex 27 and frequency $\lambda_{11} = 2.49$. (c) The same atom plotted in the vertex domain and the graph spectral domain. To determine the corresponding windowed graph Fourier transform coefficient, we take an inner product of the signals in (a) and (b): $Sf(27, 11) = \langle f, g_{27,11} \rangle = 0.358$.

As with the classical windowed Fourier transform described in Section 2, we can interpret the computation of the windowed graph Fourier transform coefficients in a second manner. Namely, we can first multiply the signal f by (the complex conjugate of) a translated window:

$$f_{\text{windowed},i}(n) := ((T_i g)^* \cdot f)(n) := [T_i g(n)]^* [f(n)], \quad (54)$$

and then compute the windowed graph Fourier transform coefficients as \sqrt{N} times the graph Fourier transform of the windowed signal:

$$\begin{aligned} \sqrt{N} \hat{f}_{\text{windowed},i}(k) &= \sqrt{N} \langle f_{\text{windowed},i}, \chi_k \rangle = \sqrt{N} \sum_{n=1}^N [T_i g(n)]^* [f(n)] \chi_k^*(n) \\ &= \sum_{n=1}^N f(n) \sum_{\ell=0}^{N-1} N \hat{g}(\lambda_\ell) \chi_\ell(i) \chi_\ell^*(n) \chi_k^*(n) \\ &= \sum_{n=1}^N f(n) g_{ik}^*(n) = \langle f, g_{ik} \rangle = Sf(i, k). \end{aligned}$$

Note that the \cdot in (54) represents component-wise multiplication.

Example 3 (cont.): In Figure 12, we illustrate this alternative interpretation of the windowed graph Fourier transform by first multiplying the signal f of Figure 11(a) by the window $T_{27}g$ (componentwise), and then taking the graph Fourier transform of the resulting windowed signal.

6.2. Frame Bounds

We now provide a simple sufficient condition for the collection of windowed graph Fourier atoms to form a *frame* (see, e.g., [30, 31, 32])

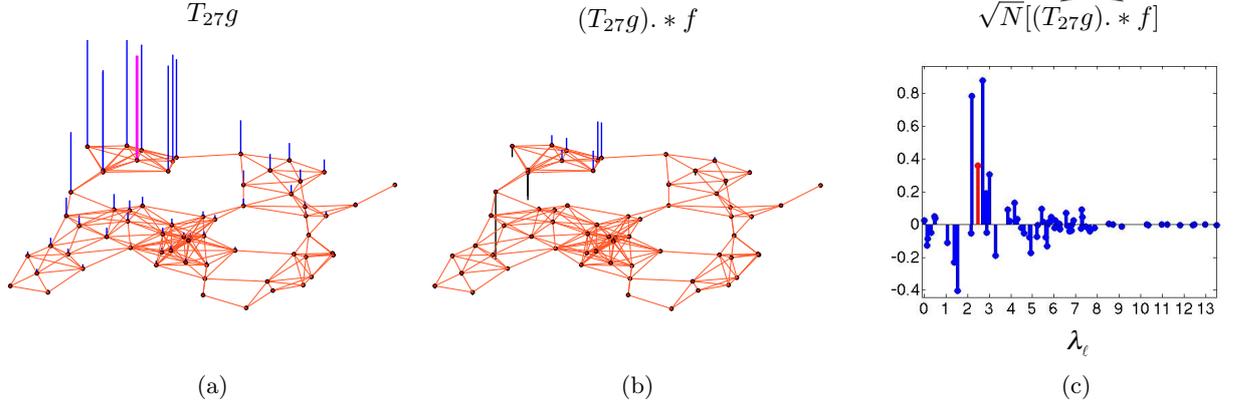


Figure 12: Alternative interpretation of the windowed graph Fourier transform. (a) A translated (real) window centered at vertex 27. (b) We multiply the window by the signal (pointwise) to compute the windowed signal in the vertex domain. (c) The windowed signal in the graph spectral domain (multiplied by a normalizing constant). The corresponding windowed graph Fourier transform coefficient is $\sqrt{N}(\widehat{T_{27}g}) * f(\lambda_{11}) = 0.358$, the same as $\langle f, g_{27,11} \rangle$, which was computed in Figure 11.

Theorem 3: If $\hat{g}(0) \neq 0$, then $\{g_{i,k}\}_{i=1,2,\dots,N; k=0,1,\dots,N-1}$ is a frame; i.e., for all $f \in \mathbb{R}^N$,

$$A\|f\|_2^2 \leq \sum_{i=1}^N \sum_{k=0}^{N-1} |\langle f, g_{i,k} \rangle|^2 \leq B\|f\|_2^2,$$

where

$$0 < N|\hat{g}(0)|^2 \leq A := \min_{n \in \{1,2,\dots,N\}} \{N\|T_n g\|_2^2\} \leq B := \max_{n \in \{1,2,\dots,N\}} \{N\|T_n g\|_2^2\} \leq N^2 \mu^2 \|g\|_2^2.$$

Proof.

$$\begin{aligned} \sum_{i=1}^N \sum_{k=0}^{N-1} |\langle f, g_{i,k} \rangle|^2 &= \sum_{i=1}^N \sum_{k=0}^{N-1} |\langle f, M_k T_i g \rangle|^2 \\ &= N \sum_{i=1}^N \sum_{k=0}^{N-1} |\langle f(T_i g)^*, \chi_k \rangle|^2 \\ &= N \sum_{i=1}^N \langle f(T_i g)^*, f(T_i g)^* \rangle \end{aligned} \quad (55)$$

$$\begin{aligned} &= N \sum_{i=1}^N \sum_{n=1}^N |f(n)|^2 |(T_i g)(n)|^2 \\ &= N \sum_{i=1}^N \sum_{n=1}^N |f(n)|^2 |(T_n g)(i)|^2 \end{aligned} \quad (56)$$

$$= N \sum_{n=1}^N |f(n)|^2 \|T_n g\|_2^2, \quad (57)$$

where (55) is due to Parseval's identity, and (56) follows from the symmetry of \mathcal{L} and the definition (25) of T_i . Moreover, under the hypothesis that $\hat{g}(0) \neq 0$, we have

$$\|T_n g\|_2^2 = N \sum_{\ell=0}^{N-1} |\hat{g}(\lambda_\ell)|^2 |\chi_\ell(n)|^2 \geq |\hat{g}(0)|^2 > 0. \quad (58)$$

Combining (57) and (58), for $f \neq 0$,

$$0 < A\|f\|_2^2 \leq \sum_{i=1}^N \sum_{k=0}^{N-1} |\langle f, g_{i,k} \rangle|^2 \leq B\|f\|_2^2 < \infty.$$

The upper bound on B follows directly from (29). \square

Remark 2: If $\mu = \frac{1}{\sqrt{N}}$, then $\{g_{i,k}\}_{i=1,2,\dots,N; k=0,1,\dots,N-1}$ is a tight frame with $A = B = N\|g\|_2^2$.

In Table 1, we compare the lower and upper frame bounds derived in Theorem 3 to the empirical optimal frame bounds, A and B , for different graphs.

Graph	μ	τ	$N \hat{g}(0) ^2$	A	B	$N^2\mu^2\ g\ _2^2$
Path graph	0.063	0.5	3.2	498.4	846.2	1000.0
		5.0	11.0	494.5	976.5	1000.0
		50.0	34.2	482.9	964.6	1000.0
Random regular graph	0.225	0.5	150.8	465.0	591.8	12676.9
		5.0	500.0	500.0	500.0	12676.9
		50.0	500.0	500.0	500.0	12676.9
Random sensor network	0.956	0.5	13.5	142.7	3702.2	228607.0
		5.0	71.7	158.4	2530.7	228607.0
		50.0	387.1	389.3	1185.6	228607.0
Comet graph	0.998	0.5	3.0	7.4	786.3	248756.2
		5.0	17.9	43.9	1584.1	248756.2
		50.0	52.9	119.1	1490.8	248756.2

Table 1: Comparison of the empirical optimal frame bounds, A and B , to the theoretical lower and upper bounds derived in Theorem 3. The number of vertices for all graphs is $N = 500$, and in all cases, the window is $\hat{g}(\lambda_\ell) = Ce^{-\tau\lambda_\ell}$, where the constant C is chosen such that $\|g\|_2 = 1$. In the random regular graph, every vertex has degree 8. The random sensor network is the one shown in Figure 8(b). In the comet graph, which is structured like the one shown in Figure 8(e), the degree of the center vertex is 200.

6.3. Reconstruction Formula

Provided the window g has a non-zero mean, a signal $f \in \mathbb{R}^N$ can be recovered from its windowed graph Fourier transform coefficients.

Theorem 4: If $\hat{g}(0) \neq 0$, then for any $f \in \mathbb{R}^N$,

$$f(n) = \frac{1}{N\|T_n g\|_2^2} \sum_{i=1}^N \sum_{k=0}^{N-1} S f(i, k) g_{i,k}(n).$$

Proof.

$$\begin{aligned} & \sum_{i=1}^N \sum_{k=0}^{N-1} S f(i, k) g_{i,k}(n) \\ &= \sum_{i=1}^N \sum_{k=0}^{N-1} \left(N \sum_{m=1}^N f(m) \chi_k^*(m) \sum_{\ell=0}^{N-1} \hat{g}(\lambda_\ell) \chi_\ell(i) \chi_\ell^*(m) \right) \left(N \chi_k(n) \sum_{\ell'=0}^{N-1} \hat{g}(\lambda_{\ell'}) \chi_{\ell'}^*(i) \chi_{\ell'}(n) \right) \\ &= N^2 \sum_{m=1}^N f(m) \sum_{\ell=0}^{N-1} \sum_{\ell'=0}^{N-1} \hat{g}(\lambda_\ell) \hat{g}(\lambda_{\ell'}) \chi_\ell^*(m) \chi_{\ell'}(n) \sum_{i=1}^N \chi_\ell(i) \chi_{\ell'}^*(i) \sum_{k=0}^{N-1} \chi_k^*(m) \chi_k(n) \\ &= N^2 \sum_{m=1}^N f(m) \sum_{\ell=0}^{N-1} \sum_{\ell'=0}^{N-1} \hat{g}(\lambda_\ell) \hat{g}(\lambda_{\ell'}) \chi_\ell^*(m) \chi_{\ell'}(n) \delta_{\ell\ell'} \delta_{mn} = N^2 f(n) \sum_{\ell=0}^{N-1} |\hat{g}(\lambda_\ell)|^2 |\chi_\ell^*(n)|^2 = N\|T_n g\|_2^2 f(n), \end{aligned}$$

where the last equality follows from (28). \square

Remark 3: In the classical case, $\|T_n g\|_2^2 = \|g\|_2^2 = 1$, so this term does not appear in the reconstruction formula (see, e.g., [7, Theorem 4.3]).

6.4. Spectrogram Examples

As in classical time-frequency analysis, we can now examine $|Sf(i, k)|^2$, the squared magnitudes of the windowed graph Fourier transform coefficients of a given signal f . In the classical case (see, e.g., [7, Theorems 4.1 and 4.3]), the windowed Fourier atoms form a tight frame, and therefore this *spectrogram* of squared magnitudes can be viewed as an energy density function of the signal across the time-frequency plane. In the graph setting, the windowed graph Fourier atoms do not always form a tight frame, and we cannot therefore, in general, interpret the graph spectrogram as an energy density function. Nonetheless, it can still be a useful tool to elucidate underlying structure in graph signals. In this section, we present some examples to illustrate this concept and provide further intuition behind the proposed windowed graph Fourier transform.

Example 4: We consider a path graph of 180 vertices, with all the weights equal to one. The graph Laplacian eigenvectors, given in (8), are the basis vectors in the DCT-II transform. We compose the signal shown in Figure 13(a) on the path graph by summing three signals: χ_{10} restricted to the first 60 vertices, χ_{60} restricted to the next 60 vertices, and χ_{30} restricted to the final 60 vertices. We design a window g by setting $\hat{g}(\lambda_\ell) = Ce^{-\tau\lambda_\ell}$ with $\tau = 300$ and C chosen such that $\|g\|_2 = 1$. The “spectrogram” in Figure 13(b) shows $|Sf(i, k)|^2$ for all $i \in \{1, 2, \dots, 180\}$ and $k \in \{0, 1, \dots, 179\}$. Consistent with intuition from discrete-time signal processing (see, e.g., [6, Chapter 2.1]), the spectrogram shows the discrete cosines at different frequencies with the appropriate spatial localization.

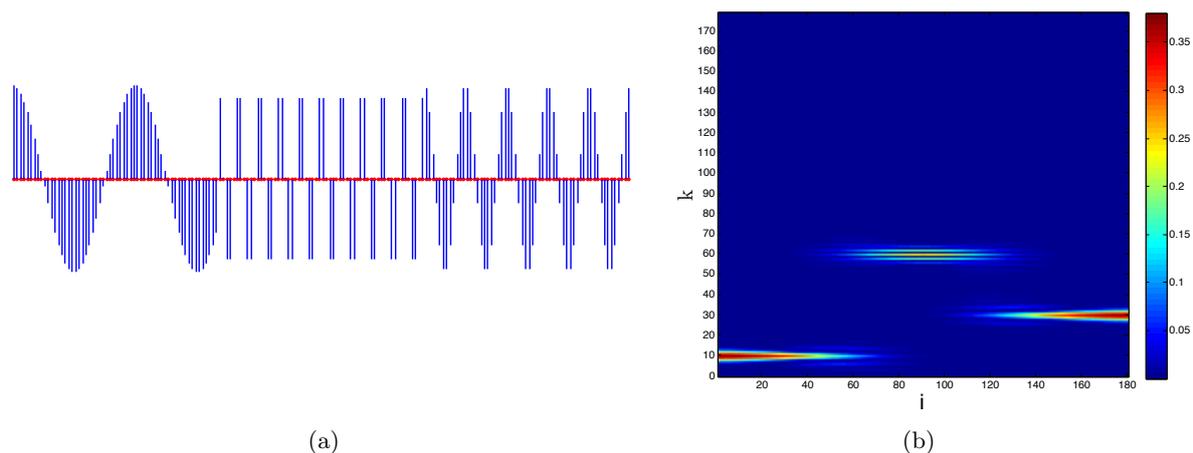


Figure 13: Spectrogram example on a path graph. (a) A signal f on the path graph that is comprised of three different graph Laplacian eigenvectors restricted to three different segments of the graph. (b) A spectrogram of f . The vertex indices are on the horizontal axis, and the frequency indices are on the vertical axis.

Example 5: We now compute the spectrogram of the signal f on the random sensor network of Example 3, using the same window $\hat{g}(\lambda_\ell) = Ce^{-\tau\lambda_\ell}$ from Example 3, with $\tau = 3$. It is not immediately obvious upon inspection that the signal f shown in both Figure 11(a) and Figure 14(a) is a highly structured signal; however, the spectrogram, shown in Figure 14(b), elucidates the structure of f , as we can clearly see three different frequency components present in three different areas of the graph. In order to mimic Example 4 on a more general graph, we constructed the signal f by first using spectral clustering (see, e.g., [10]) to partition the network into three sets of vertices, which are shown in red, blue, and green in Figure 14(a). Like Example 4, we then took the signal f to be the sum of three signals: χ_{10} restricted to the red set of vertices, χ_{27} restricted to the blue set of vertices, and χ_5 restricted to the green set of vertices.

Remark 4: In Figure 14(b), in order to make the structure of the signal more evident, we arranged the vertices of the sensor graph on the horizontal axis according to clusters, so that vertices in the same cluster

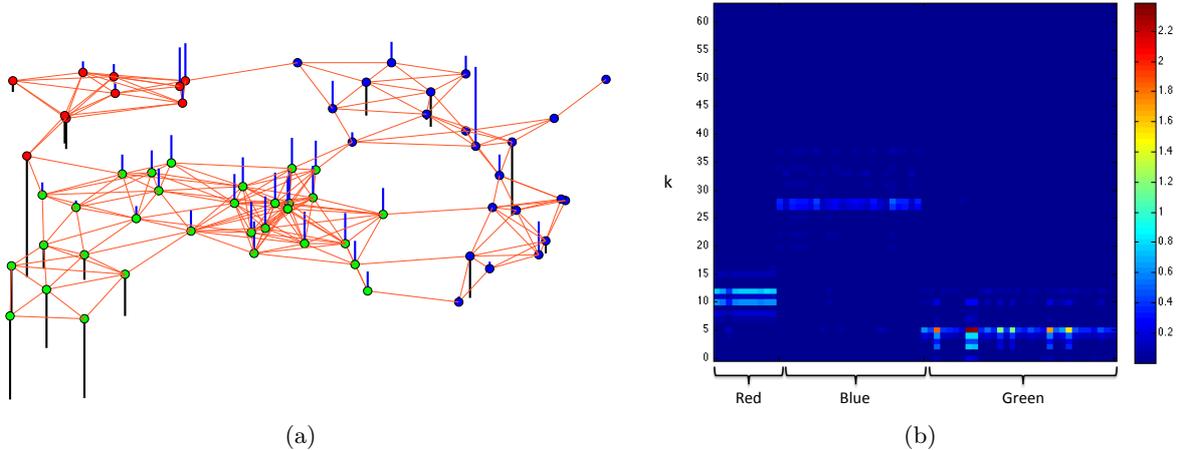


Figure 14: Spectrogram example on a random sensor network. (a) A signal comprised of three different graph Laplacian eigenvectors restricted to three different clusters of a random sensor network. (b) The spectrogram shows the different frequency components in the red, blue, and green clusters.

are close to each other. Of course, this would not be possible if we did not have an idea of the structure a priori. A more general way to view the spectrogram of a graph signal without such a priori knowledge is as a sequence of images, with one image per graph Laplacian eigenvalue and the sequence arranged monotonically according to the corresponding eigenvalue. Then, as we scroll through the sequence (i.e., play a frequency-lapse video), the areas of the graph where the signal contains low frequency components “light up” first, then the areas where the signal contains middle frequency components, and so forth. In Figure 15, we show a subset of the images that would comprise such a sequence for the signal from Example 5.

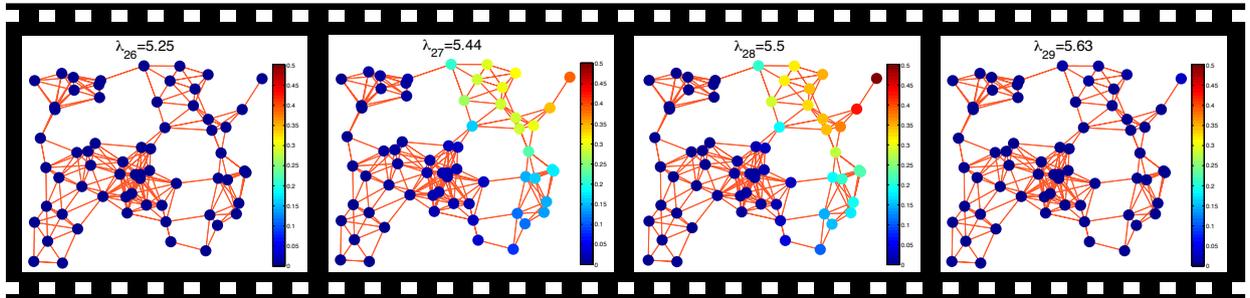


Figure 15: Frequency-lapse video representation of the spectrogram of a graph signal. The graph spectrogram can be viewed as a sequence of images, with each frame corresponding to the spectrogram coefficients at all vertices and a single frequency, $|Sf(:, k)|^2$. This particular subsequence of frames shows the spectrogram coefficients from eigenvalues λ_{26} through λ_{29} . At frequencies close to $\lambda_{27} = 5.44$, we can see the coefficients “light up” in the blue cluster of vertices from Figure 14(a), corresponding to the spectrogram coefficients in the middle of Figure 14(b).

6.5. Application Example: Signal-Adapted Graph Clustering

Motivated by Examples 4 and 5, we can also use the windowed graph Fourier transform coefficients as feature vectors to cluster a graph into sets of vertices, taking into account both the graph structure and a given signal f . In particular, for $i = 1, 2, \dots, N$, we can define $y_i := Sf(i, :) \in \mathbb{R}^N$, and then use a standard clustering algorithm to cluster the points $\{y_i\}_{i=1,2,\dots,N}$.⁵

⁵Note that if we take the window to be a heat kernel $\hat{g}(\lambda) = Ce^{-\tau\lambda}$, as $\tau \rightarrow 0$, $T_i g \rightarrow \delta_i$ and $Sf(i, k) \rightarrow \langle \sqrt{N}M_k \delta_i, f \rangle = \sqrt{N}\chi_k(i)f(i)$. Thus, this method of clustering reduces to spectral clustering [10] when (i) the signal f is constant across all

Example 6: We generate a signal f on the 500 vertex random sensor network of Example 1 as follows. First, we generate four random signals $\{f_i\}_{i=1,2,3,4}$, with each random component uniformly distributed between 0 and 1. Second, we generate four graph spectral filters $\{\widehat{h}_i(\cdot)\}_{i=1,2,3,4}$ that cover different bands of the graph Laplacian spectrum, as shown in Figure 16(a). Third, we generate four clusters, $\{C_i\}_{i=1,2,3,4}$, on the graph, taking the first three to be balls of radius 4 around different center vertices, and the fourth to be the remaining vertices. These clusters are shown in Figure 16(b). Fourth, we generate a signal on the graph as

$$f = \sum_{i=1}^4 (f_i * h_i)|_{C_i};$$

i.e., we filter each random signal by the corresponding filter (to shift its frequency content to a given band), and then restrict that signal to the given cluster by setting the components outside of that cluster to 0. The resulting signal f is shown in Figure 16(c). Fifth, we compute the windowed graph Fourier transform coefficients of f using a window $\widehat{g}(\lambda) = e^{-0.3\lambda}$. Sixth, we use a classical trick of applying a nonlinear transformation to the coefficients (see, e.g., [33, Section 3]), and define the vectors $\{y_i\}_{i=1,2,\dots,N}$ as

$$y_i(k) := \tanh\left(\alpha|Sf(i,k)|\right) = \frac{1 - e^{-2\alpha|Sf(i,k)|}}{1 + e^{-2\alpha|Sf(i,k)|}},$$

with $\alpha = 0.75$. Finally, we perform k -means clustering on the points $\{y_i\}_{i=1,2,\dots,N}$, searching for 6 clusters to give the algorithm some extra flexibility. The resulting signal-adapted graph clustering is shown in Figure 16(d).

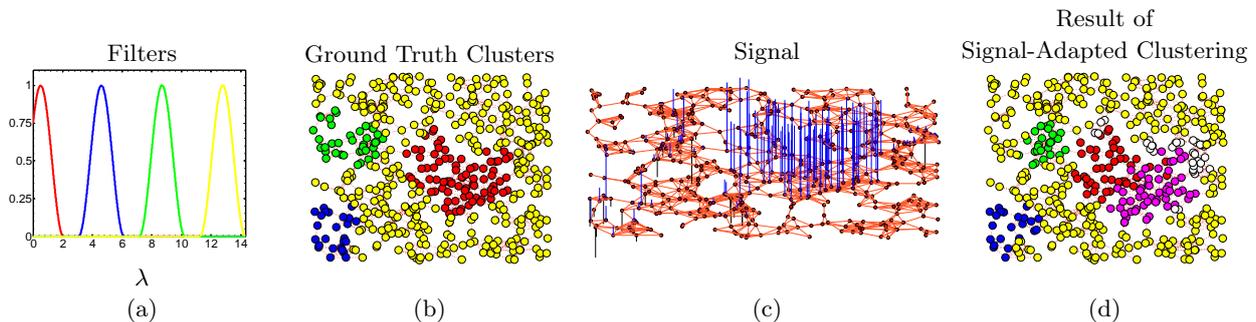


Figure 16: Signal-adapted graph clustering example. (a) The graph spectral filters $\{\widehat{h}_i(\cdot)\}_{i=1,2,3,4}$ covering different bands of the graph Laplacian spectrum. (b) The four clusters used to generate the signal f . (c) The signal f is a combination of signals in different frequency bands ($f_i * h_i$) restricted to different areas of the graph. We can see for example that the red cluster is generated by the lowpass red filter in (a), and therefore the signal f is relatively smooth around the red cluster of vertices. (d) The signal-adapted graph clustering results. We see that the red and magenta clusters roughly correspond to the original red cluster, and the white and yellow clusters roughly correspond to the original yellow cluster. There are some errors around the edges of the original red, blue, and green clusters, particularly in regions of low connectivity.

6.6. Tiling

Thus far, we have generalized the classical notions of translation and modulation in order to mimic the construction of the classical windowed Fourier transform. We have seen from the examples in the previous subsections that the spectrogram may be an informative tool for signals on both regular and irregular graphs. In this section and the following section, we examine the extent to which our intuitions from classical time-frequency analysis carry over to the graph setting, and where they deviate due to the irregularity of the data domain, and, in turn, the possibility of localized Laplacian eigenfunctions.

vertices, and (ii) each window is a delta in the vertex domain.

First, we compare tilings of the time-frequency plane (or, in the graph setting, the vertex-frequency plane). Recall that Heisenberg boxes represent the time-frequency resolution of a given dictionary atom (including, e.g., windowed Fourier atoms or wavelets) in the time-frequency plane (see, e.g., [7, Chapter 4]). As shown in the tiling diagrams of Figure 17(a) and 17(b), respectively, the Heisenberg boxes of classical windowed Fourier atoms have the same size throughout the time-frequency plane, while the Heisenberg boxes of classical wavelets have different sizes at different wavelet scales. While one can trade-off time and frequency resolutions (e.g., change the length and width of the Heisenberg boxes of classical windowed Fourier atoms by changing the shape of the analysis window), the Heisenberg uncertainty principle places a lower limit on the area of each Heisenberg box.

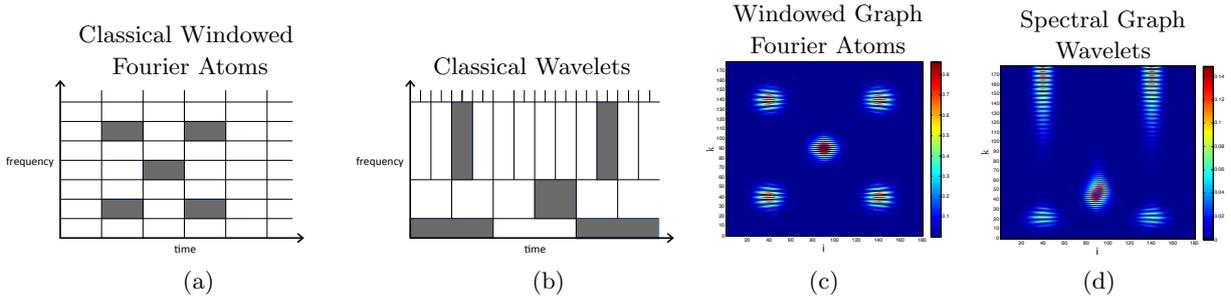


Figure 17: (a) Tiling of the time-frequency plane by classical windowed Fourier atoms. (b) Tiling of the time-frequency plane by classical wavelets. (c) Sum of the spectrograms of five windowed graph Fourier atoms on the path graph with 180 vertices. (d) Sum of the spectrograms of five spectral graph wavelets on the path graph with 180 vertices.

In Figure 17(c) and 17(d), we use the windowed graph Fourier transform to show the sums of spectrograms of five windowed graph Fourier atoms on a path graph and five spectral graph wavelets on a path graph, respectively. The plots are not too different from what intuition from classical time-frequency analysis might suggest. Namely, the sizes of the Heisenberg boxes for different windowed graph Fourier atoms are roughly the same, while the sizes of the Heisenberg boxes of spectral graph wavelets are similar at a fixed scale, but vary across scales.

In Figure 18, we plot three different windowed graph Fourier atoms – all with the same center vertex – on the Swiss roll graph. Note that all three atoms are jointly localized in the vertex domain around the center vertex 62, and in the graph spectral domain around the frequencies to which they have been respectively modulated. However, unlike the path graph example in Figure 17, the sizes of the Heisenberg boxes of the three atoms are quite different. In particular, the atom $g_{62,983}$ is extremely close to a delta function in both the vertex domain and the graph spectral domain, which of course is not possible in the classical setting due to the Heisenberg uncertainty principle. The reason this happens is that the coherence of this Swiss roll graph is $\mu = 0.94$, and the eigenvector χ_{983} is highly localized, with a value of -0.94 at vertex 62. The takeaway is that highly localized eigenvectors can limit the extent to which intuition from the classical setting carries over to the graph setting.

6.7. Limitations

In this section, we briefly discuss a few limitations of the proposed windowed graph Fourier transform.

6.7.1. Computational Complexity

While the exact computation of the windowed graph Fourier transform coefficients via (52) and (53) is feasible for smaller graphs (e.g., less than 10,000 vertices), the computational cost may be prohibitive for much larger graphs. Therefore, it would be of interest to develop an approximate computational method that scales more efficiently with the size of the graph. Recall that

$$Sf(i, k) = \langle f, g_{ik} \rangle = \langle f, M_k T_i g \rangle = \langle f (T_i g)^*, \chi_k \rangle = \widehat{f (T_i g)^*}(\lambda_k). \quad (59)$$

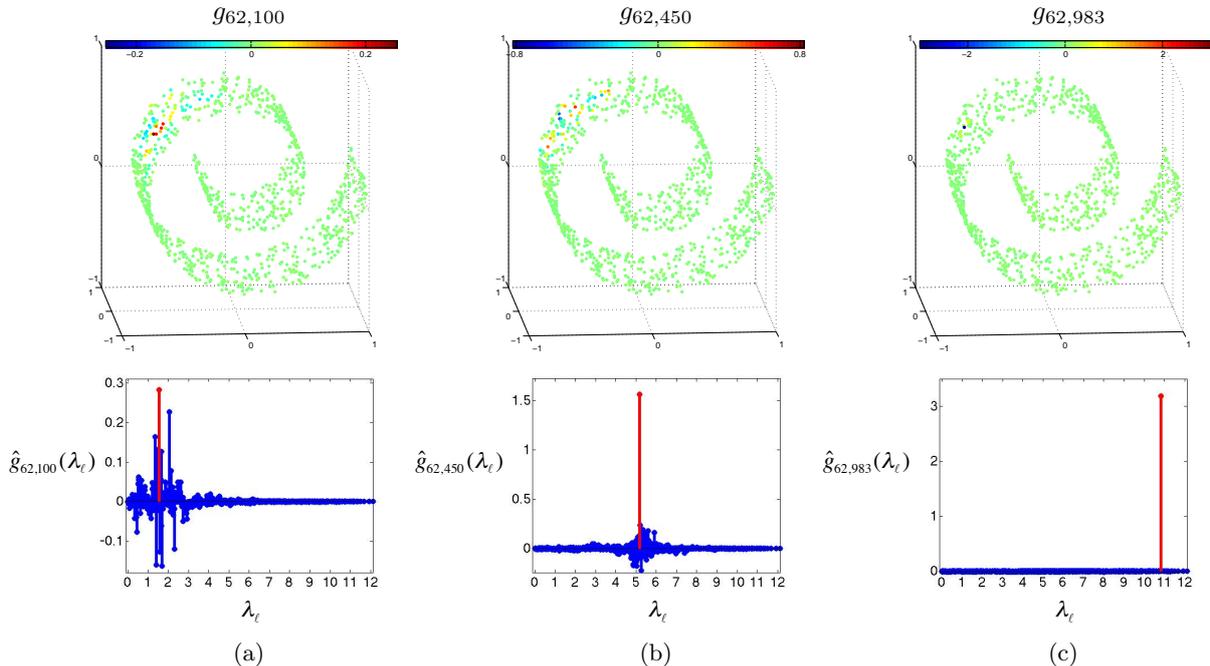


Figure 18: Three different windowed graph Fourier atoms on the Swiss roll, shown in both domains.

The quantity $T_i g$ in the last term of (59) can be approximately computed in an efficient manner via the Chebyshev polynomial method of [22, Section 6], and therefore $f(T_i g)^*$ can be approximately computed in an efficient manner. Thus, if there was a fast approximate graph Fourier transform, we could apply that to the fast approximation of $f(T_i g)^*$ in order to approximately compute the windowed graph Fourier transform coefficients $\{Sf(i, k)\}_{k=0,1,\dots,N-1}$. Unfortunately, we are not yet aware of a good fast approximate graph Fourier transform method.

6.7.2. Lack of a Tight Frame

As discussed in Sections 6.2 and 6.4, the collection of windowed graph Fourier atoms need not form a tight frame, meaning that the spectrogram can not always be interpreted as an energy density function. Furthermore, the lack of a tight frame may lead to (i) less numerical stability when reconstructing a signal from (potentially noisy) windowed graph Fourier transform coefficients [30, 31, 32], or (ii) slower computations, for example when computing proximity operators in convex regularization problems [34].

6.7.3. No Guarantees on the Joint Localization of the Atoms in the Vertex and Graph Spectral Domains

Thus far, we have seen that (i) if we translate a smooth kernel \hat{g} to vertex i , the resulting signal $T_i g$ will be localized around vertex i in the vertex domain (Section 4.4); (ii) if we modulate a kernel \hat{g} that is localized around 0 in the graph spectral domain, the resulting kernel $\widehat{M_k g}$ will be localized around λ_k in the graph spectral domain (Section 5); and (iii) a windowed graph Fourier atom, g_{ik} , is often jointly localized around vertex i in the vertex domain and frequency λ_k in the graph spectral domain (e.g., Figure 18). In classical time-frequency analysis, the windowed Fourier atoms $g_{u,\xi}$ are all jointly localized around time u and frequency ξ . So we now ask whether the windowed graph Fourier atoms are always jointly localized around vertex i and frequency λ_k ? The answer is no, and the reason once again follows from the possibility of localized graph Laplacian eigenvectors. For a smooth window \hat{g} , the translated window $T_i g$ is indeed localized around vertex i ; however, $g_{ik} = \sqrt{N} \chi_k \cdot (T_i g)$ may not be localized around vertex i when $\chi_k(n)$ is close to zero for all vertices n in a neighborhood around i . One such example is shown in Figure 19. Similarly, in order for \hat{g}_{ik} to be localized around frequency λ_k in the graph spectral domain, it suffices for

$\widehat{T}_i g$ to be localized around 0 in the graph spectral domain. However,

$$\widehat{T}_i g(\lambda_\ell) = \sqrt{N} \hat{g}(\lambda_\ell) \chi_\ell^*(i),$$

and, therefore, it is possible that the multiplication by a graph Laplacian eigenvector changes the localization of the translated window in the graph spectral domain. In classical time-frequency analysis, these phenomena never occur, because the complex exponentials are always delocalized.

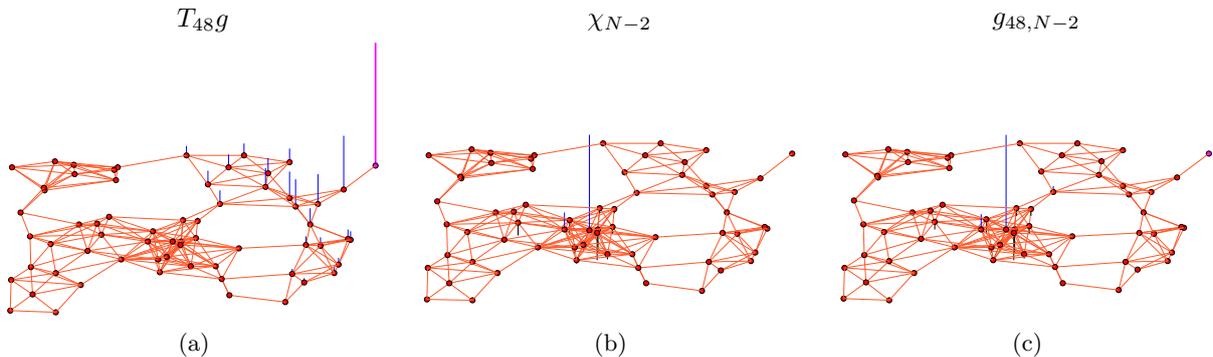


Figure 19: A windowed graph Fourier atom that is not localized around its center vertex. (a) The translated window $T_{48}g$ is localized around the center vertex 48, which is shown in magenta. (b) The graph Laplacian eigenvector χ_{N-2} is highly localized around a different vertex in the graph and is extremely close to zero in the neighborhood of vertex 48. (c) Therefore, when we modulate the translated window $T_{48}g$ by χ_{N-2} , the resulting atom $g_{48, N-2}$ is not centered around vertex 48. Note that the three figures are drawn on different scales; the magnitudes of the largest components of the signals are 1.37, 0.59, and 0.15, respectively.

Despite the lack of guarantees on joint localization, the windowed graph Fourier transform is still a useful analysis tool. First, if the coherence μ is low (close to $\frac{1}{\sqrt{N}}$), the graph Laplacian eigenvectors are delocalized, the atoms are jointly localized in the vertex and graph spectral domains, and much of the intuition from classical time-frequency analysis carries over to the graph setting. Even when the coherence is close to 1, however, it often happens that the majority of the atoms are in fact jointly localized in time and frequency. This is because only those atoms whose computations include highly localized eigenvectors are affected. We have observed empirically that there tends to be only a few graph Laplacian eigenvectors, most commonly those associated with the higher frequencies (eigenvalues close to λ_{\max}). Moreover, if χ_k is highly localized around a vertex j , then $Sf(i, k)$ will be close to 0 for any vertex i not close to j , so it is not particularly problematic that g_{ik} may not be localized around i in the vertex domain.

7. Conclusion and Future Work

We defined generalized notions of translation and modulation through multiplication with a graph Laplacian eigenvector in the graph spectral and vertex domains, respectively. We leveraged these generalized operators to design a windowed graph Fourier transform, which enables vertex-frequency analysis for signals on graphs. We showed that when the chosen window is smooth in the graph spectral domain, translated windows are localized in the vertex domain. Moreover, when the chosen window is localized around zero in the graph spectral domain, the modulation operator is close to a translation in the graph spectral domain. If we apply this windowed graph Fourier transform to a signal with frequency components that vary along a path graph, the resulting spectrogram matches our intuition from classical discrete-time signal processing. Yet, our construction is fully generalized and can be applied to analyze signals on any undirected, connected, weighted graph. The example in Figure 14 shows that the windowed graph Fourier transform may be a valuable tool for extracting information from signals on graphs, as structural properties of the data that are hidden in the vertex domain may become obvious in the transform domain.

One line of future work is to continue to improve the localization results for the translated kernels presented in Section 4.4, preferably by incorporating the graph weights. This issue is closely related to the

study of both the localization of eigenvectors and recent work in the theory of matrix functions [23]. In particular, it is related to the off-diagonal decay of entries of a matrix function $g(\mathcal{L})$, as studied in [35, 36]. To our knowledge, existing results in this area also do not incorporate the entries of the matrix \mathcal{L} (other than through the eigenvalues), but rather depend primarily on the sparsity pattern of \mathcal{L} . For more precise numerical localization results, numerical linear algebra researchers have also turned to quadrature methods to approximate the quantity $\delta_i^* g(\mathcal{L}) \delta_j$ (see, e.g., [37] and references therein).

Motivated by the spirit of vertex-frequency analysis introduced in this paper, we are also investigating a new, more computationally efficient dictionary design method to generate tight frames of atoms that are jointly localized in the vertex and graph spectral domains.

8. Appendix

8.1. The Normalized Laplacian Graph Fourier Basis Case

We now briefly consider the case when the normalized graph Laplacian eigenvectors are used as the Fourier basis, and revisit the definitions and properties of the generalized translation and modulation operators from Sections 4 and 5. Throughout we use $\tilde{\cdot}$ to denote the corresponding quantities derived from the normalized graph Laplacian $\tilde{\mathcal{L}}$. To prove many of the following properties, we use the fact that for connected graphs

$$\tilde{\chi}_0(n) = \frac{\sqrt{d_n}}{\|\sqrt{d}\|_2} = \frac{\sqrt{N}\sqrt{d_n}}{\|\sqrt{d}\|_2} \chi_0(n),$$

where the square root in \sqrt{d} is applied component-wise.

8.1.1. Generalized Convolution and Translation in the Normalized Laplacian Graph Fourier Basis

We can keep the definition (16) of the generalized convolution, with the normalized graph Laplacian eigenvalues and eigenvectors replacing those of the combinatorial graph Laplacian. Statements 1-7 in Proposition 1 are still valid; however, (24) becomes

$$\sum_{n=1}^N (f * g)(n) \sqrt{d_n} = \|\sqrt{d}\|_2 \hat{f}(0) \hat{g}(0) = \frac{1}{\|\sqrt{d}\|_2} \left[\sum_{n=1}^N f(n) \sqrt{d_n} \right] \left[\sum_{n=1}^N g(n) \sqrt{d_n} \right].$$

Accordingly, we can redefine the generalized translation operator as

$$(\tilde{T}_i f)(n) := \|\sqrt{d}\|_2 (f * \delta_i)(n) = \|\sqrt{d}\|_2 \sum_{\ell=0}^{N-1} \hat{f}(\tilde{\lambda}_\ell) \tilde{\chi}_\ell^*(i) \tilde{\chi}_\ell(n),$$

so that Property 3 of Corollary 1 becomes

$$\sum_{n=1}^N (\tilde{T}_i f)(n) \sqrt{d_n} = \sqrt{d_i} \sum_{n=1}^N f(n) \sqrt{d_n},$$

and for any $f \in \mathbb{R}^N$, Lemma 1 becomes

$$\sqrt{d_i} |\hat{f}(0)| \leq \|\tilde{T}_i f\|_2 \leq \tilde{\nu}_i \|\sqrt{d}\|_2 \|f\|_2 \leq \tilde{\mu} \|\sqrt{d}\|_2 \|f\|_2.$$

Because they only depend on the graph structure and not the specific graph weights, the localization results of Theorem 1, Corollary 2, and Corollary 3 also hold, with the constant \sqrt{N} replaced by $\|\sqrt{d}\|_2$ and an extra factor of $\sqrt{d_i}$ in the denominator of (37) and (38).

8.1.2. Generalized Modulation in the Normalized Laplacian Graph Fourier Basis

When we use the normalized Laplacian eigenvectors as a graph Fourier basis instead of the combinatorial Laplacian eigenvectors, we can define a generalized modulation as

$$(\tilde{M}_k f)(n) := f(n) \frac{\tilde{\chi}_k(n)}{\tilde{\chi}_0(n)} = f(n) \frac{\tilde{\chi}_k(n) \|\sqrt{d}\|_2}{\sqrt{d_n}},$$

so that \tilde{M}_0 is also the identity operator and $\widehat{\tilde{M}_k \delta_0}(\tilde{\lambda}_\ell) = \delta_0(\tilde{\lambda}_\ell - \tilde{\lambda}_k)$; i.e., \tilde{M}_k maps the graph spectral component $\hat{f}(0)$ from eigenvalue $\tilde{\lambda}_0$ to eigenvalue $\tilde{\lambda}_k$ in the graph spectral domain. The following theorem on the localization of a modulated kernel in the graph spectral domain follows essentially the same line of argument as Theorem 2, with $\frac{1}{\tilde{\chi}_0(n)}$ replacing \sqrt{N} in the proof.

Theorem 5: *Given a weighted graph \mathcal{G} with N vertices, if for some $\gamma > 0$, a kernel \hat{f} satisfies*

$$\sum_{\ell=1}^{N-1} \tilde{\mu}_\ell \left| \hat{f}(\tilde{\lambda}_\ell) \right| \leq \frac{\sqrt{d_{\min}} |\hat{f}(0)|}{\|\sqrt{d}\|_2 (1 + \gamma)}, \quad (60)$$

then

$$\left| \widehat{\tilde{M}_k f}(\tilde{\lambda}_k) \right| \geq \gamma \left| \widehat{\tilde{M}_k f}(\tilde{\lambda}_\ell) \right| \quad \text{for all } \ell \neq k. \quad (61)$$

8.1.3. Example: Resolution Tradeoff in the Normalized Laplacian Graph Fourier Basis

We once again consider a heat kernel of the form $\hat{g}(\tilde{\lambda}) = e^{-\tau \tilde{\lambda}}$. First, for the spread of a translated kernel in the vertex domain, we can replace the upper bound (42) by

$$\Delta_i^2(\tilde{T}_i g) \leq \|\sqrt{d}\|_2^2 \tau^2 \exp\left(\frac{\tau^2}{4(d_{\max} - 1)}\right). \quad (62)$$

Next, our localization result on the generalized modulation, Theorem 5, tells us that

$$\sum_{\ell=1}^{N-1} e^{-\tau \tilde{\lambda}_\ell} \leq \frac{\sqrt{d_{\min}}}{\tilde{\mu} \|\sqrt{d}\|_2 (1 + \gamma)} \quad (63)$$

for some $\gamma > 0$ implies $|\widehat{\tilde{M}_k g}(\tilde{\lambda}_k)| \geq \gamma |\widehat{\tilde{M}_k g}(\tilde{\lambda}_\ell)|$ for all ℓ not equal to k . For a graph \mathcal{G} with known isoperimetric dimension (see, e.g., [38], [8, Chapter 11]), the following result upper bounds the left-hand side of (63).

Theorem 6 (Chung and Yau, [38, Theorem 7]): *The normalized graph Laplacian eigenvalues of a graph \mathcal{G} satisfy*

$$\sum_{\ell=1}^{N-1} e^{-\tau \tilde{\lambda}_\ell} \leq \frac{C_\delta \|d\|_1}{\tau^{\frac{\delta}{2}}}, \quad (64)$$

where for every subset $\mathcal{V}_1 \subset \mathcal{V}$, δ , the isoperimetric dimension of \mathcal{G} , and c_δ , the isoperimetric constant, satisfy

$$|\mathcal{E}(\mathcal{V}_1, \mathcal{V}_1^c)| \geq c_\delta \left[\sum_{n \in \mathcal{V}_1} d_n \right]^{\frac{\delta-1}{\delta}},$$

and C_δ is a constant that only depends on δ .

Combining (63) and (64), for a fixed τ ,

$$\left| \widehat{M_k g}(\tilde{\lambda}_k) \right| \geq \left(\frac{\sqrt{d_{\min}} \tau^{\frac{\delta}{2}}}{\tilde{\mu} C_\delta \|\sqrt{d}\|_2 \|d\|_1} - 1 \right) \left| \widehat{M_k g}(\tilde{\lambda}_\ell) \right|, \quad \forall \ell \neq k. \quad (65)$$

Similarly, to ensure $\left| \widehat{M_k g}(\tilde{\lambda}_k) \right| \geq \gamma \left| \widehat{M_k g}(\tilde{\lambda}_\ell) \right|$, $\forall \ell \neq k$ for a desired $\gamma > 0$, it suffices to choose the diffusion parameter of the heat kernel as

$$\tau \geq \left(\frac{\tilde{\mu} C_\delta \|\sqrt{d}\|_2 \|d\|_1 (1 + \gamma)}{\sqrt{d_{\min}}} \right)^{\frac{2}{\delta}}. \quad (66)$$

Comparing (65) and (66) to (62), we see that as the diffusion parameter τ increases, our guarantee on the localization of $\widehat{M_k g}$ around frequency λ_k in the graph spectral domain improves, whereas the guarantee on the localization of $T_i g$ around vertex i in the vertex domain becomes weaker, and vice versa.

8.2. Alternative Definition of Generalized Modulation

The classical modulation operator corresponds to translation in the spectral domain. Therefore, a second approach to generalize modulation to the graph setting is to define a new weighted graph $\tilde{\mathcal{G}}$ on the graph Laplacian spectrum and then define the generalized modulation on \mathcal{G} as a generalized translation on $\tilde{\mathcal{G}}$. More specifically, we first define a new graph $\tilde{\mathcal{G}}$, whose vertices are the eigenvalues of the original graph Laplacian \mathcal{L} . One simple choice of graphs is a weighted path graph, where each eigenvalue λ_ℓ from the original graph \mathcal{G} is connected only to $\lambda_{\ell-1}$ and $\lambda_{\ell+1}$, and the weights are inversely proportional to the distances between neighboring eigenvalues. Other possibilities include assigning exponential weights based on the distance between eigenvalues, or connecting each eigenvalue to multiple neighbors on each side and assigning weights according to a thresholded weighting function like (10). Next, we form the Laplacian $\tilde{\mathcal{L}}$ on this new graph $\tilde{\mathcal{G}}$, and denote its eigenpairs by $(\tilde{\lambda}_j, \tilde{\chi}_j)$. Then generalized modulation on \mathcal{G} is defined as generalized translation on $\tilde{\mathcal{G}}$:

$$\left(\widehat{M_k f} \right) (\lambda_\ell) := \left(\tilde{T}_k \hat{f} \right) (\ell) = \sqrt{N} \hat{f}(\tilde{\mathcal{L}})_{k,\ell} = \sqrt{N} \sum_{j=0}^{N-1} \hat{f}(\tilde{\lambda}_j) \tilde{\chi}_j^*(k) \tilde{\chi}_j(\ell), \quad (67)$$

where we have indexed the vertices in $\tilde{\mathcal{G}}$ as $\ell = 0, 1, \dots, N-1$, rather than the usual 1 to N . In (67), f lives on the vertices of the original graph \mathcal{G} , \hat{f} lives on both the spectrum $\sigma(\mathcal{L})$ of the original graph and the vertices of $\tilde{\mathcal{G}}$, and \hat{f} (a graph Fourier transform of f with respect to the Laplacian eigenvectors of \mathcal{L} followed by a graph Fourier transform of \hat{f} with respect to the Laplacian eigenvectors of $\tilde{\mathcal{L}}$) lives on the spectrum $\sigma(\tilde{\mathcal{L}})$ of $\tilde{\mathcal{G}}$. Taking an inverse graph Fourier transform of (67) (with respect to the eigenvectors of the original graph Laplacian \mathcal{L}) yields

$$(M_k f)(n) = \sqrt{N} \sum_{\ell=0}^{N-1} \sum_{j=0}^{N-1} \hat{f}(\tilde{\lambda}_j) \tilde{\chi}_j^*(k) \tilde{\chi}_j(\ell) \chi_\ell^*(n).$$

We can define the kernel f in (67) in one of two ways. First, to maintain direct comparability to the generalized modulation, we can define the kernel $\hat{f}(\lambda_\ell)$ directly on $\sigma(\mathcal{L})$. For example, in Figure 20, we let $\hat{f}(\lambda_\ell) = C e^{\tau \lambda_\ell}$ (with C chosen such that $\|\hat{f}\|_2 = 1$), and then use the definition (67) to modulate a kernel on the Laplacian spectrum of the Minnesota graph.

A second option is to define f on the spectrum $\sigma(\tilde{\mathcal{L}})$. For example, in Figure 21, we let $\hat{f}(\tilde{\lambda}_j) = e^{\tilde{\tau} \tilde{\lambda}_j}$, and then modulate again via (67). One advantage of defining the kernel in this manner is that we can more easily characterize the spread of the modulated kernel in the graph spectral domain of \mathcal{G} , using, e.g., our translation localization results from Section 4.4. For example, with $\hat{f}(\cdot)$ a heat kernel and $\tilde{\mathcal{G}}$ a weighted

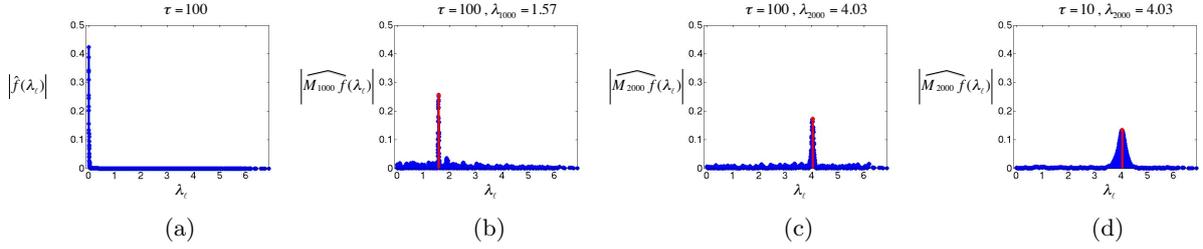


Figure 20: Generalized modulation on the Minnesota graph \mathcal{G} , via generalized translation on $\tilde{\mathcal{G}}$ with the same kernel $\hat{f}(\lambda_\ell) = C e^{-\tau \lambda_\ell}$ defined on $\sigma(\mathcal{L})$ as in Figure 10. The graph $\tilde{\mathcal{G}}$ is a weighted path graph with $\tilde{W}_{kl} = \frac{1}{|\lambda_k - \lambda_\ell| + 10^{-6}}$. (a) The normalized heat kernel $\hat{f}(\lambda_\ell)$. (b) and (c) Modulated kernels with two different center frequencies, λ_{1000} and λ_{2000} . In both cases, the modulated kernel is localized around the desired frequency of $\sigma(\mathcal{L})$ (shown in red). (d) As τ decreases, the spread of the modulated kernel around the center frequency in the graph spectral domain of \mathcal{G} generally increases.

path graph (which has a maximum Laplacian eigenvalue upper bounded by 4) as in Figure 21, the bound (42) becomes

$$\frac{\sum_{\ell=0}^{N-1} (\ell - k)^2 \left| \widehat{M_k f}(\lambda_\ell) \right|^2}{\|M_k f\|_2^2} = \Delta_k^2(\tilde{T}_\ell f) \leq 8N\tilde{\tau}^2 e^{\tilde{\tau}^2},$$

which is close to form of the desired bound on the spread of modulated kernels we mentioned in (51), especially for a graph \mathcal{G} whose spectrum is close to uniformly distributed on $[0, \lambda_{\max}]$.

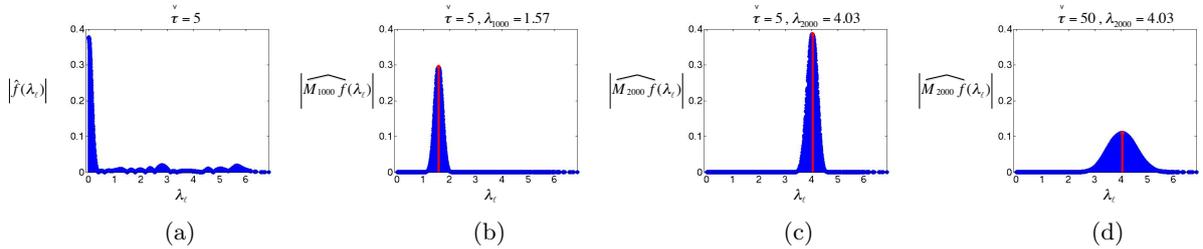


Figure 21: Generalized modulation on the Minnesota graph \mathcal{G} , via generalized translation on the same $\tilde{\mathcal{G}}$ from Figure 20, with a kernel $\hat{f}(\tilde{\lambda}_j) = e^{-\tilde{\tau} \tilde{\lambda}_j}$ defined on $\sigma(\tilde{\mathcal{L}})$. (a) We take an inverse graph Fourier transform $\hat{f}(\tilde{\lambda}_j)$ to find $\hat{f}(\lambda_\ell)$. (b) and (c) We modulate the kernel to two different center frequencies, λ_{1000} and λ_{2000} , by translating it on $\tilde{\mathcal{G}}$. In both cases, the modulated kernel is localized around the desired frequency of $\sigma(\mathcal{L})$ (shown in red). (d) As $\tilde{\tau}$ increases, the spread of the modulated kernel around the center frequency in the graph spectral domain of \mathcal{G} generally increases.

References

- [1] R. Rubinfeld, A. M. Bruckstein, M. Elad, Dictionaries for sparse representation modeling, Proc. IEEE 98 (6) (2010) 1045–1057.
- [2] 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 (2013) 83–98.
- [3] D. I Shuman, B. Ricaud, P. Vandergheynst, A windowed graph Fourier transform, in: Proc. IEEE Stat. Signal Process. Wkshp., Ann Arbor, MI, 2012, pp. 133–136.
- [4] D. Gleich, The MatlabBGL Matlab library, http://www.cs.purdue.edu/homes/dgleich/packages/matlab_bgl/index.html.
- [5] P. Flandrin, Time-Frequency / Time-Scale Analysis, Academic Press, 1999.
- [6] K. Gröchenig, Foundations of Time-Frequency Analysis, Birkhäuser, 2001.
- [7] S. G. Mallat, A Wavelet Tour of Signal Processing, Academic Press, 2008.
- [8] F. K. Chung, Spectral Graph Theory, Vol. 92 of the CBMS Regional Conference Series in Mathematics, AMS Bokstore, 1997.

- [9] X. Zhu, M. Rabbat, Approximating signals supported on graphs, in: Proc. IEEE Int. Conf. Acc., Speech, and Signal Process., Kyoto, Japan, 2012, pp. 3921–3924.
- [10] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.
- [11] O. Chapelle, B. Schölkopf, A. Zien (Eds.), Semi-Supervised Learning, MIT Press, 2006.
- [12] A. Elmoataz, O. Lezoray, S. Bougleux, Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing, IEEE Trans. Image Process. 17 (2008) 1047–1060.
- [13] Y. Dekel, J. R. Lee, N. Linial, Eigenvectors of random graphs: Nodal domains, Random Structures & Algorithms 39 (1) (2011) 39–58.
- [14] I. Dumitriu, S. Pal, Sparse regular random graphs: Spectral density and eigenvectors, Ann. Probab. 40 (5) (2012) 2197–2235.
- [15] L. V. Tran, V. H. Vu, K. Wang, Sparse random graphs: Eigenvalues and eigenvectors, Random Struct. Algo. 42 (1) (2013) 110–134.
- [16] S. Brooks, E. Lindenstrauss, Non-localization of eigenfunctions on large regular graphs, Israel J. Math.
- [17] G. Strang, The discrete cosine transform, SIAM Review 41 (1) (1999) 135–147.
- [18] R. M. Gray, Toeplitz and Circulant Matrices: A Review, Now Publishers, 2006.
- [19] R. Olfati-Saber, Algebraic connectivity ratio of Ramanujan graphs, in: Proc. Amer. Control Conf., New York, NY, 2007, pp. 4619–4624.
- [20] P. N. McGraw, M. Menzinger, Laplacian spectra as a diagnostic tool for network structure and dynamics, Phys. Rev. E 77 (3) (2008) 031102–1 – 031102–14.
- [21] N. Saito, E. Woei, On the phase transition phenomenon of graph Laplacian eigenfunctions on trees, RIMS Kokyuroku 1743 (2011) 77–90.
- [22] D. K. Hammond, P. Vandergheynst, R. Gribonval, Wavelets on graphs via spectral graph theory, Appl. Comput. Harmon. Anal. 30 (2) (2011) 129–150.
- [23] N. J. Higham, Functions of Matrices, Society for Industrial and Applied Mathematics, 2008.
- [24] L. J. Grady, J. R. Polimeni, Discrete Calculus, Springer, 2010.
- [25] K. E. Atkinson, An Introduction to Numerical Analysis, John Wiley & Sons, 1989.
- [26] M. Abramowitz, I. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, tenth Edition, Dover, 1972.
- [27] B. Metzger, P. Stollmann, Heat kernel estimates on weighted graphs, Bull. London Math. Soc. 32 (4) (2000) 477–483.
- [28] A. Agaskar, Y. M. Lu, An uncertainty principle for functions defined on graphs, in: Proc. SPIE, Vol. 8138, San Diego, CA, 2011, pp. 81380T–1 – 81380T–11.
- [29] A. Agaskar, Y. M. Lu, Uncertainty principles for signals defined on graphs: Bounds and characterizations, in: Proc. IEEE Int. Conf. Acc., Speech, and Signal Process., Kyoto, Japan, 2012, pp. 3493–3496.
- [30] O. Christensen, Frames and Bases, Birkhäuser, 2008.
- [31] J. Kovačević, A. Chebira, Life beyond bases: The advent of frames (part I), IEEE Signal Process. Mag. 24 (2007) 86–104.
- [32] J. Kovačević, A. Chebira, Life beyond bases: The advent of frames (part II), IEEE Signal Process. Mag. 24 (2007) 115–125.
- [33] A. K. Jain, F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, Pattern Recognition 24 (12) (1991) 1167–1186.
- [34] P. L. Combettes, J.-C. Pesquet, Proximal splitting methods in signal processing, in: H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, H. Wolkowicz (Eds.), Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer-Verlag, 2011, pp. 185–212.
- [35] M. Benzi, G. H. Golub, Bounds for the entries of matrix functions with applications to preconditioning, BIT 39 (1999) 417–438.
- [36] M. Benzi, N. Razouk, Decay bounds and $O(n)$ algorithms for approximating functions of sparse matrices, Elec. Trans. Numer. Anal. 28 (2007) 16–39.
- [37] G. H. Golub, G. Meurant, Matrices, Moments and Quadrature with Applications, Princeton University Press, 2010.
- [38] F. R. K. Chung, S.-T. Yau, Eigenvalues of graphs and Sobolev inequalities, Combinatorics, Prob., and Comput. 4 (1) (1995) 11–26.