A Scalable $M$-Channel Critically Sampled Filter Bank for Graph Signals

Shuni Li, Yan Jin, and David I Shuman

Abstract—We investigate a scalable $M$-channel critically sampled filter bank for graph signals, where each of the $M$ filters is supported on a different subband of the graph Laplacian spectrum. For analysis, the graph signal is filtered on each subband and downsampled on a corresponding set of vertices. However, the classical synthesis filters are replaced with interpolation operators. For small graphs, we use a full eigendecomposition of the graph Laplacian to partition the graph vertices such that the $m^{th}$ set comprises a uniqueness set for signals supported on the $m^{th}$ subband. The resulting transform is critically sampled, the dictionary atoms are orthogonal to those supported on different bands, and graph signals are perfectly reconstructable from their analysis coefficients. We also investigate fast versions of the proposed transform that scale efficiently for large, sparse graphs. Issues that arise in this context include designing the filter bank to be more amenable to polynomial approximation, estimating the number of samples required for each band, performing non-uniform random sampling for the filtered signals on each band, and using efficient reconstruction methods. We empirically explore the joint vertex-frequency localization of the dictionary atoms, the sparsity of the analysis coefficients for different classes of signals, the ability of the proposed transform to compress piecewise-smooth graph signals, and the reconstruction error resulting from the numerical approximations.

Index Terms—Graph signal processing, filter bank, non-uniform random sampling, interpolation, wavelet, compression

I. INTRODUCTION

In graph signal processing [2], transforms and filter banks can help exploit structure in the data, in order, for example, to compress a graph signal, remove noise, or fill in missing information. Broad classes of recently proposed transforms include graph Fourier transforms, vertex domain designs such as [3, 4], top-down approaches such as [5, 6, 7], diffusion-based designs such as [8, 9], spectral domain designs such as [10, 11], windowed graph Fourier transforms [12], and generalized filter banks, the focus of this paper.

The extension of the classical two channel critically sampled filter bank to the graph setting is first proposed in [13]. Fig. 1 shows the analysis and synthesis banks, where $H_1$ and $G_1$ are graph spectral filters [2], and the lowpass and highpass bands are downsampled on complementary sets of vertices. For a general weighted, undirected graph, it is not straightforward how to design the downsampling sets and the four graph spectral filters to ensure perfect reconstruction. One approach is to separate the graph into a union of subgraphs, each of which has some regular structure. For example, [14, 15] show that the normalized graph Laplacian eigenvectors of bipartite graphs have a spectral folding property that make it possible to design analysis and synthesis filters to guarantee perfect reconstruction. They take advantage of this property by decomposing the graph into bipartite graphs and constructing a multichannel, separable filter bank, while [16] adds vertices and edges to the original graph to form an approximating bipartite graph. References [17, 18] generalize this spectral folding property to $M$-block cyclic graphs, and leverage it to construct $M$-channel graph filter banks. Another class of regular structured graphs is shift invariant graphs [19, Chapter 5.1]. These graphs have a circulant graph Laplacian and their eigenvectors are the columns of the discrete Fourier transform matrix. Any graph can be written as the sum of circulant graphs, and [20, 21, 22] take advantage of this fact in designing critically sampled graph filter banks with perfect reconstruction. Another approach is to use architectures other than the critically sampled filter bank, such as lifting transforms [23, 24] or pyramid transforms [25].

Our approach in this paper, an extended version of [26], is to replace the synthesis filters with interpolation operators on each subband of the graph spectrum. While this idea was initially suggested independently in [27], we investigate it in more detail here. Our construction leverages the recent flurry of work in sampling and reconstruction of graph signals [27]. The key property we use is that any signal whose graph Fourier transform has exactly $k$ non-zero coefficients can be perfectly recovered from samples of that signal on $k$ appropriately selected vertices (see, e.g., [27, Theorem 1] [37, Proposition 1]). In Section II, we present the filter bank architecture and a method to choose the downsampling sets

![Fig. 1. Two channel critically sampled graph filter bank. Here, $H_1$ is a lowpass graph spectral filter, and $H_2$ is a highpass graph spectral filter.](http://www.macalester.edu/~dshuman/1/publications.html)
for small graphs (say 5,000 or fewer vertices).

Our main contributions in this extended version of the work include developing two fast versions of the proposed transform that scale efficiently for large, sparse graphs; i.e., that do not require a full eigendecomposition of the graph Laplacian. For the initial fast transform outlined in Section V, we design the filter bank to be more amenable to polynomial approximation, estimate the number of samples required for each band, perform non-uniform random sampling for the filtered signals on each band, and use efficient convex optimization methods to perform the interpolation for each band on the synthesis side. In Section V we develop a signal-adapted fast transform where the choices of non-uniform sampling weights and number of samples for each band depend on the actual graph signal being analyzed, as opposed to only being based on the underlying graph structure. We empirically explore these two fast transforms in Section VII including the joint vertex-frequency localization of their dictionary atoms, the sparsity of the analysis coefficients, and the reconstruction error resulting from the numerical approximations.

II. M-CHANNEL CRITICALLY SAMPLED FILTER BANK

A. Notation

We consider graph signals \( f \in \mathbb{R}^N \) residing on a weighted, undirected graph \( G = (V, E, W) \), where \( V \) is the set of \( N \) vertices, \( E \) is the set of edges, and \( W \) is the weighted adjacency matrix. Throughout, we take \( \mathcal{L} \) to be the combinatorial graph Laplacian \( \mathbf{D} - \mathbf{W} \), where \( \mathbf{D} \) is the diagonal matrix of vertex degrees. However, our theory and proposed transform also apply to the normalized graph Laplacian \( \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{W} \mathbf{D}^{-\frac{1}{2}} \), or any other Hermitian operator. We can diagonalize the graph Laplacian as \( \mathcal{L} = \mathbf{U} \Lambda \mathbf{U}^* \), where \( \Lambda \) is the diagonal matrix of eigenvalues \( \lambda_0, \lambda_1, \ldots, \lambda_{N-1} \) of \( \mathcal{L} \), and the columns \( u_0, u_1, \ldots, u_{N-1} \) of \( \mathbf{U} \) are the associated eigenvectors of \( \mathcal{L} \). The graph Fourier transform of a signal is \( f = \mathbf{U}^* f \), and \( h(\mathcal{L})f = \mathbf{U}h(\Lambda)\mathbf{U}^* f \) applies the filter \( h : [0, \lambda_{\text{max}}] \to \mathbb{R} \) to the graph signal \( f \). We let \( \mathbf{U}_S \) denote the submatrix formed by taking the columns of \( \mathbf{U} \) associated with the Laplacian eigenvalues indexed by \( S \subseteq \{0, 1, \ldots, N-1\} \), and \( \mathbf{U}_S \mathbf{R} \) denotes the submatrix formed by taking the rows of \( \mathbf{U}_R \) associated with the vertices indexed by the set \( S \subseteq \{1, 2, \ldots, N\} \).

B. Architecture

We start by constructing an ideal filter bank of \( M \) graph spectral filters, where for band endpoints \( 0 = \tau_0 < \tau_1 < \ldots < \tau_{M-1} \leq \tau_M \) (with \( \tau_M > \lambda_{\text{max}} \)), the \( m^{\text{th}} \) filter is defined as

\[
h_m(\lambda) = \begin{cases} 1, & \tau_{m-1} \leq \lambda < \tau_m, \\ 0, & \text{otherwise} \end{cases}, \quad m = 1, 2, \ldots, M. \tag{1}
\]

Fig. 2 shows an example of such an ideal filter bank. Note that for each \( \ell \in \{0, 1, \ldots, N-1\} \), \( h_m(\lambda_{\ell}) = 1 \) for exactly one \( m \).

Equivalently, we are forming a partition \( \{R_1, R_2, \ldots, R_M\} \) of \( \{0, 1, \ldots, N-1\} \) and setting

\[
h_m(\lambda_{\ell}) = \begin{cases} 1, & \ell \in R_m, \\ 0, & \text{otherwise} \end{cases}, \quad m = 1, 2, \ldots, M.
\]

The next step, which we discuss in detail in Section II-C, is to partition the vertex set \( V \) into subsets \( V_1, V_2, \ldots, V_M \) such that \( V_m \) forms a uniqueness set for \( \mathcal{U}(V_m) \).

**Definition 1** (Uniqueness set [28]). Let \( \mathcal{P} \) be a subspace of \( \mathbb{R}^n \). Then a subset \( V_\mathcal{S} \) of the vertices \( V \) is a uniqueness set for \( \mathcal{P} \) if and only if for all \( f, g \in \mathcal{P} \), \( f_{V_\mathcal{S}} = g_{V_\mathcal{S}} \) implies \( f = g \). That is, if two signals in \( \mathcal{P} \) have the same values on the vertices in the uniqueness set \( V_\mathcal{S} \), then they must be the same signal.

The following equivalent characterization of a uniqueness set is often useful.

**Lemma 1** ([30], [32]). The set \( \mathcal{S} \) of \( k \) vertices is a uniqueness set for \( \mathcal{U}(\mathcal{G}) \) if and only if the matrix whose columns are \( u_{T_1}, u_{T_2}, \ldots, u_{T_k}, \delta_{S_1}, \delta_{S_2}, \ldots, \delta_{S_{n-k}} \) is nonsingular, where each \( \delta_{S_i} \) is a Kronecker delta centered on a vertex not included in \( S \).

The \( m^{\text{th}} \) channel of the analysis bank filters the graph signal by an ideal filter on subband \( R_m \), and downsamples the result onto the vertices in \( V_m \). For synthesis, we can interpolate from the samples on \( V_m \) to \( \mathcal{U}(V_m) \). Denoting the analysis coefficients of the \( m^{\text{th}} \) branch by \( y_{V_m} \), we have

\[
f_{\text{rec}} = \sum_{m=1}^{M} \mathbf{U}_R \mathbf{U}_{V_m}^{-1} y_{V_m}. \tag{2}
\]

If there is no error in the coefficients, then the reconstruction is perfect, because \( V_m \) is a uniqueness set for \( \mathcal{U}(V_m) \), ensuring \( \mathbf{U}_{V_m} \mathbf{R}_m \) is full rank. Fig. 3 shows the architecture...
of the proposed $M$-channel critically sampled filter bank (M-CSFB) with interpolation on the synthesis side.

C. Partitioning the graph into uniqueness sets for different frequency bands

In this section, we show how to partition the set of vertices into uniqueness sets for different subbands of the graph Laplacian eigenvectors. We start with the easier case of $M = 2$ and then examine the general case.

1) $M = 2$ channels: First we show that if a set of vertices is a uniqueness set for a set of signals contained in a band of spectral frequencies, then the complement set of vertices is a uniqueness set for the set of signals with no energy in that band of spectral frequencies.

**Proposition 1.** On a graph $G$ with $N$ vertices, let $\mathcal{T} \subseteq \{0, 1, \ldots, N - 1\}$ denote a subset of the graph Laplacian eigenvalue indices, and let $\mathcal{T}^c = \{0, 1, \ldots, N - 1\} \setminus \mathcal{T}$. Then $\mathcal{S}^c$ is a uniqueness set for $\text{col}(\mathbf{U}_{\mathcal{T}^c})$ if and only if $\mathcal{S}$ is a uniqueness set for $\text{col}(\mathbf{U}_{\mathcal{T}})$.

This fact follows from either the CS decomposition [39, Equation (32)]) or the nullity theorem [40, Theorem 2.1]. We also provide a standalone proof in the Appendix that only Equation (32) or the nullity theorem [40, Theorem 2.1]. We then also provide a standalone proof in the Appendix that only Equation (32) or the nullity theorem [40, Theorem 2.1].

2) $M > 2$ channels: The issue with using the methods of Proposition 1 for the case of $M > 2$ is that the submatrix $\mathbf{U}_{\mathcal{S}, \mathcal{T}}$ is nonsingular, it is not necessarily orthogonal, and so we cannot proceed with an inductive argument. The following proposition and corollary circumvent this issue by only using the nonsingularity of the original matrix. The proof of the following proposition is due to Federico Poloni [42], and we later discovered the same method in [43], [44].

**Proposition 2.** Let $\mathbf{A}$ be an $N \times N$ nonsingular matrix, and $\mathbf{b} = \{ \beta_1, \beta_2, \ldots, \beta_M \}$ be a partition of $\{1, 2, \ldots, N\}$. Then there exists another partition $\alpha = \{ \alpha_1, \alpha_2, \ldots, \alpha_M \}$ of $\{1, 2, \ldots, N\}$ with $|\alpha_i| = |\beta_i|$ for all $i$ such that the $M$ square submatrices $\mathbf{A}_{\alpha_i, \beta_i}$ are all nonsingular.

**Proof:** First consider the case $M = 2$, and let $k = |\beta_1|$. Then by the generalized Laplace expansion [45],

$$\det(\mathbf{A}) = \sum_{\{\alpha_1 \subseteq \{1, 2, \ldots, N\} : |\alpha_1| = k\}} \sigma_{\alpha_1, \beta_1} \det(\mathbf{A}_{\alpha_1, \beta_1}) \det(\mathbf{A}_{\alpha_1^c, \beta_1^c}), \quad (3)$$

where the sign $\sigma_{\alpha_1, \beta_1}$ of the permutation determined by $\alpha_1$ and $\beta_1$ is equal to 1 or -1. Since $\det(\mathbf{A}) \neq 0$, one of the terms in the summation of (3) must be nonzero, ensuring a choice of $\alpha_1$ such that the submatrices $\mathbf{A}_{\alpha_1, \beta_1}$ and $\mathbf{A}_{\alpha_1^c, \beta_1^c}$ are nonsingular. We can choose $\{\alpha_1, \alpha_1^c\}$ as the desired partition. For $M > 2$, by induction, we have

$$\det(\mathbf{A}) = \sum_{\{\text{Partitions } \alpha \text{ of } \{1, 2, \ldots, N\} : |\alpha_i| = |\beta_i| \text{ for all } i\}} \sigma_{\alpha} \prod_{i=1}^M \det(\mathbf{A}_{\alpha_i, \beta_i}), \quad (4)$$

where again $|\sigma_\alpha| = 1$, and one of the terms in the summation in (4) must be nonzero, yielding the desired partition.

**Corollary 1.** For any partition $\{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_M\}$ of the graph Laplacian eigenvalue indices $\{0, 1, \ldots, N - 1\}$ into $M$ subsets, there exists a partition $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_M\}$ of the graph vertices into $M$ subsets such that for every $m \in \{1, 2, \ldots, M\}$, $|\mathcal{V}_m| = |\mathcal{R}_m|$ and $\mathcal{V}_m$ is a uniqueness set for $\text{col}(\mathbf{U}_{\mathcal{R}_m})$.

**Proof:** By Proposition 2 we can find a partition such that $\mathbf{U}_{\mathcal{V}_m, \mathcal{R}_m}$ is nonsingular for all $m$. Let $\mathbf{E}_m$ be the matrix formed by joining the $k_m$ columns of $\mathbf{U}$ indexed by $\mathcal{R}_m$ with $N - k_m$ Kronecker deltas centered on all vertices not included in $\mathcal{V}_m$. By Lemma 1 it suffices to show that the matrices $\mathbf{E}_m$ are all nonsingular. Yet, for all $m$, we have $|\det(\mathbf{E}_m)| = |\det(\mathbf{U}_{\mathcal{V}_m, \mathcal{R}_m})| \neq 0$.

Corollary 1 ensures the existence of the desired partition, and the proof of Proposition 2 suggests that we can find it inductively. However, given a partition of the columns of $\mathbf{A}$ into two sets $\mathcal{T}$ and $\mathcal{T}^c$, Proposition 2 does not provide a constructive method to partition the rows of $\mathbf{A}$ into two sets $\mathcal{S}$ and $\mathcal{S}^c$ such that the submatrices $\mathbf{A}_{\mathcal{S}, \mathcal{T}}$ and $\mathbf{A}_{\mathcal{S}^c, \mathcal{T}^c}$ are nonsingular. This problem is studied in the more general framework of matroid theory in [44], which gives an algorithm to find the desired row partition into two sets. We summarize this method in Algorithm 1 which takes in a partition $\{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_M\}$ of the spectral indices and constructs the partition $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_M\}$ of the vertices. In Fig. 4 we show two examples of the resulting partitions.

**Remark 1.** While Algorithm 1 always finds a partition into uniqueness sets, such a partition is usually not unique. The initial choices of $\gamma_i$ in each loop play a significant role in the final partition. In the numerical experiments in Section VII we use the greedy algorithm in [42, Algorithm 1] to find an initial choice for $\gamma_1$, permute the complement of $\gamma_1$ to the top, and then perform row reduction to find an initial choice for $\gamma_2$.

---

**Algorithm 1** Partition the vertices into uniqueness sets for each frequency band

**Input** $\mathbf{U}$, a partition $\{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_M\}$

**Output** the partition $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_M\}$
D. Transform properties

1) Dictionary atoms: Let \( M_m \in \mathbb{R}^{|V_m| \times N} \) be the down-sampling matrix for the \( m \)th channel. That is, \( M_m(i,j) = 1 \) if vertex \( j \) is the \( i \)th element of \( V_m \), and 0 otherwise. The proposed transform is a linear mapping \( \mathcal{F} : \mathbb{R}^N \rightarrow \mathbb{R}^N \) by \( \mathcal{F} = \Phi \mathcal{F} \), where the resulting dictionary is of the form \( \Phi := [h_1(\mathcal{L})M_1^T | h_2(\mathcal{L})M_2^T | \ldots | h_M(\mathcal{L})M_M^T] \). While the transform is not orthogonal, each atom is orthogonal to all atoms concentrated on other spectral bands. Informally, this is because the atoms are projections of Kronecker deltas onto the orthogonal subspaces spanned by the Laplacian eigenvectors of each band. More formally, each atom is of the form \( h_m(\mathcal{L})\delta_i \), where vertex \( i \) is in \( V_m \). If \( m \neq m' \), then the inner product of the two atoms from different bands is given by

\[
\langle h_m(\mathcal{L})\delta_i, h_{m'}(\mathcal{L})\delta_{i'} \rangle = \delta_i^\top U h_m(\Lambda)U' U h_{m'}(\Lambda)\delta_{i'} = 0,
\]

since \( U^\top U = I \) and \( h_m(\lambda)h_{m'}(\lambda) = 0 \) for all \( \lambda \) by design. Note also that the wavelet atoms at all scales \( (m > 1) \) have mean zero, as they have no energy at eigenvalue zero.

2) Signals that are sparsely represented by the M-CSFB transform: Globally smooth signals trivially lead to sparse analysis coefficients because the coefficients are only nonzero for the first set(s) of vertices in the partition. More generally, signals that are concentrated in the graph Fourier domain have sparse M-CSFB analysis coefficients, because the coefficients for any channel whose filter does not overlap with the support of the signal in the graph Fourier domain are all equal to zero.

III. ILLUSTRATIVE EXAMPLES I: EXACT CALCULATIONS

A. Joint vertex-frequency localization of atoms and example analysis coefficients

We start by empirically showing that the dictionary atoms (the columns of \( \Phi \)) are jointly localized in the vertex and graph spectral domains. On the Stanford bunny graph \([47]\) with 2503 vertices, we partition the spectrum into five bands, and show the resulting partition into uniqueness sets in Fig. 6(c). The first row of Fig. 6 shows five different example atoms whose energies are concentrated on different spectral bands. These atoms are also generally localized in the vertex domain; however, the atom at wavelet scale 3 is more spread in the vertex domain, possibly as a result of using ideal filters in the filter bank, an issue we revisit in Section IV-F. The second row of Fig. 6 shows the spectral content of all atoms in each band, with the average for each represented by a thick black line. As expected, the energies of the atoms are also localized in the spectral domain.

Next we apply the proposed transform to a piecewise-smooth graph signal \( f \) that is shown in the vertex domain in Fig. 6(a), and in the graph spectral domain in Fig. 6(b). The full set of analysis coefficients is shown in Fig. 6(d), and these are separated by band in the third row of Fig. 6(c). We see that with the exception of the lowpass channel, the coefficients are clustered around the two main discontinuities (around the midsection and tail of the bunny). The bottom row of Fig. 6(c) shows the interpolation of these coefficients onto the corresponding spectral bands. If we sum these reconstructions together, we recover the original signal in Fig. 6(a).

B. Sparse approximation

Next, we compress a piecewise-smooth graph signal \( f \) via the sparse coding optimization

\[
\arg\min_{x} ||f - \Phi x||_2^2 \text{ subject to } ||x||_0 \leq T,
\]

where \( T \) is a predefined sparsity level. After normalizing the atoms of various critically-sampled dictionaries, we use the greedy orthogonal matching pursuit (OMP) algorithm \([48], [49]\) to approximately solve (6). We show the normalized mean square reconstruction errors (NMSE) \( \frac{||f - \hat{f}||_2^2}{||f||_2^2} \) in Fig. 7(d). For the M-CSFB, Fig. 6(c) shows the partition into uniqueness sets, and Fig. 2 shows the filter bank.

IV. FAST M-CSFB TRANSFORM

In the numerical examples in the previous sections, we have computed a full eigendecomposition of the graph Laplacian and used it for all three of the filtering, sampling, and interpolation operations; however, such an eigendecomposition does not scale well with the size of the graph as it requires \( O(N^3) \) operations with naive methods. In the next three sections, we develop a fast approximate version of the proposed transform that scales more efficiently for large, sparse graphs.

A. Approximation by polynomial filters

Fast, approximate methods for computing \( h_m(\mathcal{L})f \), a function of sparse matrix times a vector, include approximating the function \( h_m(\cdot) \) by a polynomial (e.g., via a truncated Chebyshev or Legendre expansion), approximating \( h_m(\cdot) \) by a rational function, Krylov space methods (Lanczos in our case of a symmetric matrix \( \mathcal{L} \)), and the matrix version of the Cauchy integral theorem (see, e.g., \([50]-[53]\) and references therein). The first three of these methods have been examined in graph signal processing settings \([10], [33], [54], [57]\). Here, we develop an efficient, polyapproximation of the analysis side filters, focus on order \( K \) Chebyshev polynomial approximations of the form

\[
\tilde{h}(\mathcal{L})f := \sum_{k=0}^{K} \alpha_k \bar{T}_k(\mathcal{L})f.
\]

In (7), \( \bar{T}_k(\cdot) \) are Chebyshev polynomials shifted to the interval \([0, \lambda_{\text{max}}]\). Thus, \( \bar{T}_0(\mathcal{L})f = f, \bar{T}_1(\mathcal{L})f = \frac{\mathcal{L}f - f}{\lambda_{\text{max}}} \), and for

Fig. 4. Partitions of a 500 node random sensor network and the Minnesota road network \([26]\) into uniqueness sets for five different spectral bands, with the indices increasing from lowpass bands (1) to highpass bands (5).
In an attempt to at least preserve near orthogonality of the atoms across bands shown in (5).

However, the oscillations that arise in Chebyshev polynomial approximations of bandpass filters may result in larger values of \( h_m(\lambda) \hat{h}_m(\lambda) \), even when the ideal filters \( h_m(\cdot) \) and \( \hat{h}_m(\cdot) \) have supports that do not come close to overlapping. This negates the orthogonality of the atoms across bands shown in (5). In an attempt to at least preserve near orthogonality across bands, we therefore use the Jackson-Chebyshev polynomial approximations from (8) that damp the Gibbs oscillations appearing in Chebyshev expansions. With the damping,

\[
\alpha_0 = \frac{1}{2} c_0 \quad \text{and} \quad \alpha_k = \gamma_{k,K} c_k \quad \text{for} \quad k = 1, 2, \ldots, K, \tag{10}
\]

where, as presented in (8),

\[
\gamma_{k,K} = \frac{1}{(1 - \frac{k}{K+2}) \sin(\frac{\pi}{K+2}) \cos(\frac{k\pi}{K+2}) + \frac{1}{K+2} \cos(\frac{\pi}{K+2}) \sin(\frac{k\pi}{K+2})} \sin(\frac{\pi}{K+2}). \tag{11}
\]

Fig. 8 shows the Jackson-Chebyshev polynomial approximations for ideal bandpass filters of two different graphs.\(^1\)
We can quantify the worst case error introduced when approximating \( h_m(\cdot) \) by an approximant \( \tilde{h}_m(\cdot) \) as follows:

\[
||\tilde{h}_m(L) - h_m(L)||_2 = \max_{\ell \in \{0, 1, \ldots, N-1\}} |\tilde{h}_m(\lambda_{\ell}) - h_m(\lambda_{\ell})| 
\leq \max_{\lambda \in [0, \lambda_{\max}]} |\tilde{h}_m(\lambda) - h_m(\lambda)|. \tag{12}
\]

While approximation theory often aims to minimize the upper bound in \( \text{(12)} \), only the errors exactly at the graph Laplacian eigenvalues affect the overall approximation error \( ||\tilde{h}_m(L) - h_m(L)||_2 \). Since, as seen in Fig. 8, the errors of the Jackson-Chebyshev polynomial approximation are concentrated around the discontinuities of \( h_m(\cdot) \), a guiding principle when designing the filter bank to be more amenable to fast approximation is to choose the endpoints \( \{\tau_m\}_{m=1, \ldots, M-1} \) of the bandpass filters to be in gaps in the graph Laplacian spectrum. Unfortunately, we do not have access to the exact graph Laplacian eigenvalues (the reason for introducing this approximation in the first place is that they are too expensive to compute for large graphs); however, we can efficiently estimate the density of the spectrum in order to design the filters so that they are close to fewer eigenvalues of \( L \).

### 1) Estimating the spectral density

Lin et al. [61] provide an excellent overview of methods to approximate the spectral density function \( \rho_{\lambda}(s) := \frac{1}{N} \sum_{\ell=0}^{N-1} \mathbb{I}_{\{\lambda_{\ell}=s\}} \) of a matrix, which in our context for the graph Laplacian \( L \) is the probability measure \( \rho_{\lambda}(s) := \frac{1}{N} \sum_{\ell=0}^{N-1} \mathbb{I}_{\{\chi_{\ell}=s\}} \). Here, we use a variant of the Kernel Polynomial Method \( [64-66] \) described in \( [61] \) to estimate the cumulative spectral density function or empirical spectral cumulative distribution

\[
P_{\lambda}(z) := \frac{1}{N} \sum_{\ell=0}^{N-1} \mathbb{I}_{\{\lambda_{\ell} \leq z\}}. \tag{13}\]

The procedure starts by estimating \( \lambda_{\max} \), for example via the power iteration. Then for each of \( T \) linearly spaced points \( \xi \), between 0 and \( \lambda_{\max} \), we use Hutchinson’s stochastic trace estimator \( [67] \) to estimate \( \eta_i \), the number of eigenvalues less than or equal to \( \xi_i \). Defining the Heaviside function \( \Theta_{\xi_i}(\lambda) := \mathbb{I}_{\{\lambda \leq \xi_i\}} \), we have

\[
\eta_i = \text{tr} \left( \Theta_{\xi_i}(L) \right) = \mathbb{E}[x^\top \Theta_{\xi_i}(L)x] \tag{14}\]

\[
\approx \frac{1}{J} \sum_{j=0}^{J} x(j)^\top \Theta_{\xi_i}(L)x(j) \tag{15}\]

\[
\approx \frac{1}{J} \sum_{j=1}^{J} x(j)^\top \hat{\Theta}_{\xi_i}(L)x(j). \tag{16}\]

In \text{(14)}, \( x \) is a random vector with each component having an independent and identical standard normal distribution. Each vector \( x(j) \) in \text{(15)} is chosen according to this same distribution, and in our experiments, we take the default number of vectors to be \( J = 30 \). In \text{(16)}, \( \hat{\Theta}_{\xi_i} \) is the Jackson-Chebyshev approximation to \( \Theta_{\xi_i} \) discussed in Section \IV-A. If we place the \( J \) random vectors into the columns of an \( N \times J \) matrix \( X \), the computational cost of estimating the spectral distribution is dominated by computing

\[
\hat{\Theta}_{\xi_i}(L)X = \sum_{k=0}^{K} \alpha_k \tilde{T}_k(L)X \tag{17}\]

for each \( \xi_i \). Yet, we only need to compute \( \{\tilde{T}_k(L)X\}_{k=0,1,\ldots,K} \) recursively once, as this sequence can be reused for each \( \xi_i \), with different choices of the \( \alpha_k \)'s. Therefore, the overall computational cost is \( O(KJ|\mathcal{E}|) \).

As in \text{(11)}, once we compute the eigenvalue count estimates \( \{\eta_i\} \), we approximate the empirical spectral cumulative distribution \( P_{\lambda}(\cdot) \) by performing monotonic piecewise cubic interpolation \( [68] \) on the series of points \( \{(\xi_i, \frac{\eta_i}{N})\}_{i=1,2,\ldots,T} \). We denote the result as \( \hat{P}_{\lambda}(\cdot) \).

#### 2) Choosing initial band ends

When selecting the band ends \( \{\tau_m\} \) for each of the \( M \) ideal filters, we consider two factors: spectrum-adaptation and spacing. In our implementation, the filter bank can either be adapted to the spectrum or just to the support of the spectrum \([0, \lambda_{\max}]\), and it can be either evenly or logarithmically spaced (four options in all). For example, if the filter bank is only adapted to the support of the spectrum and is evenly spaced, then \( \tau_m = \frac{m}{M} \lambda_{\max} \).

Fig. 9(a)-(b) show a spectrum-adapted, logarithmically spaced choice with \( \tau_m = \tilde{P}_{\lambda}^{-1}\left(\frac{1}{2}M^{-m}\right) \) for \( m = 1, 2, \ldots, M \), such that approximately half of the eigenvalues are in the highest band, a quarter in the next highest band, and so forth.

#### 3) Adjusting the band ends

In order to make the filters more amenable to approximation, we then adjust the initial choice of band endpoints so that they lie in lower density regions of the spectrum. Specifically, for each \( m = 1, 2, \ldots, M-1 \) and some \( \Delta > 0 \), we let the final endpoint be

\[
\tau_m^* = \arg \min_{\tau \in I_m} \left\{ \frac{\hat{P}_{\lambda}(\tau + \Delta) - \hat{P}_{\lambda}(\tau - \Delta)}{2\Delta} \right\}, \tag{18}\]

where \( I_m \) is an interval around the initial choice of \( \tau_m \). Fig. 9(c) shows the objective function in \text{(18)}, along with the initial band ends, search intervals, and adjusted band ends.
Comparing Fig. 9(b) and Fig. 9(d), the band end adjustments lead to fewer eigenvalues falling close to the filter borders, reducing the error incurred by the polynomial approximation process. Algorithm 2 summarizes the filter bank design in the case of spectrum-adapted and logarithmically spaced filters.

C. Non-uniform random sampling distribution

The partitioning of the vertices into uniqueness sets described in Algorithm 1 requires a full eigendecomposition of the graph Laplacian to compute the matrix $\mathbf{U}$. Two broad approaches to more efficient sampling have recently been investigated: greedy methods [27], [30], [35], [37] and random sampling methods [32], [33], [36], which have close connections to leverage score sampling in the statistics and numerical linear algebra literature (see, e.g., [69], [71]). Reference [37] has a nice review of the computational complexities of the various greedy routines for identifying uniqueness sets. Most of these are designed specifically for lowpass signals.

We adapt the non-uniform random sampling method of [33], which scales more efficiently than greedy methods. Namely, for the $m^{th}$ band, we identify the downsampling set $V_m$ by sampling the vertices $V$ without replacement according to a discrete probability distribution $\omega_m$. To minimize the graph weighted coherence, it is ideal to take $\omega_m(i) \propto |\langle \hat{h}_m(\mathbf{L})\mathbf{X} \rangle_\delta_i|^2$ [33]; however, we do not have access to $\mathbf{U}_R$. Instead, we take

$$\omega_m(i) \propto |\langle \tilde{h}_m(\mathbf{L})\mathbf{X} \rangle_\delta_i|^2,$$

which [33] shows is an unbiased estimator of $|\mathbf{U}_R^T \delta_i|^2$ when $\mathbf{X}$ is the random matrix from [17]. Since we already compute and store the series of matrices $\{\tilde{T}_k(\mathbf{L})\mathbf{X}\}_{k=0,1,\ldots,K}$ for the spectral density estimation of Section IV-B1, we just need to compute the polynomial approximation coefficients $\{\eta_k\}$ in [17] for $h_m(\lambda)$ in order to compute $\tilde{h}_m(\mathbf{L})\mathbf{X}$.

Intuitively, the sampling distribution approximates the energy of the selected eigenvectors concentrated on each vertex. In the extreme case that the selected eigenvectors are completely concentrated on a single vertex or small neighborhood of vertices, sampling signal values outside of this set provides no additional information, justifying the zero weight in the sampling distribution. For eigenvectors whose energy is equally spread across the graph, this results in uniform sampling. In particular, for any walk-regular graph, a class that includes vertex-transitive graphs, which in turn include shift-invariant graphs such as the cycle graph, $|\mathbf{U}_R^T \delta_i|^2$ is constant across vertices $i$ for any choice of eigenvectors $\mathbf{R}$ [72, Corollary 3.2], resulting in uniform random sampling for

Algorithm 2 Spectrum-adapted and logarithmically spaced filter bank design

**Input** graph $G$, estimate for $\lambda_{\text{max}}$, number of bands $M$, $\Delta > 0$, degree $K$, number of random vectors $J$

Generate an $N \times J$ matrix $\mathbf{X}$ whose columns are i.i.d. standard normal random vectors

Compute $\{\tilde{T}_k(\mathbf{L})\mathbf{X}\}_{k=0,1,\ldots,K}$ via (8)

Choose $T$ linearly spaced points $\{\xi_i\}_{i=1,\ldots,T}$ between 0 and $\lambda_{\text{max}}$

for $i = 1, 2, \ldots, T$

Approximate $\eta_i$ via (16) and (17)

end for

Estimate the spectral density function $\tilde{P}_\lambda$ by performing monotonic cubic interpolation on the set of points $\{\xi_i, \frac{\Delta}{2}\}$ for $m = 1, 2, \ldots, M$

Compute the initial band end: $\tau_m = \tilde{P}_\lambda^{-1}(\frac{1}{2}M-m)$

end for

Set $\tau_0^* = \tau_0 = 0$, $\tau_M^* = \tau_M$.

for $m = 1, 2, \ldots, M-1$

Set the search radius:

$$r = \min\left\{\frac{\tau_m - \tau_{m-1}}{2}, \frac{\tau_{m+1} - \tau_m}{2}\right\}$$

Set the search interval: $I_m = [\tau_m - r, \tau_m + r]$

Update the band ends:

$$\tau_m^* = \arg\min_{\tau \in I_m} \frac{\tilde{P}_\lambda(\tau + \Delta) - \tilde{P}_\lambda(\tau - \Delta)}{2\Delta}$$

end for

for $m = 1, 2, \ldots, M$

Construct the ideal filter $h_m(\lambda)$ according to (1) using $\tau_m^*$ and $\tau_m^*$

end for

**Output** $\mathbf{X}$, $\{\tilde{T}_k(\mathbf{L})\mathbf{X}\}_{k=0,1,\ldots,K}$, and the polynomial filters $\{h_1(\lambda), h_2(\lambda), \ldots, h_M(\lambda)\}$, where each $h_m(\lambda)$ is a degree $K$ Jackson-Chebyshev polynomial approximation to $h_m(\lambda)$.
Algorithm 3 Construct the downsampling sets

\[
\text{Input} \ G, \ X, \ \{T_k(\mathcal{L})X\}_{k=0,1,\ldots,K}, \ \text{polynomial filters} \ \{\tilde{h}_1(\lambda), \tilde{h}_2(\lambda), \ldots, \tilde{h}_M(\lambda)\}, \ \text{signal} \ f (\text{optional})
\]

for \( m = 1, 2, \ldots, M \) do
  For \( \tilde{h}_m(\cdot) \), compute \( \{\alpha_{m,k}\} \) (the \( \{\alpha_k\} \) in (17)) via (9)-(11)
  \[\text{Compute } \tilde{h}_m(\mathcal{L})X = \sum_{k=0}^{K} \alpha_{m,k} T_k(\mathcal{L})X\]
  Set the weight for each vertex \( i \in V \):
  \[\omega_m(i) = ||\hat{U}_m(:,i)||^2_2\]
  if signal-adapted weights then
    Compute \( \hat{h}_m(\mathcal{L})f \) via (7) with the same \( \{\alpha_{m,k}\} \)
    Adapt the weights:
    \[\omega_m(i) = \omega_m(i) \cdot \log(1 + ||\hat{h}_m(\mathcal{L})f(i)||)\]
  end if
  Normalize the weights: \( \omega_m(i) = \frac{\omega_m(i)}{\sum_{j=1}^{n} \omega_m(i)} \)
  Set the initial number of samples based on (16):
  \[n_m = \frac{1}{f} \text{Trace}(X^T \tilde{h}_m(\mathcal{L})X)\]
  if signal-adapted number of samples then
    for \( m = 1, 2, \ldots, M \) do
      Set \( n_m = n_m \cdot \log(1 + ||\hat{h}_m(\mathcal{L})f||)\)
    end for
  end if
  Compute total initial number of samples: \( N_0 = \sum_m n_m \)
  for \( m = 1, 2, \ldots, M \) do
    Normalize the number of total samples:
    \[n_m = \text{round}\left(\frac{n_m}{N_0} N_T\right)\]
    where \( N_T \) is the target number of samples (e.g., \( N_T = N \) for critical sampling)
  end for
  Adjust to meet target number of samples:
  if \( \sum_m n_m > N_T \) then
    Set \( n_M = n_M - (\sum_m n_m - N_T) \)
  else if \( \sum_m n_m < N_T \) then
    Set \( n_1 = n_1 + (N_T - \sum_m n_m) \)
  end if
  for \( m = 1, 2, \ldots, M \) do
    Choose the downsampling set \( V_m \) by randomly sampling \( n_m \) vertices according to the distribution \( \omega_m \)
  end for
Output downsampling sets \( \{V_1, V_2, \ldots, V_M\} \), sampling distributions \( \{\omega_1, \omega_2, \ldots, \omega_M\} \)

D. Number of samples

One option to ensure critical sampling is to choose the number of samples for each band according to the initial filter bank design. For example, if the filter bank is designed to be adapted to the spectrum with logarithmic spacing, we can choose \( \frac{N}{2} \) samples for the highest band, \( \frac{N}{4} \) for the next highest, and so forth. However, the adjustments we make in Section IV-B3 affect the number of eigenvalues contained in each band. Since we have an estimate of the cumulative spectral distribution, one approximation for the number of samples in the adjusted \( m^{th} \) band is to round \( N \cdot (P_\lambda(\tau_m) - P_\lambda(\tau_{m-1})) \).

As a band end \( \tau_m \) may fall at a point where \( P_\lambda \) has been interpolated via cubic functions, another option is to estimate the number of eigenvalues between \( \tau_{m-1} \) and \( \tau_m \), once again with the stochastic trace estimator in (16), except using the bandpass filter \( h_m(\lambda) \) from (11). We already compute \( \hat{h}_m(\mathcal{L})X \) to calculate the sampling distribution in (19). We can substitute the columns \( \hat{h}_m(\mathcal{L})\tilde{\chi}_j \) of this matrix into (16) for an estimate of the number of eigenvalues in the \( m^{th} \) band. An added benefit of this extra step is that the thresholds \( \{\tau_m\} \) are chosen to be in areas of low spectral density, which improves the accuracy of the eigenvalue count estimate (58).

We make small adjustments to ensure the total number of samples is equal to some target \( N_T \). In our experiments, we take \( N_T = N \) to ensure critical sampling. Our default is to add samples to the lowest band if the normalized total is below \( N_T \), and remove samples from the highest band if the normalized total is above \( N_T \). Algorithm 3 summarizes the proposed method to choose the downsampling sets.

Note that \( \text{dim}(\text{col}(\hat{h}_m(\mathcal{L}))) \geq \text{dim}(\text{col}(h_m(\mathcal{L}))) \), with the difference depending on the number of Laplacian eigenvalues just outside the end points of \( h_m(\cdot) \) and the degree of approximation used for \( \hat{h}_m(\cdot) \). Therefore, we expect that to perfectly reconstruct signals in \( \text{col}(\hat{h}_m(\mathcal{L})) \), we need more samples than the number of eigenvalues in the support of \( h_m(\cdot) \). In Section VII we explore how the reconstruction error is reduced as we increase the number of samples in each band.

E. Interpolation

The exact interpolation (2) requires the eigenmatrix \( U \), and in case \( U_{V_m,R_m} \) is not full rank, the standard least squares reconstruction for the \( m^{th} \) channel

\[f_{m,\text{rec}} = U_{R_m}(U_{Y_m,R_m} U_{V_m,R_m})^{-1} U_{Y_m,R_m} Y_{V_m}\]

also requires \( U_{R_m} \). One option explored in [73], [74] is to leverage \( \{\tilde{T}_k(\mathcal{L})X\} \) again to approximate the column space of \( U_{R_m} \) by filtering at least \([R_m]\) standard normal random vectors with the filter \( \hat{h}_m(\cdot) \), possibly followed by orthonormalization via QR factorization.

A second approach suggested in [33] to efficiently reconstruct lowpass signals is to relax the optimization problem

\[\min_{\mathbf{z} \in \text{col}(U_{R_m})} ||\Omega_{m,Y_m}^{-\frac{1}{2}} (M_m \mathbf{z} - Y_{V_m})||^2_2\]

to

\[\min_{\mathbf{z} \in \mathbb{R}^n} \left\{ \mathbf{z}^T \varphi_m(\mathcal{L}) \mathbf{z} + \kappa ||\Omega_{m,Y_m}^{-\frac{1}{2}} (M_m \mathbf{z} - Y_{V_m})||_2^2 \right\}, \tag{21}\]
where $\Omega_m, \forall_m$ is a $|\mathcal{V}_m| \times |\mathcal{V}_m|$ diagonal matrix with the $m^{th}$ channel sampling weights of $\mathcal{V}_m$ along the diagonal, and $\kappa > 0$ is a parameter to trade off the two optimization objectives. The regularization term $z^T \psi_m(\mathcal{L}) z$ in (21) penalizes reconstructions with support outside of the desired spectral band. For lowpass signals, Puy et al. [33] take the penalty function $\psi_m(\lambda)$ to be a nonnegative, nondecreasing polynomial, such as $\lambda^l$, with $l$ a positive integer. For more general classes of signals (i.e., the midpass and highpass signals output from the higher bands of the proposed filter bank), it is important to keep the nonnegativity property, in order to ensure that $\psi_m(\mathcal{L})$ is positive semi-definite and the optimization problem (21) is convex. However, we can drop the nondecreasing requirement, and instead choose penalty functions concentrated outside the $m^{th}$ spectral band. Options we explore include (i) the polynomial filter $\psi_m(\lambda) = 1 - \hat{h}(\lambda)$; (ii) the rational filter $\psi_m(\lambda) = \frac{1}{h(\lambda) + \epsilon}$; and (iii) a polynomial approximation of a penalty function constructed as a piecewise cubic spline, an approach explored in [75]. See Fig. 10 for example graphs of these penalty functions.

From the first-order optimality conditions, the solution to (21) is the solution to the linear system of equations

$$\left(\kappa M_m^{-1} \Omega_m^{-1} M_m + \psi_m(\mathcal{L})\right) z = \kappa M_m^{-1} \Omega_m^{-1} M_m \forall_m vy_m,$$

which can be solved, for example, with the preconditioned conjugate gradient method. For the preconditioner, we use a diagonal matrix whose $i^{th}$ element is equal to 1 if $i \notin \mathcal{V}_m$ and $1 + \frac{\kappa}{\omega_m(i)}$ if $i \in \mathcal{V}_m$, which serves as an approximation to the matrix $\kappa M_m^{-1} \Omega_m^{-1} M_m + \psi_m(\mathcal{L})$ in (22).

\section{V. Signal-Adapted Fast M-CSFB Transform}

Just as it is helpful for interpolation to sample more signal values at vertices where the energy of the selected eigenvectors is concentrated, it is also helpful to sample more values where the energy of the filtered signals is concentrated. Thus, in this section, we make three adaptations to the fast M-CSFB transform.

First, we subtract the mean of the signal (i.e., let $f = f - \frac{1}{N} \sum f$) before sending it into the filter bank, and then add this constant back to every vertex when summing the interpolations from the $M$ channels. To ensure critical sampling, we only allow $N - 1$ total samples in addition to this mean measurement. Second, we adapt the sampling weights by setting $\omega_m(i) = \omega_m(i) \cdot \log \left(1 + \left| \hat{h}_m(\mathcal{L}) f(i) \right| \right)$. Thus, if a filtered signal on a given band is concentrated on a certain region of the graph, the sampling weights are concentrated on the intersection of that region and the set of vertices where the energies of the selected eigenvectors are concentrated.

Third, beyond the distribution of samples within each band, we need to decide how many samples to allocate to each band. In the exact computation (small graph) case, allocating the samples according to the number of eigenvalues contained in the disjoint bands ensures perfect reconstruction. However, with approximate computations, it is beneficial to the overall reconstruction error to do a better job of interpolation on the bands whose filtered signals have the most energy. We set the initial number of samples by multiplying the estimate of the number of eigenvalues in the band with $\log(1 + \left| \hat{h}_m(\mathcal{L}) f(i) \right|)$, as shown in Algorithm 3. In the extreme case of a filtered signal with no energy, this choice leads to zero measurements and a reconstruction of the all zero vector.

Note that when analyzing a single signal, these adaptations do not add significantly to the computational complexity of the transform. However, if we are repeating the transform on many different signals residing on the same graph, we do need to rerun the random selection of vertices for each signal.
VI. ILLUSTRATIVE EXAMPLES II: APPROXIMATE CALCULATIONS

First, in Fig. 11 we explore the tradeoff between the number of samples and the reconstruction error. Here, we take $K = J = 50$, $\kappa = 1$, the tolerance for the relative residual in the preconditioned conjugate gradient to be $10^{-10}$, and $\{\varphi_m\}$ to be the spline-based penalty filters. Note that the polynomial approximated filters have wider supports as compared to the ideal filters. By performing critical sampling based on the estimated supports of the ideal filters, we may not have enough samples to reconstruct signals from the (wider) filtered subspace. In order to get a better reconstruction, we can include more samples for each band. The second and third rows up from the bottom in Fig. 11 show the absolute values of the reconstruction errors, averaged over 50 trials of the random sampling, when the number of random samples is equal to the estimated number of eigenvalues in the specified band (173 in the lowpass case and 445 in the bandpass case). The last row demonstrates how the reconstruction errors decrease as we increase the number of samples for each band, up to three times the estimated number of eigenvalues. We also see the effect of adapting the non-uniform sampling distributions to the filtered signal, as discussed in Section V.

Next, we examine average temperatures for March 2018, taken from the Gridded 5km GHCN-Daily Temperature and Precipitation Dataset (nClimGrid) [76], [77] of the United States National Oceanic and Atmospheric Administration (NOAA). The measurements are taken on a grid with spacing of $\frac{\pi}{2}$ of a degree for both latitude and longitude. We form an unweighted graph by connecting each measurement location to its eight neighbors on the grid, if they contain measurements, as shown in Fig. 12(a)-(b). We then eliminate isolated vertices and small components (e.g., islands) to yield a connected graph with 469,404 vertices (measurement locations). Fig. 12(c)-(e) show an estimate of the cumulative spectral distribution, the resulting approximate filter bank with five bands and degree 50 polynomial approximations, and the temperatures.

An intuitive explanation of the distribution for the second band, shown in Fig. 12(g), is that we want to sample with higher probability near the edges, as there are fewer neighbors from whom to interpolate the local average value. The 27,021 vertices selected for $V_2$, shown in Fig. 12(i), are spread across the graph, with a higher density near the boundaries. The corresponding non-uniform sampling distribution and realization for the second band in the signal-adapted case are shown in Fig. 12(h) and Fig. 12(j). The non-zero analysis coefficients of the bandpass filters tend to coincide with topographical changes, as shown Fig. 12(f). Accordingly, the signal-adapted transform selects more samples from these regions.

Two scaling functions for the fast M-CSFB are shown in Fig. 12(k)-(l), with the image for the latter one zoomed in near the Great Salt Lake. Because $K = 50$, the atoms are localized within 50 hops of the center vertex. When the center is in a region where each vertex has eight neighbors, the atom is symmetric, resembling the scaling function of a 2D-wavelet transform. When the center is near an edge, like the one shown in Fig. 12(l), the atom adapts to the shape of the underlying graph. While the signal adaptation changes the distribution of center locations, it does not change the shape of the atoms; that is, if a vertex $i$ is chosen as a center location for a given scale in both the non-adapted and signal-adapted versions of the transform, the corresponding atoms are identical.

The magnitudes of the analysis coefficients, shown in Fig. 12(m)-(n), decay quickly, as the signal is generally smooth.
with some discontinuities that tend to coincide with topographical changes. The signal-adapted transform allocates more samples to the bands on the lower end of the spectrum (57,539 total for the first two bands, as opposed to 43,384 in the non-adapted case). Fig. 12(o)-(p) show the reconstruction errors for the two versions of the fast $M$-CSFB transform, with the same parameters listed at the beginning of this section, except $J = 30$. The MSE of the signal-adapted transform is 0.0479, as opposed to 0.4994 without the signal adaptation. The first driver of this reduction is the allocation of additional samples to the first band, where 0.4586 of the 0.4994 MSE is incurred when not adapted to the signal. Additionally, more of this band’s samples are taken in the upper and lower thirds of the country, where the energy of the filtered signal is concentrated.

In Table 4, we compare the computation times of the proposed transform to those of the exact graph Fourier transform (i.e., a full diagonalization of $\mathcal{L}$); diffusion wavelets [8] with five scales (one scaling and four wavelets) and a precision of $\epsilon = 1e-4$; and a graph quadrature mirror filter bank [13] with polynomial approximation order of $K = 50$. For the fast $M$-CSFB (original and signal adapted versions), we consider two scenarios. Scenario A is faster, but less accurate, with $K = 25$, a conjugate gradient (CG) tolerance of $1e-8$, and a maximum of 100 CG iterations. For Scenario B, we take $K = 50$, the CG tolerance to be $1e-10$, and a maximum of 250 iterations. For all fast $M$-CSFB cases, we let $M = 5$, $J = 30$ and $\kappa = 1$, and subtract out the mean of the signal before applying the filter bank. For larger graphs, calculations such as a full diagonalization are either not possible due to memory limits or would take in excess of a day to compute. We denote these by NA. We apply all transforms to the signals shown above for the sensor network, bunny graph, and temperature network, and to Gaussian random vectors with independent entries for the net25 and community (100 communities) graphs.

VII. CONCLUSION AND EXTENSION

We have proposed a fast, critically sampled transform that approximately projects a graph signal onto different bands of the graph Laplacian spectrum. To improve computational efficiency, we leveraged the computation of $\{T_k(\mathcal{L})X\}$ in
multiple ways: to estimate the spectral density for the design of the filter bank, to estimate the number of samples for each band, and to estimate the non-uniform sampling distribution. The key idea behind the filter bank design is to choose the end points of each band to be in less dense regions of the spectrum so that the resulting filters are more amenable to polynomial approximation. Adapting the non-uniform sampling distribution and allocation of the samples across the bands to the specific signal being analyzed improves the speed and accuracy of the synthesis process without adding to the computational complexity of the setup and analysis steps.

If the graph Laplacian eigenvalues are distinct, using the exact M-CSFB transform of Section III with $M = N$ and the filter endpoints $\tau_0 = 0$, $\tau_M = \lambda_{\text{max}} + 1$, and $\tau_m = \lambda_m + \frac{m}{2}$ for $m = 1, 2, \ldots, N - 1$ yields exactly the graph Fourier transform. Thus, the proposed transform can be seen as a fast version of the graph Fourier transform with a coarser resolution in the spectral domain. This resolution is controlled by the number of spectral bands; as we add more bands for finer resolution in the spectral domain, we need higher degree polynomials to accurately approximate the filters, which in turn slows down the computations and may result in atoms whose energy is more spread in the vertex domain. We have seen that for a reasonably small number of bands compared to the number of vertices $N$, a low degree polynomial approximation suffices and the atoms are jointly localized in the vertex and graph spectral domains.

On the other hand, the atoms of the proposed transform can also be viewed as a subset of the atoms of a spectral graph wavelet transform [10], albeit with a different set of filters. Both transforms yield atoms of the form $\tilde{h}_m(\mathcal{L})\delta_i$, but the spectral graph wavelet transform includes every vertex $i$ as a center vertex for every scale $m$.

As with the classical wavelet construction, it is possible to iterate the filter bank on the output from the lowpass channel. This could be beneficial, for example, in the case that we want to visualize the graph signal at different resolutions on a sequence of coarser and coarser graphs. An interesting question for future work is how iterating the filter bank with fewer channels at each step compares to a single filter bank with more channels supported on a smaller spectral intervals.

### Table I

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Bunny</td>
<td>0.1</td>
<td>0.01</td>
<td>5.4e-30</td>
<td>9.8</td>
<td>0.02</td>
<td>2.5e-29</td>
<td>295.7</td>
<td>0.08</td>
<td>1.4e-28</td>
<td>8544.8</td>
<td>0.6</td>
<td>4.5e-28</td>
</tr>
<tr>
<td>Andrianov Net25 Graph</td>
<td>1</td>
<td>0.19</td>
<td>10.8e-30</td>
<td>280.4</td>
<td>1.1</td>
<td>7.8e-23</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Community Graph</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temperatures</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### VIII. Appendix

**Proof of Proposition 7** We assume without loss of generality that $T = \{0, 1, 2, \ldots, k - 1\}$. Suppose first that the set $S$ is a uniqueness set for $\text{col}(U_T^+$), but $S^\perp$ is not a uniqueness set for $\text{col}(U_T^-)$. Then by Lemma 1 the matrix

\[
A = \begin{bmatrix}
\mathbf{u}_0 & \mathbf{u}_1 & \cdots & \mathbf{u}_{k-1} & \delta S_1 & \delta S_2 & \cdots & \delta S_{N-k}
\end{bmatrix}
\]

has full rank, and the matrix

\[
B = \begin{bmatrix}
\mathbf{u}_0 & \mathbf{u}_{k+1} & \cdots & \mathbf{u}_{N-1} & \delta S_1 & \delta S_2 & \cdots & \delta S_k
\end{bmatrix}
\]

is singular, implying

\[
\text{span}(\mathbf{u}_k, \mathbf{u}_{k+1}, \ldots, \mathbf{u}_{N-1}, \delta S_1, \delta S_2, \ldots, \delta S_k) \neq \mathbb{R}^N. \tag{23}
\]

Since $\dim(\text{span}(\mathbf{u}_k, \mathbf{u}_{k+1}, \ldots, \mathbf{u}_{N-1})) = N - k$ and $\dim(\text{span}(\delta S_1, \delta S_2, \ldots, \delta S_k)) = k$, equation (23) implies that there must exist a vector $\mathbf{x}$ such that $\mathbf{x}^\top \text{span}(\mathbf{u}_k, \mathbf{u}_{k+1}, \ldots, \mathbf{u}_{N-1})$ and $\mathbf{x}$ is orthogonal to $\delta S_1, \delta S_2, \ldots, \delta S_k$. Yet, $\mathbf{x} \in \text{col}(U_T^-)$ implies $\mathbf{x}$ is orthogonal to $\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{k-1}$, and, similarly, $\mathbf{x}$ is orthogonal to $\delta S_1, \delta S_2, \ldots, \delta S_{N-k}$. In matrix notation, we have $A^\top \mathbf{x} = 0$, so $A^\top$ has a non-trivial null space, and thus the square matrix $A$ is not full rank and $S$ is not a uniqueness set for $\text{col}(U_T^-)$, a contradiction. We conclude that if $A$ is full rank, then $B$ must be full rank and $S^\perp$ is a uniqueness set for $\text{col}(U_T^-)$, completing the proof of sufficiency. Necessity follows from the same argument, with the roles of $A$ and $B$ interchanged.

### References


—, extendi...