Compact Support Biorthogonal Wavelet Filterbanks for Arbitrary Undirected Graphs

Sunil K. Narang*, *Student Member, IEEE,* and Antonio Ortega, *Fellow, IEEE*

Abstract

In our recent work, we proposed the design of perfect reconstruction orthogonal wavelet filterbanks, called *graph-QMF*, for arbitrary undirected weighted graphs. In that formulation we first designed "one-dimensional" two-channel filterbanks on bipartite graphs, and then extended them to "multi-dimensional" separable two-channel filterbanks for arbitrary graphs via a bipartite subgraph decomposition. We specifically designed wavelet filters based on the spectral decomposition of the graph, and stated necessary and sufficient conditions for a two-channel graph filter-bank on bipartite graphs to provide aliasing-cancellation, perfect reconstruction and orthogonal set of basis (orthogonality). While, the exact *graph-QMF* designs satisfy all the above conditions, they are not exactly k-hop localized on the graph. In this paper, we relax the condition of orthogonality to design a biorthogonal pair of graph-wavelets that can have compact spatial spread and still satisfy the perfect reconstruction conditions. The design is analogous to the standard Cohen-Daubechies-Feauveau's (CDF) construction of factorizing a maximally-flat Daubechies half-band filter. Preliminary results demonstrate that the proposed filterbanks can be useful for both standard signal processing applications as well as for signals defined on arbitrary graphs.

Note: Code examples from this paper are available at http://biron.usc.edu/wiki/index.php/Graph_Filterbanks

EDICS Category: DSP-WAVL, DSP-BANK, DSP-MULT, DSP-APPL, MLT

I. INTRODUCTION

A. Motivation

Graphs provide a flexible model for representing data in many domains such as networks, computer vision, and high dimensional data-clouds. The data on these graphs can be visualized as a finite collection of samples, leading to a *graph-signal*, which can be defined as the information attached to each node (scalar or vector values mapped to the set of vertices/links) of the graph. Examples include measured values by sensor network nodes or traffic

Sunil K. Narang is with the Signal & Image Processing Institute, Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, California, USA 90089 email: narang.sunil@gmail.com.

Antonio Ortega is with the Signal & Image Processing Institute, Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, California, USA 90089 email: antonio.ortega@sipi.usc.edu.

measurements on the links of an Internet graph. The formulation of datasets as graph-signals has been subject to a lot of study recently, especially for the purpose of analysis, compression and storage. Major challenges are posed by the size of these datasets, making them difficult to visualize, process or analyze. This has led to a recent interest in extending wavelet techniques to signals defined on graphs. These techniques can provide multiresolution representations, so that smaller graphs with smooth approximations of the original graph-signals can be obtained and used for processing. Moreover, these signal representations can be used to develop local analysis tools, so that a graph-signal can be processed "locally" around a node or vertex, using data from a small neighborhood of nodes around the given node.

The trade-off between spatial and frequency localization is fundamental in the study of wavelet transforms for regular signals. This trade-off is being studied for graph signals as well [1], [2], [3]. We can say that a graph signal, for example an elementary basis function in a wavelet representation, is localized in the *vertex domain* if most of the signal energy is concentrated in a k-*hop localized* neighborhood around a vertex, so that the degree of localization will depend on how large k is. Similarly, a graph signal can be said to be localized in the *spectral domain*, defined in terms of the eigenvalues and eigenvectors of the *graph Laplacian matrix*, if the projection of the signal onto these eigenvectors has most of its energy concentrated in a band of graph-frequencies around a center frequency.

There has been a significant recent interest in the extension of wavelet transforms to graph signals, including wavelets on unweighted graphs for analyzing computer network traffic [4], diffusion wavelets and diffusion wavelet packets [5], [6], [7], the "top-down" wavelet construction of [8], graph dependent basis functions for sensor network graphs [9], lifting based wavelets on graphs [10], [11], [12], multiscale wavelets on balanced trees [13], spectral graph wavelets [1], and our recent work on graph wavelet filterbanks [2].

In this paper, we focus on the problem of designing wavelet transforms that are *invertible, compactly supported on the graph and critically sampled (CS)*. Critical sampling, i.e., the number of wavelet coefficients generated by the transform is equal to the number of vertices in the graph, is important in order to achieve a compact representation (e.g., for compression). Also, in a critically sampled transform a subset of vertices will include low frequency information. This leads to a natural approximation of the original graph by a smaller graph (containing only those vertices and their corresponding graph signal).

Of the above mentioned approaches, only the lifting based approaches [10], [11], the wavelets on balanced trees [13] and the graph wavelet filterbanks [2] achieve critical sampling, although in all cases there are some restrictions on the kinds of graphs on which the transforms can be defined.

In terms of localization, the lifting-based approaches [10], [11] are compactly supported in the graph, i.e., their basis functions are *strictly* k-hop localized the vertex domain, which implies that the output at each vertex n can

be computed exactly from the data at the vertex n and a k-hop neighborhood around it. Instead, in our filterbank-based approach [2] basis functions had good vertex domain localization but were not compactly supported. Compact support could be achieved by approximating their spectral response by a polynomial, but this comes at the expense of introducing a small reconstruction error (i.e., there was no longer perfect reconstruction).

The main contribution in this paper is to extend the filterbank approach to design graph wavelets of [2] so as to achieve compact support. Note that the lifting based techniques [10], [11] are also compactly supported but they are vertex domain designs for which it is difficult to control the quality (e.g., localization) of their spectral representation. Instead, in the work presented here filters are designed in the spectral domain while guaranteeing compact vertex domain support, so that we can directly control explicitly the trade-off between localization in the vertex domain and the spectral domain.

As in [2], the building blocks our design are *two channel wavelet filterbanks on bipartite graphs*, which provide a decomposition of any graph-signal into a lowpass (smooth) graph-signal, and a highpass (detail) graph-signal. For arbitrary graphs the filterbanks can be extended in two ways: a) by implementing the proposed wavelet filterbanks on a bipartite graph approximation of the original graph, which provides a "one-dimensional" analysis, or b) by decomposing the graph into multiple link-disjoint bipartite subgraphs, and applying the proposed filterbanks iteratively on each the subgraphs (or on some of them), leading to a "multi-dimensional" analysis [2].

In [2] we showed that downsampling/upsampling operations in bipartite graphs lead to a *spectral folding* phenomenon, which is analogous to aliasing in regular signal domain. We utilized this property to propose two channel critically sampled wavelet filterbanks, called *graph-QMF*, on arbitrary undirected weighted graphs. We specifically designed wavelet filters based on the spectral decomposition of the graph, and stated necessary and sufficient conditions for a two-channel graph filterbank on bipartite graphs to provide aliasing-cancellation, perfect reconstruction and orthogonal set of basis (orthogonality). While the exact *graph-QMF* designs satisfy all the above conditions, they are not compactly supported on the graph. In order to design compactly supported graph-QMF transforms, we performed a Chebychev polynomial approximation of the exact filters in the spectral domain, and this incurred error in the reconstruction of the signal, and loss of orthogonality. Here, we propose an alternative to graph-QMF design where we relax the conditions of orthogonality, and design a biorthogonal pair of graph-wavelets, which we call *graphBior*. These new designs lead to a representation that is exactly k-hop localized, and still satisfies the perfect reconstruction conditions. This design is analogous to the standard Cohen-Daubechies-Feauveau's (CDF) [14] construction to obtain maximally half-band filters. Even though these filterbanks are not orthogonal, we show that they can be designed to nearly preserve energy. In particular, we compute expressions for Riesz bounds of the filterbanks, and choose graph-wavelets with the maximum ratio of lower and upper Riesz bounds.

We will show that it is possible to design these filterbanks based on both the normalized Laplacian and the random-walk Laplacian, leading to *nonzeroDC graphBior* and *zeroDC graphBior* designs, respectively. As the name suggests, the *zeroDC* design has the advantage of leading to highpass operators with a zero response for the *all constant* signal, which will be useful in applications in Euclidean space (e.g., when applying graph wavelets to regular domain signals such as images). Instead, in the *nonzeroDC* design the DC frequency corresponds to a signal where the value at each node depends on its degree. Finally, because our designs are biorthogonal, the norms of highpass and lowpass are not equal. For applications where a normalization is required, we propose unity gain compensation (GC) techniques for both types of designs. A comparison of our proposed design vis-a-vis existing transforms is shown in Table I.

Method	DC	CS	PR	Comp	OE	GS
Wang & Ramchandran [9]	No	No	Yes	Yes	No	No
Crovella & Kolaczyk [4]	Yes	No	No	Yes	No	No
Lifting Scheme [10]	Yes	Yes	Yes	Yes	No	Yes
Wavelets on balanced trees [13]	Yes	Yes	Yes	Yes	No	Yes
Diffusion Wavelets [5]	Yes	No	Yes	Yes	Yes	No
Spectral Wavelets [1]	Yes	No	Yes	Yes	No	No
graph-QMF filterbanks (exact) [2]	No	Yes	Yes	No	Yes	No
graph-QMF filterbanks (approx.) [2]	No	Yes	No[1]	Yes	No[1]	No
nonzeroDC graphBior filterbanks	**No**	**Yes**	**Yes**	**Yes**	**No**	**No**
zeroDC graphBior filterbanks	**Yes**	**Yes**	**Yes**	**Yes**	**No**	**No**

TABLE I: Comparison of graph wavelet designs in terms of key properties: zero highpass response for constant graph-signal (DC), critical sampling (CS), perfect reconstruction (PR), compact support (Comp), orthogonal expansion (OE), requires graph simplification (GS).

The outline of the rest of the paper is as follows: in Section II, we introduce some notations, a general formulation of two channel wavelet filterbank on graphs and the graph-QMF filterbanks proposed in [2], which are orthogonal and perfect reconstruction. In Section III, we describe proposed nonzeroDC graphBior filterbanks on graphs, and in Section IV we design and describe the properties of zeroDC graphBior filterbanks. In Section V, we describe extension of proposed filterbanks to arbitrary graphs via bipartite subgraph decomposition, and multiresolution implementation. In Section VI, we conduct some experiments to demonstrate the properties and applications of the proposed filterbanks. Section VII concludes our paper.

II. PRELIMINARIES

A graph can be denoted as $G = (\mathcal{V}, E)$ with vertices (or nodes) in set $\mathcal{V}$ and links as tuples (i, j) in E. The graphs considered in this work are undirected graphs without self-loops and without multiple links between nodes. The links can only have positive weights. The size of the graph $N = |\mathcal{V}|$ is the number of nodes and the geodesic

---

[1]The exact Graph-QMF solutions are perfect reconstruction and orthogonal, but they are not compact support. Localization is achieved with a matrix polynomial approximation of the original filters, which incur some loss of orthogonality and reconstruction error, which can be arbitrarily reduced by increasing the degree of approximation.

distance metric is given as $d_{\mathcal{G}}(i,j)$, which represents the sum of link weights along the shortest path between nodes i and j, and is considered infinite if i and j are disconnected. Define $\mathcal{N}_{j,n}$ as the j-hop neighborhood of node n, i.e., $\mathcal{N}_{j,n} = \{m \in \mathcal{V}, d_{\mathcal{G}}(i,j) \leq j\}$. Denote the identity matrix as $\mathbf{I}$, and let δ_n be an impulse function, i.e., $\delta_n(n) = 1$ and $\delta_n(m) = 0$ for all $m \neq n$. Define $< \mathbf{f}, \ \mathbf{g} >= \mathbf{f}^{\top}\mathbf{g}$ as the inner-product of vector $\mathbf{f}$ and $\mathbf{g}$, where $(.)^{\top}$ is the transpose operator. Define $\mathbf{A} = [w_{ij}]$, the adjacency matrix of the graph, d_i the degree (sum of link-weights) of node i, and $\mathbf{D} = diag\{d_i\}_{i=1,2,...N}$, the diagonal degree matrix of graph, so that $\mathbf{L} = \mathbf{D} - \mathbf{A}$ is the *unnormalized Laplacian matrix* of the graph.

A. Graph vertex domain

We define a signal $f : \mathcal{V} \to \mathbb{R}$ on a graph as a set of scalars, where each scalar is assigned to one of the vertices of the graph. This can be extended to vector values at each vertex. Further, a graph based transform is defined as a linear transform $\mathbf{T} : \mathbb{R}^N \to \mathbb{R}^M$ applied in the vertex domain, such that the operation at each node n is a linear combination of the value of the graph-signal $f(n)$ at the node n and the values $f(m)$ on nearby nodes $m \in \mathcal{N}_{j,n}$, i.e.,

$$y(n) =< \mathbf{T}(n,.), \ \ \mathbf{f} >= T(n,n)f(n) + \sum_{m \in \mathcal{N}_{j,n}} T(n,m)f(m), \tag{1}$$

where $\mathbf{T}(n,.)$ is the nth row of the transform $\mathbf{T}$. In analogy to the 1-D regular case, we would sometimes refer to graph-transforms as graph-filters, and $\mathbf{T}(n,.)$ for $n = 1, 2, ...N$ as the *impulse response* of the transform $\mathbf{T}$ at the nth node. Note that due to the irregularity of links, the impulse response of a graph filter varies from one vertex to the other. A desirable feature of graph filters is *spatial localization*, which typically means that the energy of each impulse response (i.e., each row) of the graph filter is concentrated in a local region around a node. In this paper, we use the definition proposed in [3] to define the spatial spread of any signal $\mathbf{f}$ on a graph G as:

$$\Delta_G^2(\mathbf{f}) := \inf_i \frac{1}{||\mathbf{f}||_2^2} \sum_{j \in \mathcal{V}} [d_{\mathcal{G}}(i,j)]^2 [f(j)]^2. \tag{2}$$

Here, $\{[f(j)]^2/||\mathbf{f}||^2\}_{j=1,2,...,N}$ can be interpreted as a probability mass function (pmf) of signal $\mathbf{f}$, and $\Delta_G^2(\mathbf{f})$ is the variance of the geodesic distance function $d_{\mathcal{G}}(i,.) : \mathcal{V} \to \mathbb{R}$ at node i, in terms of this spatial pmf. Thus, $\Delta_G^2(\mathbf{T}(n,.))$ should be small for all $n = 1, 2, ...N$ for good spatial localization. In our analysis, we compute the spatial spread of a graph transform $\mathbf{T}$ to be the average of the spatial spread (2) of impulse responses over all vertices, i.e.,

$$\Delta_G^2(\mathbf{T}) := \frac{1}{N} \sum_{n=1}^{N} \Delta_G^2(\mathbf{T}(n,.)) \tag{3}$$

B. Graph spectral domain

As in our previous design [2], we use the symmetric normalized Laplacian matrix $\mathcal{L} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ to define spectral properties of the graph. In this paper, we use the same Laplacian matrix $\mathcal{L}$ to design nonzeroDC filterbanks. Because $\mathcal{L}$ is a real symmetric matrix, it has a complete set of orthonormal eigenvectors, which we denote by $\{\mathbf{u}_l\}_{l=0,1,2,...,N-1}$. These eigenvectors have associated real, non-negative eigenvalues $\{\lambda_l\}_{l=0,1,2,...,N-1}$ satisfying $\mathcal{L}\mathbf{u}_l = \lambda_l\mathbf{u}_l$, for $l = 0, 1, 2, ..., N-1$, and ordered as: $\lambda_0 \leq \lambda_1 \leq ...\lambda_{N-1}$. In our analysis, we assume the graph to be connected [2]. Zero appears as a unique minimum eigenvalue of the graph and the maximum eigenvalue is always less than or equal to 2, with equality if and only if the graph is a bipartite graph [15, Section 2].

We denote the *spectrum of the graph* by $\sigma(\mathcal{L}) := \{\lambda_0, \lambda_1, ...\lambda_{N-1}\}$. A graph signal $\mathbf{f}$ is represented in the spectral domain as its projection onto the eigenvectors, denoted as : $\hat{f}(\lambda_l) =< \mathbf{f}, \mathbf{u}_l >$. An eigenspace V_λ is defined as the space spanned by eigenvectors of $\mathcal{L}$ associated with eigenvalue λ, and the *eigenspace projection matrix* is defined as:

$$\mathbf{P}_\lambda := \sum_{\lambda_l = \lambda} \mathbf{u}_l\mathbf{u}_l^\top,$$

where $\mathbf{u}_l^\top$ is the transpose of eigenvector $\mathbf{u}_l$. The eigenspace projection matrices are idempotent and $\mathbf{P}_\lambda$ and $\mathbf{P}_\gamma$ are orthogonal if λ and γ are distinct eigenvalues of the Laplacian matrix, i.e.,

$$\mathbf{P}_\lambda\mathbf{P}_\gamma = \delta(\lambda - \gamma)\mathbf{P}_\lambda, \tag{4}$$

where $\delta(\lambda)$ is the Kronecker delta function. Further the sum of all eigenspace projection matrices for any graphs is an identity matrix. Closely related to the symmetric Laplacian matrix $\mathcal{L}$ is the random-walk graph Laplacian, which is defined as $\mathcal{L}_r := \mathbf{D}^{-1}\mathbf{L}$. We use $\mathcal{L}_r$ to design zeroDC filterbanks. Note that $\mathcal{L}_r$ has the same set of eigenvalues as $\mathcal{L}$, and if $\mathbf{u}_l$ is an eigenvector of $\mathcal{L}$ associated with λ_l, then $\mathbf{D}^{-1}\mathbf{u}_l$ is an eigenvector of $\mathcal{L}_r$ associated with the eigenvalue λ_l. Similar to (2), the spectral spread of a graph signal c an be defined as:

$$\Delta_\sigma^2(\mathbf{f}) := \min_{\mu \in \mathbb{R}_+} \left\{ \frac{1}{||\mathbf{f}||_2^2} \sum_{\lambda \in \sigma(\mathcal{L})} [\lambda - \mu]^2 \left[\hat{f}(\lambda)\right]^2 \right\}, \tag{5}$$

where $\{[\hat{f}(\lambda)]^2/||\mathbf{f}||_2^2\}_{\lambda=\lambda_0,\lambda_1,...,\lambda_{N-1}}$ is the pmf of $\mathbf{f}$ across the spectrum of the Laplacian matrix, and μ is the mean of λ with respect to this pmf. $\Delta_\sigma^2(\mathbf{f})$ computes the variance of λ with respect to the spectral pmf function. Thus, for a signal to have good spectral localization, the value of $\Delta_\sigma^2(\mathbf{T}(n,.))$ should be small for all vertices. In our analysis, we compute the spectral response of any transform $\mathbf{T}$ as the average spectral response of impulse

---

[2]For graphs with multiple separate connected components, we analyze each component as a separate graph.

responses at all vertices, i.e.,

$$|\hat{\mathbf{T}}(\lambda)|^2 := \frac{1}{N} \sum_{i=1}^{N} |\hat{\mathbf{T}}(n,.)(\lambda)|^2, \tag{6}$$

and use $|\hat{\mathbf{T}}(\lambda)|^2$ as the spectral pmf function to compute the spectral spread of $\mathbf{T}$[3].

C. Spectral graph filters

For designing compact support wavelet filters on graphs we use the same approach as in [2], and define analysis wavelet filters $\mathbf{H}_i$ and $\mathbf{G}_i$ in terms of spectral kernels $\hat{h}_i(\lambda)$ and $\hat{g}_i(\lambda)$ for $i = 0, 1$ respectively. The corresponding transform matrices are represented as:

$$\begin{aligned} \mathbf{H}_i = \hat{h}_i(\mathcal{L}) = \sum_{\lambda \in \sigma(G)} \hat{h}_i(\lambda)\mathbf{P}_\lambda, \\ \mathbf{G}_i = \hat{g}_i(\mathcal{L}) = \sum_{\lambda \in \sigma(G)} \hat{g}_i(\lambda)\mathbf{P}_\lambda. \end{aligned} \tag{7}$$

These filters have the following interpretation: the output of a spectral filter with kernel $\hat{h}(\lambda)$ can be expanded as $\mathbf{f}_H = \mathbf{H}\mathbf{f} = \sum_{\lambda \in \sigma(G)} \hat{h}_i(\lambda) \mathbf{P}_\lambda \mathbf{f}$, where $\mathbf{f}_\lambda = \mathbf{P}_\lambda \mathbf{f}$ is the component of input signal $\mathbf{f}$ in the λ-eigenspace. Thus, filter $\mathbf{H}$ either attenuates or enhances different harmonic components of the input signal depending upon the magnitude of $\hat{h}(\lambda)$. Therefore, we will also refer to $\hat{h}(\lambda)$ as the *spectral response* of filter $\mathbf{H}$. For a general kernel function, the filtering operations corresponding to $\mathbf{H}_i$ and $\mathbf{G}_i$ may not have compact support, and would require a full spectral decomposition of Laplacian matrix. However it has been shown in [1], that the spectral response can be approximated as a polynomial of degree K, and the corresponding filters can be computed iteratively with K one-hop operations at each node. Further, any graph filters with a K degree polynomial spectral response are exactly K-*hop localized* (compact support) [1, Lemma 5.2], and can be efficiently computed without diagonalizing the Laplacian matrix. The computational complexity of the filtering operations in the polynomial case, reduces to $\mathcal{O}(K|E|)$ for degree K and $|E|$ number of links in the graph. Thus, the degree K in case of polynomial spectral response can be interpreted as the length of the corresponding spectral filters [4].

D. Spectral wavelet filterbanks

In [2], we described the construction of a two-channel wavelet filterbank on a bipartite graph $\mathcal{B} = (L, H, E)$[5], characterized by filtering operations $\{\mathbf{H}_i, \mathbf{G}_i\}_{i=0,1}$ and a function $\beta(n)$, which provides a decomposition of graph-

---

[3] Note that the definitions of spread presented here are heuristic and do not have a well-understood theoretical background. Another definition of spectral spread in graphs is given in [3]. If the graph is not regular, the choice of which Laplacian matrix ($\mathcal{L}$ or $\tilde{\mathcal{L}}$) to use for computing spectral spreads also affects the results. The purpose of these definitions and the subsequent examples is to show that a trade-off exists between spatial and spectral localization in graph wavelets.

[4] Note that, having a polynomial spectral response for compact support is necessary only in case of spectral graph filters. There can be non-spectral graph-filters, for example, graph wavelets proposed in [4], that have compact support without being a polynomial in the spectral domain.

[5] A bipartite graph $G = (L, H, E)$ is a graph whose vertices can be divided into two disjoint sets L and H, such that every link connects a vertex in L to one in H.

signal $\mathbf{f}$ into a lowpass (approximation) graph-signal $\mathbf{f}_{low}$ and a highpass (details) graph-signal component $\mathbf{f}_{high}$. The
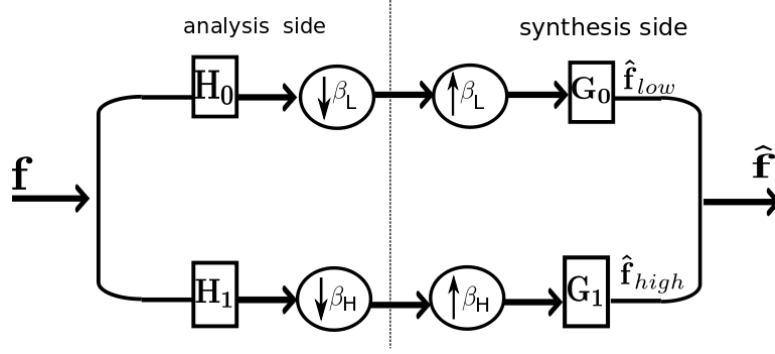

Fig. 1: Block diagram of a two-channel wavelet filterbank on graph.

transforms $\mathbf{H}_i, \mathbf{G}_i$ for $i = 0, 1$ of the two channels are *graph transforms* with spectral kernels h_i and g_i respectively, as given in (7). In the analysis side of the filterbank, the input signal is first operated upon by transform $\mathbf{H}_0$ in the lowpass channel and $\mathbf{H}_1$ in the high pass channel. A subsequent downsampling upsampling (DU) operation, discards and replaces with zeros the output coefficients on the set H in the lowpass channel and on the set L in the highpass channel. Since L and H are disjoint and complementary subsets of vertex set $\mathcal{V}$, the retained set of output coefficients is critically sampled. Algebraically, the DU operation can be represented with a function $\beta(n)$, such that $\beta(n) = 1$, if node $n \in L$ and $\beta(n) = -1$ if node $n \in H$. Thus, the DU operation in the lowpass channel is given as $\frac{1}{2}(1 + \beta(n))$ and in the highpass channel $\frac{1}{2}(1 - \beta(n))$. It can also be written in the matrix form as $\frac{1}{2}(\mathbf{I} + \mathbf{J}_\beta)$ for lowpass channel and $\frac{1}{2}(\mathbf{I} - \mathbf{J}_\beta)$ for highpass channel, where $\mathbf{J}_\beta = diag\{\beta\}$ is a diagonal matrix. The output signal $\mathbf{f}_{du}$ of the DU operation in the two channels is expressed in Figure 2.
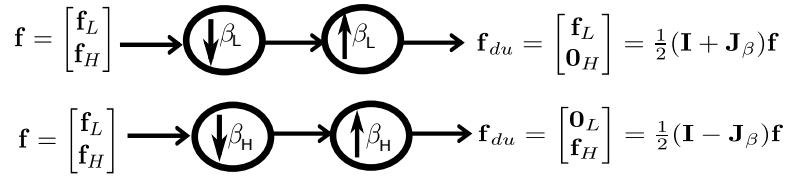

Fig. 2: DU operations in the proposed two-channel filterbank.

The overall transfer matrix of the two channel filterbank can be written as:

$$
\begin{aligned}
\mathbf{T} & = \frac{1}{2}\mathbf{G}_0(\mathbf{I} + \mathbf{J}_\beta)\mathbf{H}_0 + \frac{1}{2}\mathbf{G}_1(\mathbf{I} - \mathbf{J}_\beta)\mathbf{H}_1 \\
& = \underbrace{\frac{1}{2}(\mathbf{G}_0\mathbf{H}_0 + \mathbf{G}_1\mathbf{H}_1)}_{\mathbf{T}_{eq}} + \underbrace{\frac{1}{2}(\mathbf{G}_0\mathbf{J}_\beta\mathbf{H}_0 - \mathbf{G}_1\mathbf{J}_\beta\mathbf{H}_1)}_{\mathbf{T}_{alias}},
\end{aligned}
\tag{8}
$$

where $\mathbf{T}_{eq}$ is the transfer function of the filterbank without the *DU* operation and $\mathbf{T}_{alias}$ arises primarily due to the *DU* operations. Using the above formulation, we derived the following results[6]:

[6]see [2] for proofs and details.

*Theorem 2.1 (***Spectral folding phenomenon [2, Prop.** 1]**)**: Given a bipartite graph $\mathcal{B} = (L, H, E)$ with Laplacian matrix $\mathcal{L}$ and with the binary function β defined as above, if $\mathbf{u}_\lambda$ is the eigenvector of a unique eigenvalue λ of graph $\mathcal{B}$ then

$$\mathbf{J}_\beta \mathbf{u}_\lambda = \pm \mathbf{u}_{2-\lambda}. \tag{9}$$

The ambiguity of $\pm$ sign appears in (9) since both $\mathbf{u}_\lambda$ and $-\mathbf{u}_\lambda$ can be eigenvectors of $\mathcal{B}$ for eigenvalue λ. Further, if the eigenvalue λ appears with multiplicity greater than 1, and if $\mathbf{P}_\lambda$ is the projection matrix corresponding to the eigenspace V_λ, then

$$\mathbf{J}_\beta \mathbf{P}_\lambda = \mathbf{P}_{2-\lambda} \mathbf{J}_\beta. \tag{10}$$

According to (9) and (10) the eigenvector (or eigenspace) of eigenvalue λ in a bipartite graph changes to the eigenvector (or eigenspace) of eigenvalue $2 - \lambda$ after multiplying with $\mathbf{J}_\beta$. In other words, the eigenspace folds across the imaginary axis at $\lambda = 1$, hence we call this a *spectral folding phenomenon*[7].

*Theorem 2.2 (***Perfect reconstruction property [2, Sec. III.B]**)**: Given a bipartite graph $\mathcal{B} = (L, H, E)$, the filtering operations $\{\mathbf{H}_i, \mathbf{G}_i\}$ and the binary function $\beta(n)$ as defined above, a necessary and sufficient condition for the *perfect reconstruction* in the two channel filterbanks is that for all λ in $\sigma(\mathcal{B})$,

$$\hat{g}_0(\lambda)\hat{h}_0(\lambda) + \hat{g}_1(\lambda)\hat{h}_1(\lambda) = 2,$$
$$\hat{g}_0(\lambda)\hat{h}_0(2 - \lambda) - \hat{g}_1(\lambda)\hat{h}_1(2 - \lambda) = 0. \tag{11}$$

The advantage of representing PR conditions as in (11) is that the filterbank can be designed in the spectral domain of the graph by designing spectral kernels which satisfy (11). These kernels can be designed as continuous functions of $\lambda \in [0\ 2]$, which obviates the need to evaluate (11) only at the spectrum $\sigma(\mathcal{B})$, thus making the design independent of the structure of the graph. The *graph-QMF* solution provided in [2], satisfies (11), and is also orthogonal. While there exist many exact graph-QMF solutions, we show in Section III-A (see last paragraph) that none of the solutions has compact support. Therefore, we shift our focus to biorthogonal solutions which are PR and have compact support.

III. NONZERODC GRAPHBIOR FILTERBANKS

Given the parameter k, in order to design k-hop localized filters to satisfy the perfect reconstruction conditions given in (11), we need to design four polynomial spectral kernels of degree k, namely lowpass analysis kernel

---

[7]Note that spectral folding phenomenon only occurs if the binary function $\beta(n)$ is defined on one of the natural partitions L or H of the bipartite graph $\mathcal{B} = (L, H, E)$, and not for any other partition.

$\hat{h}_0(\lambda)$, highpass analysis kernel $\hat{h}_1(\lambda)$, lowpass synthesis kernel $\hat{g}_0(\lambda)$, and highpass synthesis kernel $\hat{g}_1(\lambda)$. If we choose analysis and synthesis highpass kernels to be:

$$\begin{aligned} \hat{h}_1(\lambda) &= \hat{g}_0(2-\lambda) \\ \hat{g}_1(\lambda) &= \hat{h}_0(2-\lambda), \end{aligned} \tag{12}$$

then, (11) reduces to a single constraint for all eigenvalues, given as:

$$\hat{h}_0(\lambda)\hat{g}_0(\lambda) + \hat{h}_0(2-\lambda)\hat{g}_0(2-\lambda) = 2. \tag{13}$$

Further, define $\hat{p}(\lambda) = \hat{h}_0(\lambda)\hat{g}_0(\lambda)$, then (13) can be written as:

$$\hat{p}(\lambda) + \hat{p}(2-\lambda) = 2. \tag{14}$$

In our approach, we first design $\hat{h}_0(\lambda)$ and $\hat{g}_0(\lambda)$, and then $\hat{h}_1(\lambda)$ and $\hat{g}_1(\lambda)$ can be obtained using (12). Further, since $\hat{p}(\lambda)$ is the product of two lowpass kernels, it is also a lowpass kernel. Therefore, the objective is to design $\hat{p}(\lambda)$ as a polynomial *half-band kernel*[8], which satisfies (13), and then obtain kernels $\hat{h}_0(\lambda)$ and $\hat{h}_1(\lambda)$ via spectral factorization. This design is similar to the design proposed by Cohen-Daubechies-Feauveau [14] for finding a maximally flat pair of lowpass and highpass filters under the given length constraint, and then dividing up the residual factors between the two filters in a way that makes the basis function nearly orthogonal. The following result is useful in our analysis:

Proposition 1: If $\hat{h}_0(\lambda)$ and $\hat{g}_0(\lambda)$ are polynomial kernels, then any $\hat{p}(\lambda) = h_0(\lambda)g_0(\lambda)$, which satisfies (14) for all $\lambda \in [0\ 2]$, is an odd degree polynomial.

Proof: By changing the variable so that $\lambda = 1 + l$, we can write (14) as:

$$\hat{p}(1+l) + \hat{p}(1-l) = 2, \tag{15}$$

where $\hat{p}(1+l) = \hat{h}_0(1+l)\hat{g}_0(1+l)$ and $l \in [-1\ 1]$. If $\hat{h}_0(l)$ and $\hat{g}_0(l)$ are polynomial in l then the functions $\hat{p}(1+l)$ and $\hat{p}(1-l)$ are also polynomials in l, and can be expressed as:

$$\begin{aligned} \hat{p}(1+l) &= \sum_{k=0}^{K} c_k(l)^k, \\ \hat{p}(1-l) &= \sum_{k=0}^{K} c_k(-l)^k. \end{aligned} \tag{16}$$

---

[8]An ideal half band kernel $h(\lambda)$ in the spectral domain of a bipartite graph can be defined as a kernel with $h(\lambda) = 1$ for $\lambda \le 1$, and $h(\lambda) = 0$ otherwise.

Using (16) in (15), we get:

$$\hat{p}(1+l) + \hat{p}(1-l) = \sum_{k=0}^{K} c_k((l)^k + (-l)^k) = 2c_0 + \sum_{k=1}^{K/2} c_{2k}l^{2k}. \tag{17}$$

Thus $\hat{p}(1+l) + \hat{p}(1-l)$ is an even polynomial function of l. In order for both (15) and (17) to be true for all $l \in [-1\ 1]$, $c_0 = 1$ and all other even power coefficients c_n in the polynomial expansion of $\hat{p}(1+l)$ must be 0. Therefore, the solution $\hat{p}(1+l)$, expressed as:

$$\hat{p}(1+l) = 1 + \sum_{n=0}^{K} c_{2n+1}l^{2n+1}, \tag{18}$$

is an odd degree polynomial. Thus, ignoring the trivial case $\hat{p}(1+l) = 1$, the highest degree of $\hat{p}(1+l)$ and $\hat{p}(1-l)$ (and hence $\hat{p}(\lambda)$) is always odd. ∎

A. Designing half-band kernel $\hat{p}(\lambda)$

The following known results help us prove the existence of a polynomial $\hat{p}(\lambda)$ that satisfies (15), and obtain its spectral factorization:

Lemma 1 (Bezout's identity [16, prop. 3.13]): Given any two polynomials $a(l)$ and $b(l)$ of continuous variable l,

$$a(l)x(l) + b(l)y(l) = c(l), \tag{19}$$

has a solution $[x(l),\ y(l)]$, if and only if $gcd(a(l), b(l))$ divides $c(l)$, where $gcd(a(l), b(l))$ refers to the greatest common divisor of polynomials $a(l)$ and $b(l)$.

Theorem 3.1 (Complementary Filters [16, prop. 3.13]): Given a polynomial kernel $\hat{h}_0(l)$, there exists a complementary polynomial kernel $\hat{g}_0(l)$ which satisfies the perfect reconstruction relation in (13), if and only if $\hat{h}_0(1+l)$ and $\hat{h}_0(1-l)$ are coprime.

Proof: Let us denote $a(l) = h_0(1+l)$, $b(l) = h_0(1-l) = a(-l)$, and $c(l) = 2$. Then, (13) can be written in the same form as (19), i.e.,

$$a(l)x(l) + a(-l)y(l) = 2, \tag{20}$$

We first note that if a polynomial solution $[x(l),\ y(l)]$ of (20) exists, then

$$a(-l)x(-l) + a(l)y(-l) = 2, \tag{21}$$

also has a polynomial solution. Subsequently, combining (20) and (21) and choosing $\hat{g}_0(1+l) = 1/2(x(l)+y(-l))$, we find that

$$a(-l)g_0(1+l) + a(l)g_0(1-l) = 2, \tag{22}$$

also has a polynomial solution. However, based on Lemma 1, (20) has a polynomial solution *if and only if* $gcd(h_0(1+l), h_0(1-l))$ divides $c(l) = 2$, which is a prime number. This implies $gcd(h_0(1+l), h_0(1-l))$ is either 1 or 2 for all $l \in [-1 \; 1]$, which is true iff $\hat{h}_0(1+l)$ and $\hat{h}_0(1-l)$ do not have any common roots. This implies that $\hat{h}_0(1+l)$ and $\hat{h}_0(1-l)$ are coprime. ∎

Corollary 3.2 ([16, exercise. 3.12]): There is always a complementary filter for the polynomial kernel $(1+l)^k$, i.e.,

$$(1+l)^k R(l) + (1-l)^k R(-l) = 2 \tag{23}$$

always has a real polynomial solution $R(l)$ for $k \geq 0$.

Proof: Let us denote $a(l) = (1+l)^k$, $b(l) = (1-l)^k$, $x(l) = R(l)$, $y(l) = R(-l)$ and $c(l) = 2$. Then, (23) can be written in the same form as (19). Since $a(l)$ and $b(l)$, in this case are coprime, therefore $gcd(a(l), b(l)) = 1$ divides $c(l) = 2$. Hence, a polynomial $R(l)$, which satisfies (23) always exists. ∎

For a perfect reconstruction biorthogonal filterbank, we need to design a polynomial half-band kernel $\hat{p}(\lambda)$ that satisfies (13) for all $\lambda \in [0 \; 2]$, or equivalently $\hat{p}(l)$ that satisfies (15) for all $l \in [-1 \; 1]$. Following Daubechies' approach [14], we propose a *maximally-flat* design, in which we assign K roots to $\hat{p}(\lambda)$ at the lowest eigenvalue (i.e., at $\lambda = 0$). Subsequently, we select $\hat{p}(\lambda)$ to be the shortest length polynomial, which has K roots at $\lambda = 0$ and satisfies (15). This implies that $\hat{p}(1+l)$ has K roots at $l = -1$, and can be expanded as:

$$\hat{p}(1+l) = (1+l)^K \underbrace{\sum_{m=0}^{k} r_m l^m}_{R(l)}. \tag{24}$$

where $R(l)$ is the residual k degree polynomial of l. By Corollary 3.2, there always exist such a polynomial $R(l)$. On the other hand, Proposition 1 says that any $\hat{p}(1+l)$ that satisfies (15) has to be an odd-degree polynomial. Hence, $\hat{p}(1+l)$ can also be expanded as:

$$\hat{p}(1+l) = 1 + \sum_{n=0}^{M} c_{2n+1} l^{2n+1}. \tag{25}$$

for a given M. Comparing (24) and (25), we get:

$$(1+l)^K \sum_{m=0}^{k} r_m l^m = 1 + \sum_{n=0}^{M} c_{2n+1} l^{2n+1}. \tag{26}$$

Comparing the constant terms in the left and right side of (26), we get $r_0 = 1$. Further, comparing the highest powers on both sides of (26) we get:

$$M = \frac{K + k - 1}{2} \tag{27}$$

Further, the right side in (26) has M constraints $c_{2n} = 0$ for $n = \{1, 2, ...K\}$, and the left side in (26) has k unknowns r_m for $m = \{1, 2, ...k\}$. In order to get a unique $\hat{p}(1+l)$ that satisfies (15), we must have equal number of unknowns and constraints, i.e,

$$M = k = \frac{K + k - 1}{2} \quad \Rightarrow \quad M = K - 1. \tag{28}$$

Thus, (26) can be written as:

$$(1 + l)^K (1 + \sum_{m=1}^{K-1} r_m l^m) = 1 + \sum_{n=0}^{K-1} c_{2n+1} l^{2n+1}, \tag{29}$$

and $K - 1$ unknowns can be found uniquely, by solving a linear system of $K - 1$ equations. Note that given K, the length of $\hat{p}(l)$ (i.e, highest degree) is $K + M = 2K - 1$. As an example, we design $\hat{p}(\lambda)$ with $K = 2$ zeros at $l = 0$. In this case $\hat{p}(1 + l)$ can be written as:

$$\hat{p}(1 + l) = (1 + l)^2 (1 + r_1 l) = 1 + (r_1 + 2)l + (1 + 2r_1)l^2 + r_1 l^3$$

Since $\hat{p}(1+l)$ is an odd polynomial, the term corresponding to l^2 is zero, i.e., $1 + 2r_1 = 0$ or $r_1 = -1/2$. Therefore, $\hat{p}(1 + l)$ is given as:

$$\hat{p}(1 + l) \quad = \quad (1 + l)^2 (1 - \frac{1}{2}l), \tag{30}$$

which implies that:

$$p(\lambda) \quad = \quad \frac{1}{2} \lambda^2 (3 - \lambda). \tag{31}$$

In Figure 3, we plot $\hat{p}(\lambda)$ for various values of K, and it can be seen that by increasing K, we get a $\hat{p}(\lambda)$ representing a better approximation to the ideal halfband filter.

Note that the graph-QMF designs in [2] were based on selecting $\hat{g}_0(\lambda) = h_0(\lambda)$ hence $p_{QMF}(\lambda) = h_0^2(\lambda)$. Thus, if $\hat{h}_0(\lambda)$ is a polynomial kernel then $p_{QMF}(\lambda)$ is the square of a polynomial, and therefore should have an even degree. However, as proven in Proposition 1, any polynomial $\hat{p}(\lambda)$ which satisfies (13) is an odd-degree polynomial. **Therefore, $\hat{h}_0(\lambda)$ in the graph-QMF designs, cannot be an exact polynomial.**

B. Spectral factorization of half-band kernel $\hat{p}(\lambda)$

Once we obtain $\hat{p}(\lambda)$ by using the above mentioned design, we need to factor it into filter kernels $\hat{h}_0(\lambda)$ and $\hat{g}_0(\lambda)$. Since $\hat{p}(\lambda)$ is a real polynomial of odd degree, it has at least one real root and all the complex roots occur in conjugate pairs. Since we want the two kernels to be polynomials with real coefficients, each complex conjugate root pair of $\hat{p}(\lambda)$ should be assigned together to either $\hat{h}_0(\lambda)$ or $\hat{g}_0(\lambda)$. While any such factorization would lead to
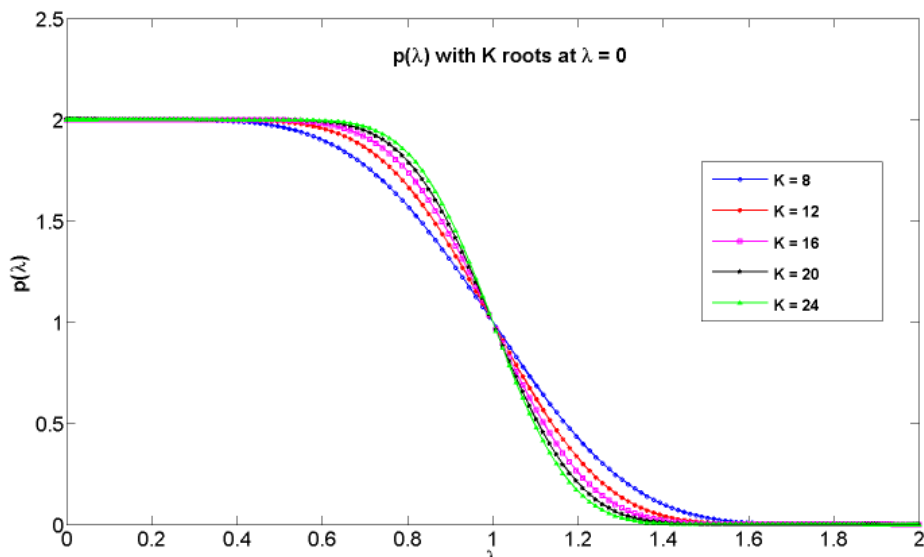
Fig. 3: The spectral distribution of $\hat{p}(\lambda)$ with K zeros at $\lambda = 0$

perfect reconstruction biorthogonal filterbanks, of particular interest is the design of filterbanks that are as close to orthogonal as possible. For this, we define a criterion based on energy preservation. In particular, we compute the Riesz bounds of analysis wavelet transform $\mathbf{T}_a$, which are the tightest lower and upper bounds, $A > 0$ and $B < \infty$, of $||\mathbf{T}_a\mathbf{f}||_2$, for any graph-signal $\mathbf{f}$ with $||\mathbf{f}||_2 = 1$. For near-orthogonality, we require $A \approx B \approx 1$. The bounds A and B can be computed as the minimum and maximum singular values of the overall analysis side transform $\mathbf{T}_a$ of the two channel filterbank. The transform matrix $\mathbf{T}_a$ consists of some rows of lowpass transform $\mathbf{H}_0$ (those corresponding to the L set) and some rows of highpass transform $\mathbf{H}_1$ (those corresponding to the H set) and can be written as:

$$\mathbf{T}_a = \frac{1}{2}(\mathbf{I} + \mathbf{J}_\beta)\mathbf{H}_0 + \frac{1}{2}(\mathbf{I} - \mathbf{J}_\beta)\mathbf{H}_1 \tag{32}$$

The singular values of $\mathbf{T}_a$ are also the square roots of eigenvalues of $\mathbf{T}_a^\top \mathbf{T}_a$, which can be expanded as (see [2] for details):

$$\begin{aligned}
\mathbf{T}_a^\top \mathbf{T}_a &= 1/2 \sum_{\lambda \in \sigma(\mathcal{B})} \underbrace{(h_0^2(\lambda) + h_1^2(\lambda))}_{C(\lambda)} \mathbf{P}_\lambda \\
&+ 1/2 \sum_{\lambda \in \sigma(\mathcal{B})} \underbrace{(h_1(\lambda)h_1(2 - \lambda) - h_0(\lambda)h_0(2 - \lambda))}_{D(\lambda)} \mathbf{J}_\beta \mathbf{P}_\lambda,
\end{aligned} \tag{33}$$

where $\mathbf{J}_\beta$ is the diagonal matrix of binary function β. In (33), the term $D(\lambda)$ consists of product terms $\hat{h}_0(\lambda)\hat{h}_0(2-\lambda)$ and $\hat{h}_1(\lambda)\hat{h}_1(2 - \lambda)$, which are small for λ away from the transition band around 1 (since these are the products of a low pass and a high pass kernel). Further, in the transition band when λ is close to 1, the value of $D(\lambda) \approx \hat{h}_0^2(\lambda) - \hat{h}_1^2(\lambda)$ is very small compared to $C(\lambda) \approx \hat{h}_0^2(\lambda) + \hat{h}_1^2(\lambda)$. Therefore, we can ignore $D(\lambda)$ in comparison

to $C(\lambda)$, and (33) can be approximately reduced to:

$$\mathbf{T}_a^\top \mathbf{T}_a \approx 1/2 \sum_{\lambda \in \sigma(\mathcal{B})} \underbrace{(h_0^2(\lambda) + h_1^2(\lambda))}_{C(\lambda)} \mathbf{P}_\lambda \tag{34}$$

Thus, $\mathbf{T}_a^\top \mathbf{T}_a$ is a spectral transform with eigenvalues $1/2(\hat{h}_0^2(\lambda) + \hat{h}_1^2(\lambda))$ for $\lambda \in \sigma(\mathcal{B})$, and the Riesz Bounds can be given as:

$$
\begin{aligned}
A &= \sqrt{\inf_\lambda \frac{1}{2}(\hat{h}_0^2(\lambda) + \hat{h}_1^2(\lambda))} \\
B &= \sqrt{\sup_\lambda \frac{1}{2}(\hat{h}_0^2(\lambda) + \hat{h}_1^2(\lambda))}
\end{aligned}
\tag{35}
$$

We define Θ, as the measure of orthogonality, given as:

$$\Theta = 1 - \frac{|B - A|}{|B + A|}. \tag{36}$$

For orthogonal filterbanks $\Theta = 1$. We choose filters with least dissimilar lengths, and for near orthogonal designs, compute Θ for all such possible factorizations (there are $\binom{2K-1}{K}$ possible choices), and choose the factorization with the maximum absolute value of Θ. Note that the computation of Θ, and hence the choice of the best solution depends on the exact distribution of eigenvalues λ in the interval $[0\ 2]$, which in turn depends on the underlying graph. For a graph independent design, we approximate A^2 and B^2 as the lowest and highest values, respectively, of $1/2(h_0^2(\lambda) + h_1^2(\lambda))$ at 100 uniformly sampled points from the continuous region $[0\ 2]$, respectively, and use these approximations to compute Θ.

C. Unity gain compensation

In order to avoid unnecessary growth of dynamic range in the output, it is desirable to normalize the filterbanks, such that the *impulse responses* of lowpass and highpass filters have equal (ideally unity) gain. In the graph-QMF case [2], the orthogonality condition ensures that all filters have unity gain. However, this is not true for the proposed graphBior filterbanks. Similar *gain compensations* have also been proposed for biorthogonal DWT filterbanks (for example, see [17]). The JPEG2000 standard, for example, requires the lowpass filter to have unity response for the DC frequency ($\omega = 0$), and the highpass filter to have unity response for the Nyquist frequency ($\omega = \pi$). In this paper, we follow similar specifications, as given in JPEG2000, to normalize the gains of graphBior filters. In the bipartite graph case, $\omega = 0$ and $\omega = \pi$ correspond to the lowest ($\lambda = 0$) and the highest ($\lambda = 2$) magnitude eigenvalues, respectively. Thus, given a filter $\mathbf{H}$ with spectral kernel $h(\lambda)$, the gain factor of $\mathbf{H}$ is equal to $|1/h(0)|$, if $\mathbf{H}$ is a lowpass filter, and is equal to $|1/h(2)|$, if $\mathbf{H}$ is a highpass filter. In the filterbank implementation, a GC block is applied at the analysis side in each channel after filtering and downsampling, and an inverse GC block is applied at the synthesis side prior to filtering and upsampling. As a result, the filterbank remains perfect

reconstruction.

D. Nomenclature and design of graphBior filterbanks

The proposed biorthogonal filterbanks are specified by four parameters $(k_0, \ k_1, \ l_0, \ l_1)$, where k_0 is the number of roots of low pass analysis kernel $\hat{h}_0(\lambda)$ at $\lambda = 0$, k_1 is the number of roots of low pass synthesis kernel $\hat{g}_0(\lambda)$ at $\lambda = 0$, l_0 is the highest degree of low pass analysis kernel $\hat{h}_0(\lambda)$, and l_1 is the highest degree of low pass synthesis kernel $\hat{g}_0(\lambda)$. The other two filters, namely $\hat{h}_1(\lambda)$ and $\hat{g}_1(\lambda)$, can be computed as in (12). Given these specifications, we design $\hat{p}(\lambda) = \hat{h}_0(\lambda)\hat{g}_0(\lambda)$ as a maximally flat half band polynomial kernel with $K = k_0 + k_1$ roots at $\lambda = 0$. As a result, $\hat{p}(\lambda)$ turns out to be a $2K - 1$ degree polynomial, and we factorize it into $\hat{h}_0(\lambda)$ and $\hat{g}_0(\lambda)$, with least dissimilar lengths (i.e., we choose $l_0 = K$ and $l_1 = K - 1$). We use Θ as the criterion to compare various possible factorizations, and choose the one with the maximum value of Θ. This leads to a unique design of biothogonal filterbanks. We term our proposed filterbanks as *graphBior*(k_0, k_1). We designed graphBior filterbanks for various values of (k_0, k_1), and we observed that designs with $k_0 = k_1$ stand out, as they are close to orthogonal and have near-flat pass-band responses. The lowpass and highpass analysis kernels are plotted in Figure 4, and their coefficients are shown in Table II.

graphBior(k_0,k_1)	**filter coefficients**
$k_0 = 6$, $k_1 = 6$	$\hat{h}_1$ = [-0.3864 4.0351 -17.0630 36.5763 -39.8098 17.6477 0 0 0 0 0 0] $\hat{h}_0$ = [0.4352 -4.9802 23.2396 -55.4662 67.2657 -29.0402 -13.0400 7.5253 9.5267 -4.8746 -2.0616 1.2633 1.2071]
$k_0 = 7$, $k_1 = 7$	$\hat{h}_1$ = [0.3115 -3.9523 21.0540 -60.3094 98.0605 -85.9222 31.7578 0 0 0 0 0 0 0] $\hat{h}_0$ = [-0.4975 6.8084 -39.6151 126.2423 -234.3683 241.5031 -97.6557 -46.2635 62.1232 -19.3648 -2.0766 6.5886 -4.5632 0.5775 1.5614]
$k_0 = 8$, $k_1 = 8$	$\hat{h}_1$ = [-0.3232 4.7284 -29.7443 104.3985 -221.0705 282.7915 -202.6283 62.8477 0 0 0 0 0 0 0 0] $\hat{h}_0$ = [0.4470 -6.9872 47.5460 -183.6940 440.0924 -670.0905 643.3979 -396.0713 209.9824 -154.0976 92.8617 -30.8228 16.6112 -12.7664 3.2403 -0.0284 1.3793]

TABLE II: Polynomial expansion coefficients (highest degree first) of graphBior (k_0, k_1) filters (approximated to 4 decimal places) on a bipartite graph. Refer to the Matlab code for more accurate coefficients.

IV. ZERODC GRAPHBIOR FILTERBANKS

In many application such as images, videos and wireless sensor networks etc., where the underlying graph resides in a physical space, an all constant signal (a DC signal) has a physical interpretation, and the wavelet
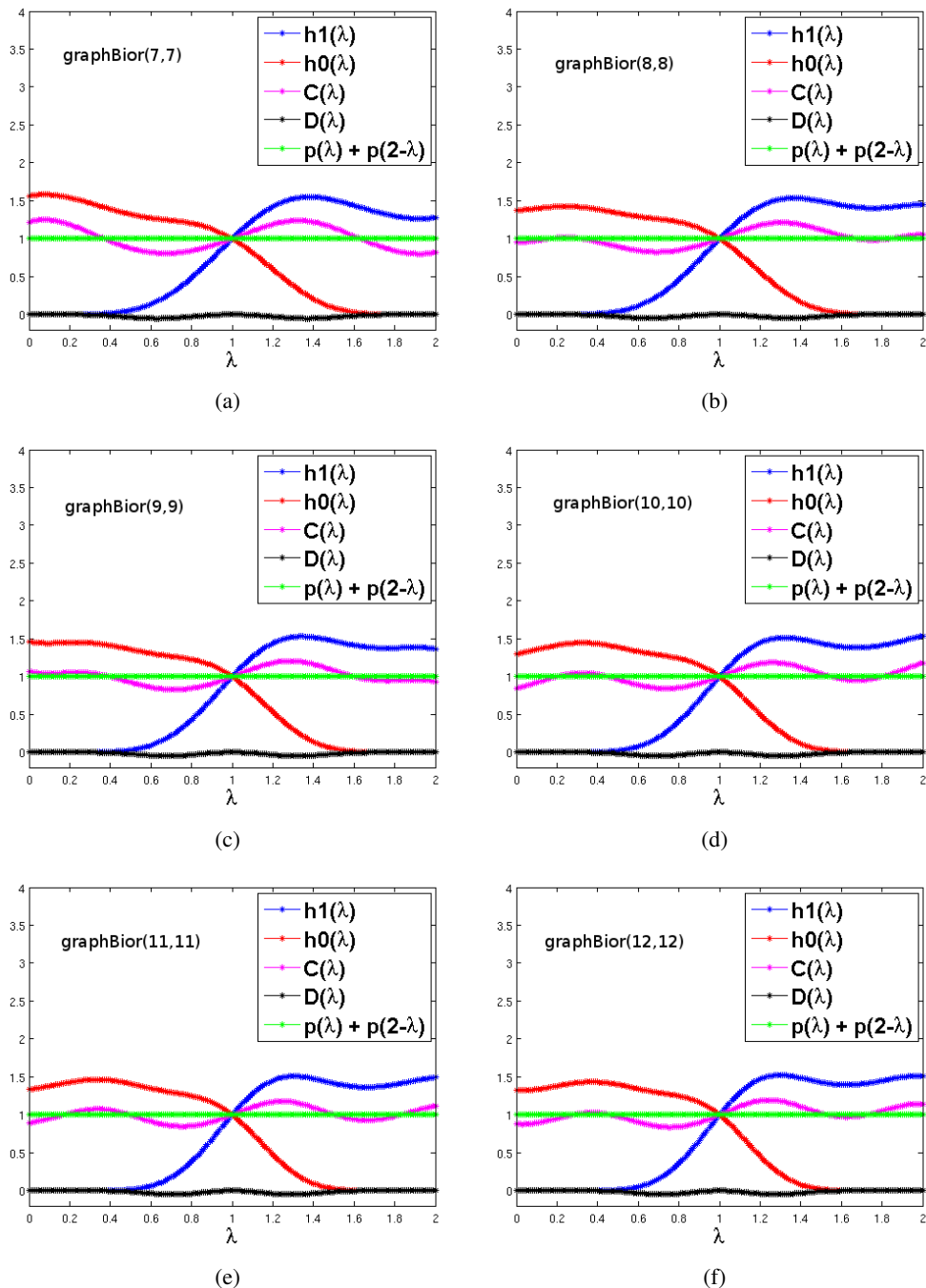
Fig. 4: Spectral responses of graphBior(k_0, k_1) filters on a bipartite graph. In each plot, $\hat{h}_0(\lambda)$ and $\hat{h}_1(\lambda)$ are lowpass and highpass analysis kernels, $C(\lambda)$ and $D(\lambda)$ constitute the spectral response of the overall analysis filter $\mathbf{T}_a$, as in (33). For near-orthogonality $D(\lambda) \approx 0$ and $C(\lambda) \approx 1$. Finally, $(p(\lambda) + p(2 - \lambda))/2$ represents perfect reconstruction property as in (15), and should be constant equal to 1, for perfect reconstruction.

filters are designed to be orthogonal to the dc signal. In our proposed nonzeroDC graphBior design, wavelet transforms (highpass transforms) are designed to be orthogonal to the eigenvector corresponding to 0 eigenvalue of the normalized Laplacian matrix $\mathcal{L}$, which is $\mathbf{D}^{1/2}\mathbf{1}$ where $\mathbf{D}$ is the degree matrix. If the graph is almost regular (i.e., has almost same degree at all nodes), then the vector $\mathbf{D}^{1/2}\mathbf{1}$ is almost constant. To obtain a zero DC

response for other non-regular graphs we propose *zeroDC graphBior filterbanks* designs. Construction-wise, the only difference between *zeroDC graphBior filterbanks* and *nonzeroDC graphBior filterbanks* is that the former are designed using random-walk Laplacian matrix $\mathcal{L}_r$ while the latter are designed using normalized Laplacian matrix $\mathcal{L}$. The two Laplacian matrices are *similar*, hence their eigenvalues are identical. Therefore, no change is needed in the design if spectral kernels designed above. In order to compute zeroDC filterbanks, we simply replace $\mathcal{L}$ by $\mathcal{L}_r$ in (7), i.e.,

$$
\begin{aligned}
\mathbf{H}_{ri} &= \hat{h}_i(\mathcal{L}_r) \\
\mathbf{G}_{ri} &= \hat{g}_i(\mathcal{L}_r),
\end{aligned}
\tag{37}
$$

and replacing symmetric filters $\mathbf{H}_i$ and $\mathbf{G}_i$ by $\mathbf{H}_{ri}$ and $\mathbf{G}_{ri}$, respectively, in the two-channel nonzeroDC filterbank implementation shown in Figure 1. The following results describe the properties of proposed zeroDC filterbanks:

*Proposition 2 (***zero DC response***):* For any connected graph, if the spectral kernel $\hat{h}(\lambda)$ is such that $\hat{h}(0) = 0$, then the transform $\mathbf{H}_r = \hat{h}(\mathcal{L}_r)$ has a zero DC response, i.e., $\mathbf{H}_r \mathbf{1} = \mathbf{0}$. Further, if $\hat{h}(\lambda)$ is a polynomial kernel then the transforms $\mathbf{H}_r = \hat{h}(\mathcal{L}_r)$ and $\mathbf{H} = \hat{h}(\mathcal{L})$ are related as:

$$
\mathbf{H}_r = \mathbf{D}^{-1/2}\mathbf{H}\mathbf{D}^{1/2}.
\tag{38}
$$

Proof: The random walk Laplacian matrix $\mathcal{L}_r$ can be diagonalized as:

$$
\mathcal{L}_r = \mathbf{D}^{-1/2}\mathbf{U}\boldsymbol{\Lambda}(\mathbf{D}^{-1/2}\mathbf{U})^{-1} = \mathbf{D}^{-1/2}\mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\top}\mathbf{D}^{1/2}.
$$

Therefore, any function of $\mathcal{L}_r$ can be written as:

$$
\mathbf{H}_r = \hat{h}(\mathcal{L}_r) = \mathbf{D}^{-1/2}\mathbf{U}\hat{h}(\boldsymbol{\Lambda})\mathbf{U}^{\top}\mathbf{D}^{1/2} = \mathbf{D}^{-1/2}\hat{h}(\mathcal{L})\mathbf{D}^{1/2} = \mathbf{D}^{-1/2}\mathbf{H}\mathbf{D}^{1/2}.
$$

Further, if $\mathbf{u}_l$ is an eigenvector of normalized Laplacian matrix $\mathcal{L}$ corresponding to eigenvalue λ_l, then by definition $\mathbf{H}\mathbf{u}_l = \hat{h}(\lambda_l)\mathbf{u}_l$. Since $\mathbf{u}_0 = \mathbf{D}^{1/2}\mathbf{1}$ is the eigenvector of $\mathcal{L}$ with eigenvalue 0, this implies:

$$
\mathbf{H}_r\mathbf{1} = \mathbf{D}^{-1/2}\mathbf{H}\mathbf{D}^{1/2}\mathbf{1} = \lambda_0\mathbf{1} = \mathbf{0}.
$$

Thus $\mathbf{H}_r$ has zero DC response. ∎

*Proposition 3 (***perfect reconstruction property***):* The zeroDC filterbanks designed using graphBior spectral kernels are also perfect reconstruction.

Proof: Similar to (8), the overall transfer function of the zeroDC filterbank can be written as:

$$
\begin{aligned}
\hat{\mathbf{f}} &= \frac{1}{2}\mathbf{G}_{r0}(\mathbf{I}+\mathbf{J}_\beta)\mathbf{H}_{r0}\mathbf{f} + \frac{1}{2}\mathbf{G}_{r1}(\mathbf{I}-\mathbf{J}_\beta)\mathbf{H}_{r1}\mathbf{f} \\
&= \frac{1}{2}(\mathbf{G}_{r0}\mathbf{H}_{r0}+\mathbf{G}_{r1}\mathbf{H}_{r1})\mathbf{f} + \frac{1}{2}(\mathbf{G}_{r0}\mathbf{J}_\beta\mathbf{H}_{r0}-\mathbf{G}_{r1}\mathbf{J}_\beta\mathbf{H}_{r1})\mathbf{f}.
\end{aligned}
\tag{39}
$$

Using the similarity relation given in (38), we can simplify (39) as:

$$
\begin{aligned}
\hat{\mathbf{f}} &= \frac{1}{2}(\mathbf{D}^{-1/2}\mathbf{G}_0\mathbf{D}^{1/2}\mathbf{D}^{-1/2}\mathbf{H}_0\mathbf{D}^{1/2}+\mathbf{D}^{-1/2}\mathbf{G}_1\mathbf{D}^{1/2}\mathbf{D}^{-1/2}\mathbf{H}_1\mathbf{D}^{1/2})\mathbf{f} \\
&+ \frac{1}{2}(\mathbf{D}^{-1/2}\mathbf{G}_0\mathbf{D}^{1/2}\mathbf{J}_\beta\mathbf{D}^{-1/2}\mathbf{H}_0\mathbf{D}^{1/2}-\mathbf{D}^{-1/2}\mathbf{G}_1\mathbf{D}^{1/2}\mathbf{J}_\beta\mathbf{D}^{-1/2}\mathbf{H}_1\mathbf{D}^{1/2})\mathbf{f}.
\end{aligned}
\tag{40}
$$

In (40), the matrices $\mathbf{D}^{1/2}, \mathbf{J}_\beta$, and $\mathbf{D}^{-1/2}$ are diagonal matrices and hence commute with each other. Therefore,

$$
\mathbf{D}^{1/2}\mathbf{J}_\beta\mathbf{D}^{-1/2} = \mathbf{J}_\beta\mathbf{D}^{1/2}\mathbf{D}^{-1/2} = \mathbf{J}_\beta
\tag{41}
$$

Thus, (40), can be simplified as:

$$
\begin{aligned}
\hat{\mathbf{f}} &= \frac{1}{2}(\mathbf{D}^{-1/2}\mathbf{G}_0\mathbf{H}_0\mathbf{D}^{1/2}+\mathbf{D}^{-1/2}\mathbf{G}_1\mathbf{H}_1\mathbf{D}^{1/2})\mathbf{f} \\
&+ \frac{1}{2}(\mathbf{D}^{-1/2}\mathbf{G}_0\mathbf{J}_\beta\mathbf{H}_0\mathbf{D}^{1/2}-\mathbf{D}^{-1/2}\mathbf{G}_1\mathbf{J}_\beta\mathbf{H}_1\mathbf{D}^{1/2})\mathbf{f} \\
&= \mathbf{D}^{-1/2}\mathbf{T}_{eq}\mathbf{D}^{1/2}\mathbf{f}+\mathbf{D}^{-1/2}\mathbf{T}_{alias}\mathbf{D}^{1/2}\mathbf{f} \\
&= \mathbf{D}^{-1/2}(\mathbf{T}_{eq}+\mathbf{T}_{alias})\mathbf{D}^{1/2}\mathbf{f},
\end{aligned}
\tag{42}
$$

where $\mathbf{T}_{eq}$ and $\mathbf{T}_{alias}$ correspond to the overall transfer function of nonzeroDC filterbanks, as defined in (8). *Therefore, the zeroDC filterbank implementation is equivalent to pre-multiplying the input by $\mathbf{D}^{-1/2}$ and post-multiplying the output by $\mathbf{D}^{1/2}$, and if the nonzeroDC filterbank is PR (i.e., $\mathbf{T}_{eq}+\mathbf{T}_{alias} = c\mathbf{I}$) then the corresponding zeroDC filterbank is also PR.* ∎

*Proposition 4 (***Riesz bounds***):* The zeroDC filterbanks form a Riesz basis with lower bound $A\sqrt{d_{min}/d_{max}}$ and upper bound $B\sqrt{d_{max}/d_{min}}$, where A and B are the lower and upper bounds of the Riesz basis formed by corresponding nonzeroDC graphBior filterbanks.

Proof: Referring to Figure 1, the wavelet coefficient vector $\mathbf{w}$ produced in the zeroDC filterbanks can be written as:

$$
\begin{aligned}
\mathbf{w}_r = \mathbf{T}_{ra}\mathbf{f} &= \frac{1}{2}(\mathbf{I}-\mathbf{J}_\beta)\mathbf{H}_{r0}\mathbf{f}+\frac{1}{2}(\mathbf{I}+\mathbf{J}_\beta)\mathbf{H}_{r1}\mathbf{f} \\
&= \frac{1}{2}(\mathbf{H}_{r1}+\mathbf{H}_{r0})\mathbf{f}+\frac{1}{2}\mathbf{J}_\beta(\mathbf{H}_{r1}-\mathbf{H}_{r0})\mathbf{f} \\
&= \frac{1}{2}\mathbf{D}^{-1/2}(\mathbf{H}_1+\mathbf{H}_0)\mathbf{D}^{1/2}\mathbf{f}+\frac{1}{2}\mathbf{D}^{-1/2}\mathbf{J}_\beta(\mathbf{H}_1-\mathbf{H}_0)\mathbf{D}^{1/2}\mathbf{f} \\
&= \mathbf{D}^{-1/2}\mathbf{T}_a\mathbf{D}^{1/2}\mathbf{f}
\end{aligned}
\tag{43}
$$

This implies that the nth output can we written as:

$$w_r[n] = \sum_{m=1}^{N} \sqrt{\frac{d_m}{d_n}} T_a(n,m)f[m] \tag{44}$$

Note that if the graph is almost regular, i.e., $\frac{d_m}{d_n} \approx 1$, then $w_r[n] \approx \sum_{m=1}^{N} T_a(n,m)f[m] = w[n]$, where $w[n]$ is the nth output of the corresponding nonzeroDC filterbank. In order to obtain a worst-case bound, if we define $\mathbf{f_D} = \mathbf{D}^{1/2}\mathbf{f}$, and $\mathbf{w_D} = \mathbf{D}^{1/2}\mathbf{w}$, then (43) can be written as $\mathbf{w_D} = \mathbf{T}_a\mathbf{f_D}$. Thus, if the corresponding nonzeroDC filterbank is biorthogonal with Riesz bounds A and B, then $A||\mathbf{f_D}|| \leq ||\mathbf{w_D}|| \leq B||\mathbf{f_D}||$ (the 2-norm). However,

$$d_{min}\sum_{i=1}^{N} w^2(i) \leq ||\mathbf{w_D}||^2 = \sum_{i=1}^{N} d_i w^2(i) \leq d_{max}\sum_{i=1}^{N} w^2(i)$$

$$d_{min}\sum_{i=1}^{N} f^2(i) \leq ||\mathbf{f_D}||^2 = \sum_{i=1}^{N} d_i f^2(i) \leq d_{max}\sum_{i=1}^{N} f^2(i), \tag{45}$$

where d_{min} is the minimum degree in the graph (1 if there is an isolated node), and d_{max} is the maximum degree. Using (45), we obtain:

$$d_{min}||\mathbf{w}||^2 \leq ||\mathbf{w_D}||^2 \leq B^2||\mathbf{f_D}||^2 \leq B^2 d_{max}||\mathbf{f}||^2$$

$$d_{min}A^2||\mathbf{f}||^2 \leq A^2||\mathbf{f_D}||^2 \leq ||\mathbf{w_D}||^2 \leq d_{max}||\mathbf{w}||^2, \tag{46}$$

and

$$\left(A\frac{d_{min}}{d_{max}}\right)||\mathbf{f}||^2 \leq ||\mathbf{w}||^2 \leq \left(B\frac{d_{max}}{d_{min}}\right)||\mathbf{f}||^2 \tag{47}$$

Thus, the zero graphBior filterbanks defines a Riesz basis in the graph-signal space, with lower bound $A_r = A\sqrt{d_{min}/d_{max}}$ and upper-bound $B_r = B\sqrt{d_{max}/d_{min}}$. ∎

Note that for regular graphs $d_{min} = d_{max}$, hence $\{A_r, B_r\} = \{A, B\}$. However, for irregular graphs the measure of orthogonality $\Delta_r = A_r/B_r = (d_{min}/d_{max})\Theta$ tend to be smaller than Θ, which implies that the basis functions in zeroDC filterbanks are more coherent than the basis functions in nonzeroDC filterbanks. This is also confirmed empirically in Table III.

The decision of whether to use zeroDC graphBior or nonzeroDC graphBior filterbanks, depends upon the interpretation of an all constant signal $\mathbf{1}$ and its degree normalized form $\mathbf{D}^{1/2}\mathbf{1}$ in the context of the problem. For example, in graphs arising from physical domains (sensor networks, transport networks, images and videos etc.), the graph signals are often nearly constant (or piecewise constant). In these cases, the all-constant signal should be preserved as the lowpass signal, and therefore the zeroDC filterbanks should be preferred over the nonzeroDC filterbanks. On the other hand, it is shown in [18] that for highly irregular graphs (such as online social networks, Internet etc.) the spectral analysis gets influenced by the presence of high degree nodes, and thus misses

the structure around low degree nodes. Therefore, the degree normalized nonzeroDC filterbanks should be used for these cases. All the examples presented in Section VI belong to the former category (i.e., they arise in physical domains). Therefore, the zeroDC filterbanks are found to perform better than the nonzeroDC filterbanks.

V. MULTI-DIMENSIONAL AND MULTI-RESOLUTION IMPLEMENTATIONS

So far we have described how to implement graphBior filterbanks on bipartite graphs. This is because bipartite graphs provide perfect reconstruction conditions in terms of simple conditions on spectral responses in these filterbanks. However, not all graphs are bipartite. For arbitrary graphs, we proposed in [2], [19] to decompose the graph G into K link-disjoint bipartite subgraphs, each defined on the entire set of vertices and their union covering almost all of the links in the graph. Consequently, we implemented filtering/downsampling operation in K stages, restricting the operations in each stage to only one bipartite graph. An example of 2-dimensional bipartite subgraph decomposition is shown in Figure 5, in which the graph G is divided into 4 clusters LL, LH, HL and HH. The first bipartite graph B_1 corresponds to partitions $L1 = LL \cup LH$ and $H1 = HL \cup HH$, and all the links connecting nodes in the two partitions. Subsequently, these links are removed from G and the second bipartite subgraph B_2 corresponds to partitions $L2 = HL \cup LL$ and $H2 = LH \cup HH$, and all the links between $L2$ and $H2$ from the remaining set of links. The remaining links are either discarded or used to further compute third and fourth bipartite subgraphs etc. The block diagram of a 2 "dimensional" graphBior filterbank is shown in Figure 6,
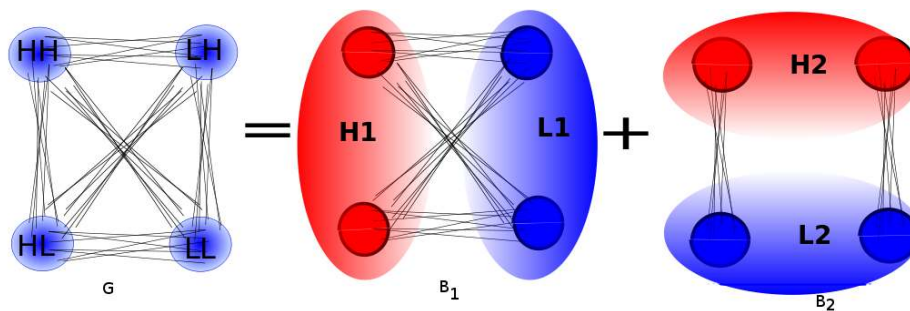

Fig. 5: Two dimensional decomposition of a graph.

where a dimension is interpreted as filtering and downsampling on a single bipartite subgraph. Note that this design is analogous to separable filterbank implementation on regular multidimensional signals. For example in the case of separable transforms for 2D signals, filtering in one dimension (e.g., row-wise) is followed by filtering of the outputs along the second dimension (column-wise). Moreover, the separable graphBior filterbanks are PR for any arbitrary partitions LL, LH, HL and HH induced on the graph. The choice of a specific bipartite subgraph decomposition depends on various factors. For highly structured graphs such as graph representation of regular signals, the bipartite subgraphs which preserve the structure are more useful (see, for example, Section VI-B). For arbitrary graphs, there can be various criteria. One criterion is to compute a graph decomposition that generates
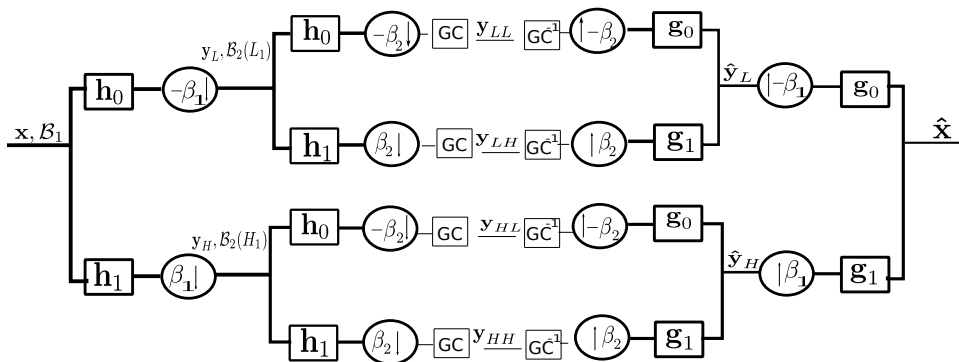
Fig. 6: Separable two-dimensional filterbank on graphs. The graph is first decomposed into two bipartite subgraph as shown in Figure 5. The binary function β_1 is such that $\beta_1(H_1) = 1$ and $\beta_1(L_1) = -1$. Similarly the binary function β_2 is such that $\beta_1(H_2) = 1$ and $\beta_1(L_2) = -1$. For each bipartite graph the graph transform pair $\{h_0, h_1\}$ forms the analysis low-pass and analysis high-pass graphBior filters respectively and $\{g_0, g_1\}$ are corresponding synthesis filters. GC: gain-compensation block, GC^{-1}: inverse GC block.

minimum number of bipartite subgraphs whose union covers all the links in the graph. An example of decomposition scheme based on such criterion is *Harary's algorithm* proposed in [2], which provides a $\lceil log_2 K \rceil$ bipartite subgraph decomposition of a K-colorable graph[9]. Another criterion introduced in [19], proposes subgraph decompositions so that the neighborhood sets of each node on different bipartite subgraphs are maximally disjoint. This leads to uncorrelated filtering operations on different graphs. However, whether the above mentioned decomposition schemes are optimal in some sense, or more generally whether there are other ways to extend graphBior filterbanks to arbitrary graphs, is part of our on-going research.

The multiresolution decomposition (MR) property in graphs implies successive coarser approximations of the graph and graph signal. For example, in a 1-dimensional implementation, the output samples in the set L are treated as signal for the next resolution level, and the vertices in L are reconnected to form a downsampled graph that preserves properties of the original graph such as the intrinsic geometric structure (e.g., some notion of distance between vertices), connectivity, graph spectral distribution, and sparsity. The graph coarsening problem has received a great deal of attention from graph theorists, and, in particular, from the numerical linear algebra community (see [20], [21] and the reference therein). Further, Pesenson (e.g., [22]) has leveraged the analogy between the graph Fourier transform and the classical Fourier transform to extend the concept of bandlimited sampling to signals defined on graphs. Namely, certain classes of signals can be downsampled on particular subgraphs and then stably reconstructed from the reduced set of samples. In our designs, any of the above mentioned coarsening scheme can be used to compute the graph at the next level.

[9]A K-colorable graph can be divided into K clusters such that there are no links connected nodes in the same clusters. $\lceil \, . \, \rceil$ is the ceiling operator.

VI. Experiments

A. *Performance comparisons of two-channel filterbanks on graphs*

In order to compare various graphBior designs proposed in this paper and previously proposed graphQMF designs, we simulate M instances of random graphs. In all the experiments the random graphs are bipartite graphs with 300 nodes in each partition and probability of connection $2log(N)/N$. The isolated vertices in the graph are removed in each realization.

In order to show the trade-off between vertex domain and spectral domain localizations, we plot in Figure 7, the spatial spread (2) and spectral spread (5) of various two-channel spectral filterbanks on $M = 10$ instances of random bipartite graphs. We first observe that that the graph-QMF based on ideal half-band kernels (magenta diamonds in the plot) have very small spectral spread but very large spatial spread, as compared to other designs. This is due to the brick-wall spectral response of these filterbanks. The same graphQMF filterbanks when designed using smooth Meyer kernel based half-band filters (black squares in the plot), have lower spatial spread (though still higher than most of the graphBior filterbanks) but higher spectral spread. However, both of these designs do not have a compact support. On the other hand, the proposed graphBior filterbanks exploit the spatial/spectral tradeoff better and have compact support support. The filters with smaller filterlengths are spatially more localized but spectrally less localized than the filters with higher filterlengths. The filter length of graphBior designs is chosen to be the maximum of the two filter lengths (i.e, K). Among graphBior designs, the zeroDC filterbanks (red triangles in the plot) perform slightly worse than the nonzeroDC filterbanks, which is due to the extra normalizations introduced in the formers to make their DC response zero.

The exact graphQMF filterbanks provide PR but are not compact support. In [2], we proposed polynomial approximation of the exact graphQMF kernels which are compact support but results in some reconstruction error. A comparison between proposed graphBior filterbanks and the graph-QMF filterbanks, in terms of perfect reconstruction error (SNR) and orthogonality (Θ) is shown in Table III. The reconstruction SNR and orthogonality Θ are computed as an average over 20 instances of randomly generated graph-signals on $M = 10$ random bipartite graphs. It can be seen from Table III that all graphBior designs provide perfect reconstruction ($SNR > 100dB$). The graph-QMF filters in comparison are closer to orthogonal (i.e., Θ almost 1), but have considerably lower reconstruction SNR. We now consider some applications of our proposed filterbanks.

B. *Graph based image processing*

In this section, we describe an application of proposed graphBior filterbanks for image-analysis. This is an extension of our previous work in [2], [23], where we proposed a graph based edge-aware representation of image-signals. While standard separable extensions of wavelet filterbanks to higher dimensional signals, such as 2-D
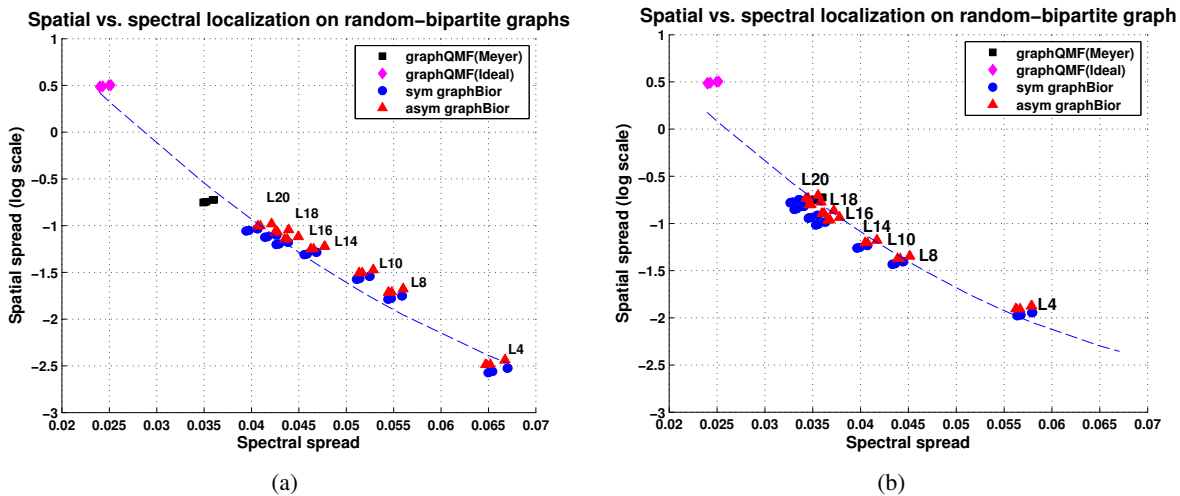
Fig. 7: The spatial vs. spectral spread a) of the highpass filters, and b) of the lowpass filters. The spatial/spectral coordinates for graphQMF filterbanks are same in both plots, since the lowpass and highpass filters are symmetric around $\lambda = 1$. However, the lowpass and highpass filterbanks in graphBior designs are neither symmetric nor equal length. Therefore, the spatial/spectral spreads of the two channels are different. The dashed line is a quadratic polynomial fit of the data points in the least square sense.

L	Graph	QMF	nonzeroDC	graphBior	zeroDC	graphBior
	SNR (dB)	Θ	SNR (dB)	Θ	SNR (dB)	Θ
4	32.20	0.98	286.84	0.88	286.54	0.70
8	32.25	0.98	282.89	0.87	282.71	0.66
10	42.17	1.00	270.05	0.81	270.00	0.65
14	48.09	1.00	230.83	0.85	230.73	0.64
16	44.78	0.99	222.08	0.94	222.05	0.64
18	45.23	0.99	190.53	0.92	190.43	0.63
20	54.61	1.00	170.78	0.94	170.68	0.63

TABLE III: Comparison between graph-QMF filterbanks (polynomial approximations) and graphBior filterbanks on random bipartite graphs.

images, provide useful multi-resolution analysis, they do not capture the intrinsic geometry of the images. For example, these extensions can capture only limited (mostly horizontal and vertical) directional information. Images can also be viewed as graphs, by treating pixels as nodes, pixel intensities as graph-signals, and by connecting pixels with their neighbors in various ways. The advantage of formulating images as graphs is that different graphs can represent the same image, which offers flexibility of choosing the graphs that have useful properties. In [2], we proposed an 8-connected graph representation of images, in which each pixel is connected to 8 of its nearest neighbors (4 diagonal, 2 vertical and 2 horizontal) as shown in Figure 8. The graph is not bipartite, but can be decomposed into two bipartite subgraphs, one containing links in the horizontal and vertical direction and other in the diagonal directions. The proposed graphBior filterbanks can then be applied as two "dimensional" filterbanks as in Figure 6. The advantage of using graphBior wavelet filterbanks as against standard separable filterbanks, is
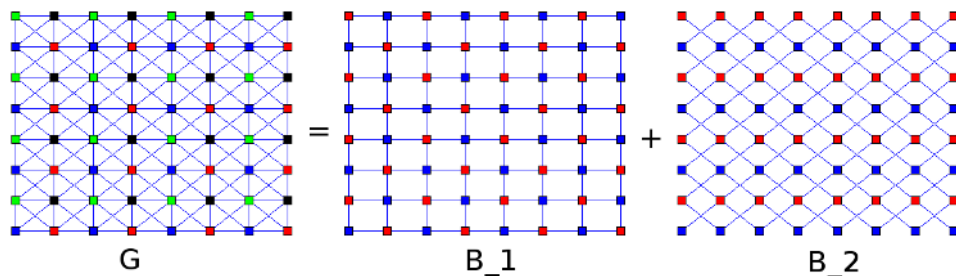
Fig. 8: Two dimensional decomposition of 8-connected image-graph

that the former provide more filtering directions (diagonal and rectangular) than the latter (only rectangular), at the same order of computational complexity. For multi-resolution analysis, the downsampled set of nodes in the LL channel, are again connected to 8 of their neighbors, to create a downsampled graph and the graphBior filterbanks are implemented iteratively on the downsampled graphs. Thus the downsampling ratio at each level is same in both graphBior filterbanks and standard separable filterbanks.

In [23], we proposed an *edge-aware* implementation for piece-wise smooth images, in which the bipartite subgraphs obtained in Figure 8, can be simplified by removing the links between pixels across which the pixel intensity changes drastically. These links can be found using any standard edge-detection algorithm (we use Canny edge detection in our experiments and remove connected components less than 50 pixels before computing the graph). The advantage of edge-aware graph representations is that it avoids filtering across edges, which leads to a very significant reduction in the number of large coefficients near edge (and thus corresponding reductions in rate). Note that in a compression application, this would require generating an edge map at the encoder and then sending it to the decoder. However, recent work [24], [25] using transforms based on similar edge-map information have shown that even with the extra overhead of sending the edge map we can achieve reductions in overall transmitted rate.

In order to demonstrate the advantage of graph based implementation of proposed filterbanks, we choose *coins.png* image as shown in Figure 9(a) with many round shaped coins. We implement graphBior filterbanks of length 10, (i.e., $graphBior(5,5)$), and compare them against standard separable CDF $9/7$ wavelet filterbanks in a non-linear approximation of images, using 4 resolution levels. Figure 9 shows the reconstruction of coins.png using all lowpass coefficients and top 4% of the highpass coefficients in terms of magnitude. Since the standard separable wavelet filterbanks filter only in horizontal and vertical directions, they produce lots of large magnitude wavelet coefficients (and hence blurring artifacts) near the edges (see Figure 9(b)). The zeroDC graphBior filterbank implementation on the regular 8-connected image graph (Figure 9(c)) does slightly better since it also provide filtering in diagonal directions. However, the best performance in terms of reconstruction quality is observed for proposed edge-aware zeroDC graphBior filterbanks, especially in preserving the edge structure (see Figure 9(d)). This is due to the fact
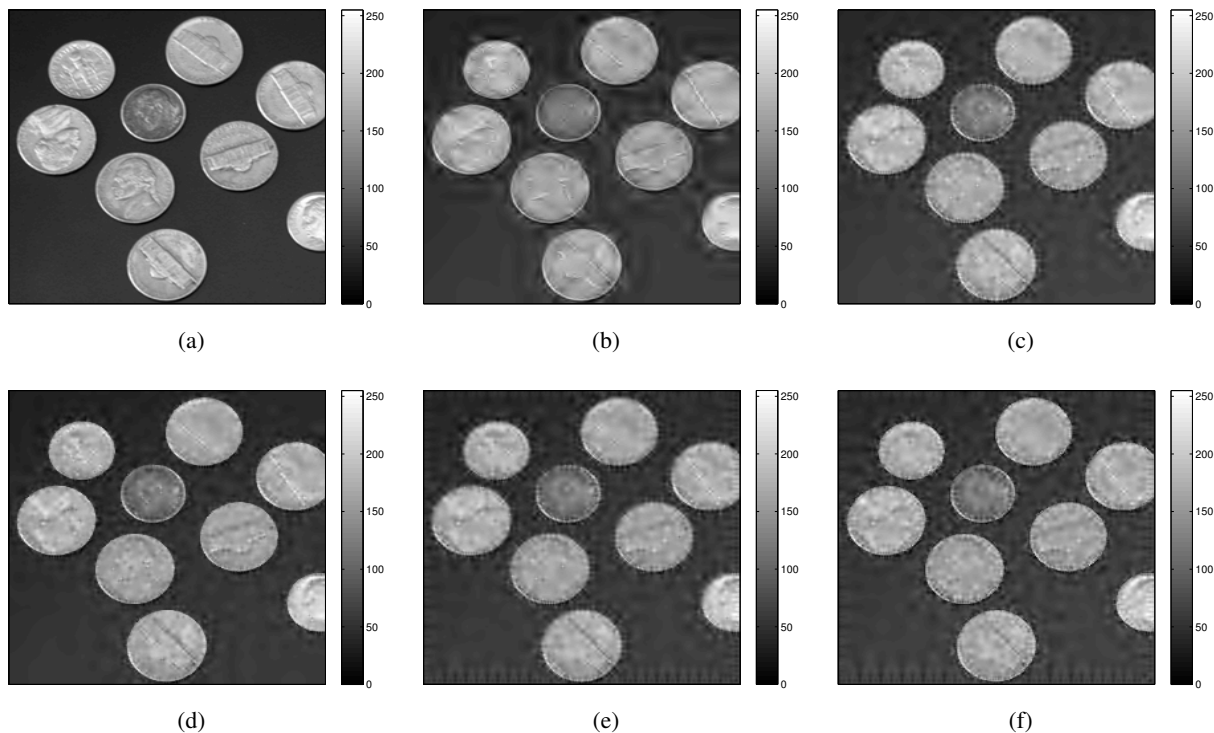
Fig. 9: Reconstruction of "Coin.png" (512×512) from all lowpass coefficients and 3% highpass coefficients after a 4-level decomposition. (a) Original image, (b) standard CDF $9/7$ filters, (c) zeroDC filterbanks on regular 8-connected image graph (d) zeroDC filterbanks on edge-aware image graph (e) nonzeroDC filterbanks on regular 8 connected image graph and (f) nonzeroDC filterbanks on edge-aware image graph.

that the underlying graphs in this approach are disconnected at the edges, and hence the filtering operations do not cross the edges. Theoretically, the nonzeroDC filterbanks should perform almost the same as zeroDC filterbanks for regular degree graphs. The 8-connected image-graphs are almost regular except at the boundaries, and edges, and we observe in Figures 9(e) and 9(f), that significant ringing artifacts are produced near these places, when using nonzeroDC filterbanks. The problem of boundary artifacts also arises when using standard filterbanks on images, which is usally solved by providing signal extensions at the boundaries. Whether such signal extensions can be proposed for graph representation of images, is an open issue. Figure 10, shows PSNR and SSIM [26] values plotted against fraction of detail coefficients used in the reconstruction of *coins.png* image, and it can be seen from both the plots that zeroDC graphBior filterbanks perform better (up to 2dB better in PSNR) than the standard CDF $9/7$ filterbanks. Thus, the results show that the proposed graphBior filterbanks provide advantages over the standard wavelet transforms, with the same order of computational complexity.

C. Compression and Learning on arbitrary graphs

The proposed filterbanks are useful in analyzing and compressing signals defined on arbitrary graphs. As a proof of concept, we implement proposed graphBior filterbanks on the Minnesota traffic graph used in [2]. The graph
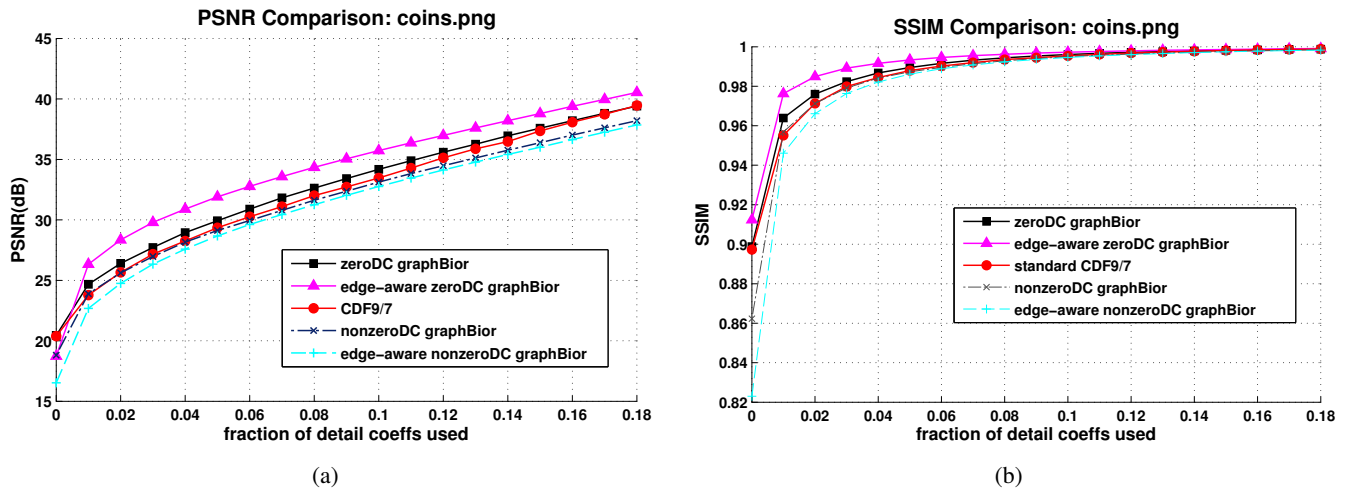
Fig. 10: Reconstruction of coins.png image from all low-pass coefficient and a fraction of wavelet coefficients (sorted in the order of magnitudes). The fraction value is plotted on the x-axis. (a) PSNR of the reconstructed images, (b) SSIM of the reconstructed image.

is shown in Figure 11(a), and the graph signal to be analyzed is shown in Figure 11(b), where the color of a node represents the signal value at that node. The graph is perfectly 3-colorable and hence, it can be decomposed using Harary's decomposition [2] into $\lceil log_2(3) \rceil = 2$ bipartite subgraphs, which are shown in Figure 11(c-d), and a 2-dimensional graphBior filterbank given in Figure 6 with filterlength $= 10$ is implemented on the graph.

The output in the 4 channels can be interpreted as follows: the LL channel providing a smooth approximation of the original signal on a subset of nodes, and the remaining channels providing details required for perfect reconstruction. Moreover, the total number of outputs in all channels is equal to the total number of input samples, hence the transform is critically sampled. The HL channel does not sample any output and is empty. The output coefficients of both zeroDC and nonzeroDC filterbanks in LL, LH and HH channels are shown in Figure 12. Note that the graph-signal is piece-wise constant, hence the proposed zeroDC filterbanks (bottom row in Figure 12) provide a sparser approximation than the nonzeroDC filterbanks (top row in Figure 12). As a result, the non-linear approximation of the graph signal with only 1% highpass coefficients (and all low pass coefficients) provide better SNR when using zeroDC filterbanks than when using nonzeroDC filterbanks as shown in Figure 13.

VII. CONCLUSIONS

In this paper we have presented novel graph-wavelet filterbanks that provide a critically sampled representation with compactly supported basis functions. The filterbanks come in two flavors: a) nonzeroDC filterbanks, and b) zeroDC filterbanks. The former filterbanks are designed as polynomials of the normalized graph Laplacian matrix, and the latter filterbanks are extensions of the former to provide a zero response by the highpass operators. Preliminary results showed that the filterbanks are useful not only for arbitrary graph but also to the standard
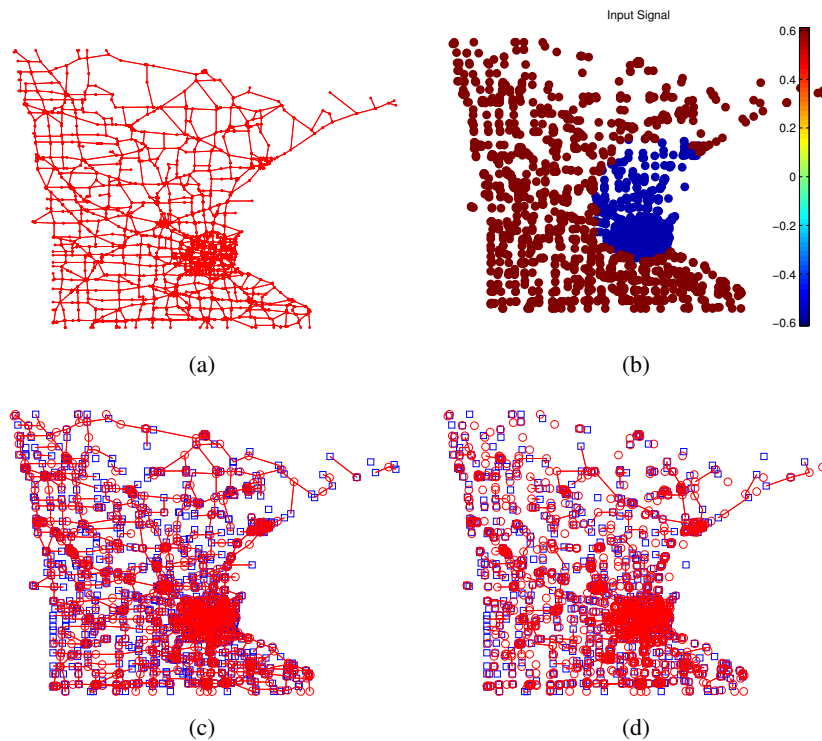
Fig. 11: (a) The Minnesota traffic graph G, and (b) the graph-signal to be analyzed. The colors of the nodes represent the sample values. (c)(d) bipartite decomposition of G into two bipartite subgraphs using Harary's decomposition.

regular signal processing domains. Extensions of this work will focus on the application of these filters to different scenarios, including, for example, social network analysis, sensor networks etc.

REFERENCES

[1] David K. Hammond, Pierre Vandergheynst, and Rémi Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, Mar 2011.

[2] S.K. Narang and Ortega A., "Perfect reconstruction two-channel wavelet filter-banks for graph structured data," *IEEE trans. on Sig. Proc.*, vol. 60, no. 6, June 2012.

[3] A. Agaskar and Y. M. Lu, "Uncertainty principles for signals defined on graphs: Bounds and characterizations," in *ICASSP*, Kyoto, Japan, Mar. 2012, pp. 3493–3496.

[4] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *INFOCOM 2003*, Mar 2003, vol. 3, pp. 1848–1857.

[5] R. Coifman and M. Maggioni, "Diffusion wavelets," *Applied and Computational Harmonic Analysis*, vol. 21, pp. 53–94, 2006.

[6] M Maggioni, J. C. Bremer, R. R. Coifman, and A. D. Szlam, "Biorthogonal diffusion wavelets for multiscale representations on manifolds and graphs," in *Proc. SPIE Wavelet XI*, Sep. 2005, vol. 5914.

[7] J. C. Bremer, R. R. Coifman, M. Maggioni, and A. D. Szlam, "Diffusion wavelet packets," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 95–112, 2006.

[8] A. D. Szlam, M. Maggioni, R. R. Coifman, and J. C. Bremer, Jr., "Diffusion-driven multiscale analysis on manifolds and graphs: top-down and bottom-up constructions," in *Proc. SPIE Wavelets*, Aug. 2005, vol. 5914, pp. 445–455.
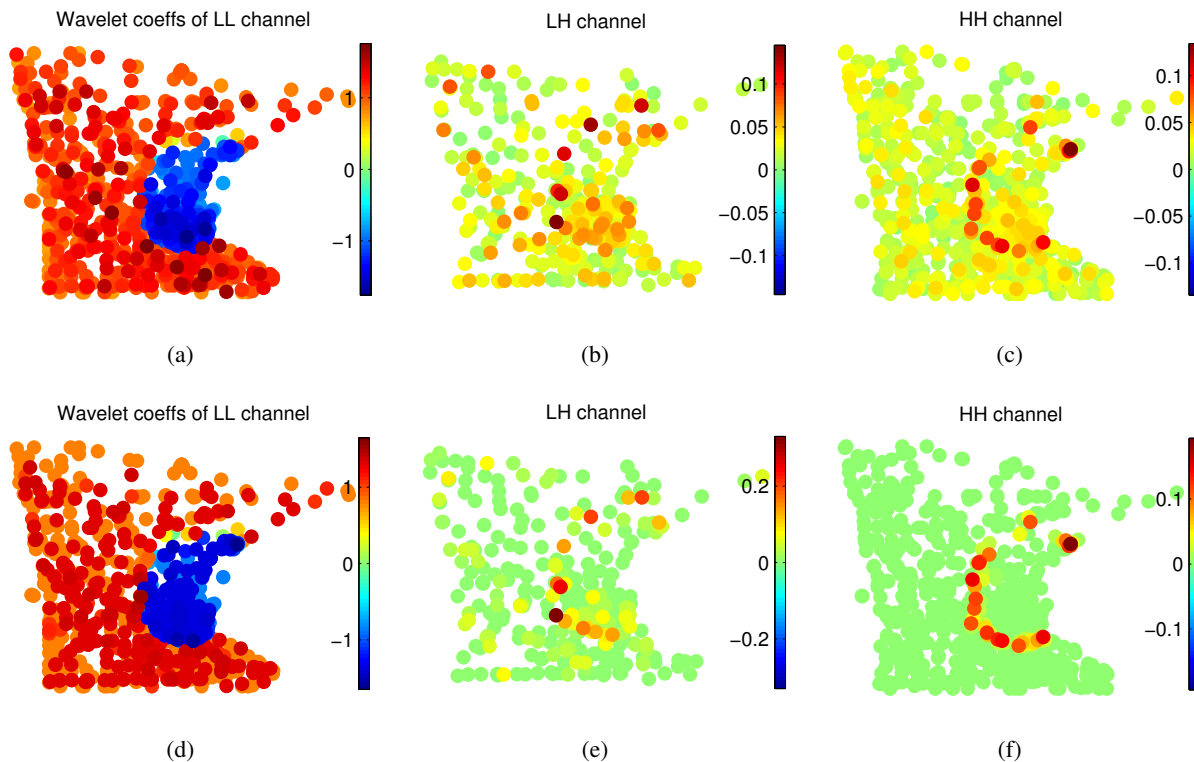
Fig. 12: output coefficients of the graphBior filterbanks with parameter $(k_0, k_1) = (7, 7)$. The node-color reflects the value of the coefficients at that point. Top-row: wavelet coefficients of nonzeroDC graphBior, bottom-row: wavelet coefficients of zeroDC graphBior,
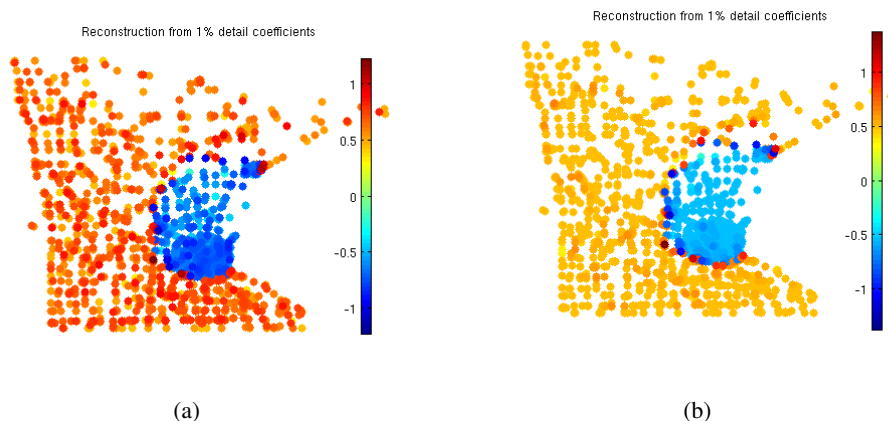

Fig. 13: Reconstructed graph-signals from all coefficients of LL channel and top 1% (in magnitude) wavelet coefficients form other channels. (a) nonzeroDC graphBior (SNR 15.50 dB) (b) zeroDC graphBior (SNR 36.24 dB).

[9] W. Wang and K. Ramchandran, "Random multiresolution representations for arbitrary sensor network graphs," in *ICASSP*, May 2006, vol. 4, pp. IV–IV.

[10] G. Shen and A. Ortega, "Transform-based distributed data gathering," *Sig. Proc., IEEE Trans. on*, vol. 58, no. 7, pp. 3802 –3815, july 2010.

[11] M. Jansen, G. P. Nason, and B. W. Silverman, "Multiscale methods for data on graphs and irregular multidimensional situations,"

Journal of the Royal Statistical Society, vol. 71, no. 1, pp. 97125, 2009.

[12] S. K. Narang and A. Ortega, "Lifting based wavelet transforms on graphs," *(APSIPA ASC' 09)*, Oct. 2009.

[13] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning," in *Proc. Int. Conf. Mach. Learn.*, Haifa, Israel, Jun. 2010, pp. 367–374.

[14] A. Cohen, I. Daubechies, and J.-C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 45, no. 5, pp. 485–560, 1992.

[15] D. Jakobson, S. D. Miller, I. Rivin, and Z. Rudnick, "Eigenvalue spacings for regular graphs," in *IN IMA VOL. MATH. APPL.* 1999, pp. 317–327, Springer.

[16] M. Vetterli and J. Kovačevic, *Wavelets and subband coding*, Prentice-Hall, Inc., NJ, USA, 1995.

[17] M. D. Adams and R. Ward, "Wavelet transforms in the JPEG-2000 standard," in *In Proc. of IEEE PacRim*, 2001, pp. 160–163.

[18] M. Mihail and C.Papadimitriou, "On the eigenvalue power law," in *RANDOM 2002*, Sep 2002, pp. 254–262.

[19] S.K. Narang and A. Ortega, "Multi-dimensional separable critically sampled wavelet filterbanks on arbitrary graphs," in *in ICASSP'12*, Mar 2012.

[20] D. Ron, I. Safro, and A. Brandt, "Relaxation-based coarsening and multiscale graph organization," *Multiscale Model. Simul.*, vol. 9, no. 1, pp. 407–423, Sep. 2011.

[21] C. Walshaw, "The graph partitioning archive," http://staffweb.cms.gre.ac.uk/~wc06/partition/.

[22] I. Pesenson, "Sampling in Paley-Wiener spaces on combinatorial graphs," *Trans. Amer. Math. Soc*, vol. 360, no. 10, pp. 5603–5627, 2008.

[23] S. K. Narang, Y. H. Chao, and A. Ortega, "Graph-wavelet filterbanks for edge-aware image processing," *IEEE SSP Workshop*, pp. 141–144, Aug. 2012.

[24] G. Shen, W.S. Kim, S.K. Narang, A. Ortega, J. Lee, and H.C. Wey, "Edge-adaptive transforms for efficient depth map coding," in *Picture Coding Symposium (PCS), 2010*, Dec 2010.

[25] W.S. Kim, S.K. Narang, and A. Ortega, "Graph based transforms for depth video coding," in *in ICASSP'12*, Mar 2012.

[26] H. R. Sheikh Z. Wang, A. C. Bovik and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. on Image Proc.*, vol. 13, no. 4, 2004.