CRITICALLY-SAMPLED GRAPH FILTER BANKS WITH SPECTRAL DOMAIN SAMPLING

Kana Watanabe[1], *Akie Sakiyama*[1], *Yuichi Tanaka*[1,2], *and Antonio Ortega*[3]

[1] Tokyo University of Agriculture and Technology, Tokyo, Japan
[2] PRESTO, Japan Science and Technology Agency, Saitama, Japan
[3] University of Southern California, CA, USA

ABSTRACT

This paper presents a framework for perfect reconstruction two-channel critically-sampled graph filter banks with spectral domain sampling. Graph signals have a unique characteristic: sampling in the vertex and graph spectral domains are generally different, in contrast to classical signal processing. Conventional graph filter banks are designed using vertex domain sampling, whereas the proposed approach utilizes a novel spectral domain sampling. Our proposed technique leads to perfect reconstruction transforms for any type of undirected graphs and can be applied both to combinatorial and symmetric normalized graph Laplacians. Some filter bank designs and an experiment on nonlinear approximation are shown to validate their effectiveness.

***Index Terms*—** Graph signal processing, spectral graph wavelets, spectral domain sampling, dictionary design

1. INTRODUCTION

1.1. Motivation

Graph signal processing has emerged in many fields demanding high-dimensional complex-structured data analysis [1], such as social/sensor/neuronal/traffic/electric networks, images/videos/point clouds, and machine learning [2–8].

One of the key topics in graph signal processing is design of transforms/dictionaries that represent signals on graphs sparsely. There have been many approaches so far [9–18]. Among them, critically-sampled (CS) perfect reconstruction transforms [9, 10, 15] are often required since the number of transformed coefficients in the spectral domain is the same as that of samples in the original vertex domain signal. This leads to an efficient signal processing system.

However, they can be realized in very restricted situations. For example, a class of CS spectral graph wavelets/filter banks are only applicable for signals on bipartite graphs, and the graph Laplacian used must be normalized [9, 10]. It has also been shown that, except for bipartite graphs, perfect reconstruction with low-degree polynomial filters cannot be obtained [19]. A proposed M-channel CS system can be applied to arbitrary graphs but requires interpolation at the synthesis side, which is completely different from the structure of the analysis bank [20]. Some vertex domain transforms [16, 21, 22] are CS and perfect reconstruction, but they are applicable only for specific type of graphs similar to the spectrum-based methods. In some cases, their spectral responses are not very clear. These restricted conditions for the graph wavelets/filter banks stem from effects of sampling in the graph spectral domain.

This work was supported in part by JST PRESTO Grant Number JPMJPR1656.

Sampling of graph signals has a unique behavior. In classical (digital) signal processing, sampling is defined in an intuitive way, i.e., every other sample is taken for downsampling (by two), and one zero is inserted between samples for upsampling (by two). It is also well known that downsampling broadens the bandwidth of the signal and the upsampling shrinks it and creates aliasing components. In other words, for time domain signals, we can define the same sampling in two different domains: Time and frequency (DFT) domains. In contrast to that, it has been found that sampling of graph signals in the vertex domain *does not* broaden or shrink the bandwidth [23] with the only exception being the case of bipartite graphs. This is problematic when we try to extend the classical signal processing framework into the graph setting since the effect of vertex domain downsampling does not have a simple representation in the frequency domain. Spectral domain sampling of graph signals has therefore been proposed in order to define sampling of graph signals in the graph spectral domain [23].

In this paper, we propose a framework of two-channel CS graph filter banks with spectral domain sampling (CSSGFBs). The proposed graph filter bank is the first one that satisfies all of the following desired properties.

1. Critical sampling.
2. Perfect reconstruction with the symmetric structure, i.e., the analysis and synthesis banks have similar building blocks.
3. Independent of the graphs and variation operators used.

In the proposed framework, perfect reconstruction is possible not only for the ideal filter banks, but also non-ideal ones, and they are designed in the graph spectral domain.

Along with the structure of the CSSGFBs, we show that perfect reconstruction is possible for any undirected graph, in contrast to the conventional graph filter banks. The perfect reconstruction condition can be quite similar to that of the classical signal processing. Some design examples are shown along with an application to nonlinear approximation.

1.2. Notation

A graph $\mathcal{G}$ is represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ denote sets of nodes and edges, respectively. Number of nodes is given as $N = |\mathcal{V}|$, unless otherwise specified. The (m, n)-th element of adjacency matrix $\mathbf{A}$ is $a_{mn} > 0$ if the mth and nth vertices are connected, or zero otherwise, where a_{mn} denotes the weight of the edge between m and n. The degree matrix $\mathbf{D}$ is a diagonal matrix, and its mth diagonal element is $d_{mm} = \sum_n a_{mn}$. The combinatorial graph Laplacian is $\mathbf{L} := \mathbf{D} - \mathbf{A}$ and the symmetric normalized graph Laplacian is $\boldsymbol{\mathcal{L}} := \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$. Since $\mathbf{L}$ (or $\boldsymbol{\mathcal{L}}$) is a real symmetric matrix, $\mathbf{L}$ can always be decomposed into $\mathbf{L} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\top}$, where $\mathbf{U} = [\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{N-1}]$ is the orthonormal eigenvector matrix,

$\mathbf{\Lambda} = \mathrm{diag}(\lambda_0, \lambda_1, \ldots, \lambda_{N-1})$ $(0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{N-1})$ is the diagonal eigenvalue matrix in which the eigenvalue is *graph frequency*, and $\cdot^\top$ represents the transpose of a matrix or a vector.

$\boldsymbol{f} \in \mathbb{R}^N$ is a graph signal, where the nth sample $f[n]$ is assumed to be located on the nth vertex of the graph. The graph Fourier transform (GFT) is defined as $\widetilde{f}[i] = \langle \boldsymbol{u}_i, \boldsymbol{f} \rangle = \sum_{n=0}^{N-1} u_i[n] f[n]$.

2. SAMPLING OF GRAPH SIGNALS

In this section, we introduce two strategies of sampling of graph signals: vertex and spectral domain methods. They are illustrated in Fig. 1.

2.1. Vertex Domain Sampling

The conventional and widely used method for sampling graph signals in the vertex domain, which corresponds to the intuitive counterpart of down- and upsampling in classical signal processing, is defined as follows:

Definition 1 (Sampling of graph signals in vertex domain)**.** *Let $\mathcal{G}_0 = (\mathcal{V}_0, \mathcal{E}_0)$ and $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ be the original graph and the reduced-size graph, respectively, where every vertex in $\mathcal{G}_1$ has one-to-one correspondence to one of the vertices in $\mathcal{G}_0$. In the vertex domain, downsampling of $\boldsymbol{f}$ to $\boldsymbol{f}_d \in \mathbb{R}^{|\mathcal{V}_1|}$ and upsampling of $\boldsymbol{f}_d$ to $\boldsymbol{f}_u \in \mathbb{R}^{|\mathcal{V}_0|}$ is defined as follows.*

(GD1): Vertex domain downsampling. Keeping samples in $\mathcal{V}_1$.

$$f_d[n] = f[n'] \text{ if } v_{0,n'} \in \mathcal{V}_0 \text{ corresponds to } v_{1,n} \in \mathcal{V}_1. \quad (1)$$

(GU1): Vertex domain upsampling. Placing samples on $\mathcal{V}_1$ into the corresponding vertices in $\mathcal{G}_0$.

$$f_u[n] = \begin{cases} f_d[n'] & \text{if } v_{1,n'} \in \mathcal{V}_1 \text{ corresponds to } v_{0,n} \in \mathcal{V}_0 \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

2.2. Spectral Domain Sampling

We now describe spectral domain sampling of graph signals introduced in [23]. Several slightly different definitions are possible. Here, we use one and refer to [23] for alternative definitions.

Definition 2 (Sampling of graph signals in graph spectral domain)**.** *Let $\mathbf{L}_0 \in \mathbb{R}^{N \times N}$ and $\mathbf{L}_1 \in \mathbb{R}^{N/2 \times N/2}$ be the graph Laplacians for the original graph and that for the reduced-size graph*[1]*, respectively, and assume that their eigendecompositions are given by $\mathbf{L}_0 = \mathbf{U}_0 \mathbf{\Lambda}_0 \mathbf{U}_0^\top$ and $\mathbf{L}_1 = \mathbf{U}_1 \mathbf{\Lambda}_1 \mathbf{U}_1^\top$. The downsampled graph signal $\boldsymbol{f}_d \in \mathbb{R}^{N/2}$ and upsampled graph signal $\boldsymbol{f}_u \in \mathbb{R}^N$ in the graph spectral domain are defined as follows.*

(GD2): Spectral domain downsampling. $\widetilde{\boldsymbol{f}}$ is divided into two that represent lower and higher graph spectra, respectively. Then the flipped version of the higher frequency component is added to the lower frequency component.

$$\widetilde{f}_d[i] = \widetilde{f}[i] + \widetilde{f}[N - i - 1] \quad (3)$$

where $i = 0, \ldots, N/2 - 1$. This is easily represented in matrix form $\boldsymbol{f}_d = \mathbf{U}_1 \mathbf{S}_d \mathbf{U}_0^\top \boldsymbol{f}$, where $\mathbf{S}_d = \begin{bmatrix} \mathbf{I}_{N/2} & \mathbf{J}_{N/2} \end{bmatrix}$, in which $\mathbf{I}$ and $\mathbf{J}$ are the identity and counter-identity matrices, respectively.

[1]We consider even N for the sake of simplicity.

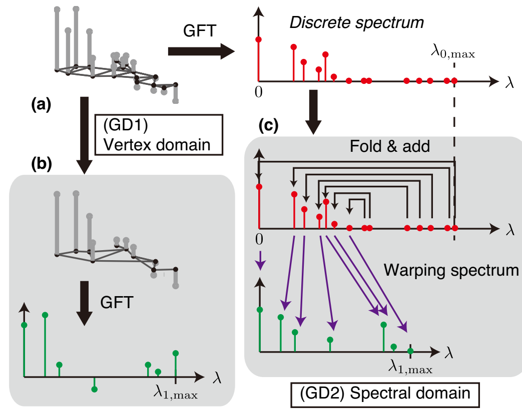

Fig. 1. Downsampling of signals on graphs. The signal is downsampled by two and bandlimited. The shaded areas represent *different* signals. (a) Original graph signal. (b) (GD1): vertex domain downsampling. (c) (GD2): spectral domain downsampling.

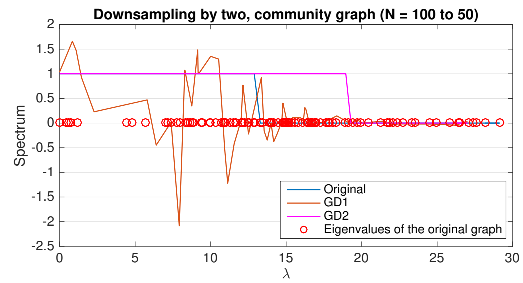

Fig. 2. Spectra of the original and downsampled signals. The underlying graph is a community graph with $N = 100$ and the signal is downsampled by two.

(GU2): Spectral domain upsampling. Repeating the downsampled spectrum and its flipped copy.

$$\widetilde{f}_u[i] = \begin{cases} \widetilde{f}_d[i] & i = 0, \ldots, N/2 - 1 \\ \widetilde{f}_d[N - i - 1] & i = N/2, \ldots, N - 1. \end{cases} \quad (4)$$

That is, $\boldsymbol{f}_u = \mathbf{U}_0 \mathbf{S}_d^\top \mathbf{U}_1^\top \boldsymbol{f}_d$.

An example for the downsampling of the graph signal on a community graph is shown in Fig. 2. In this example, the original signal is set to be bandlimited and the reduced-size graph is made with the Kron reduction technique [24]. From the knowledge of classical signal processing, one expects the spectrum after downsampling to be broadened. However, as shown in the figure, the vertex domain sampling does not have such a characteristic; Its spectrum rapidly oscillates. In contrast to that, the spectral domain sampling presents the expected characteristics from its intuitive definition. This is why we need to consider the graph signal processing systems using the spectral domain sampling.

3. CS GRAPH FILTER BANKS WITH VERTEX DOMAIN SAMPLING

The CS graph filter banks for bipartite graphs [9, 10] are introduced in this section. Let us define a bipartite graph $G = (\mathcal{L}, \mathcal{H}, \mathcal{E})$ where

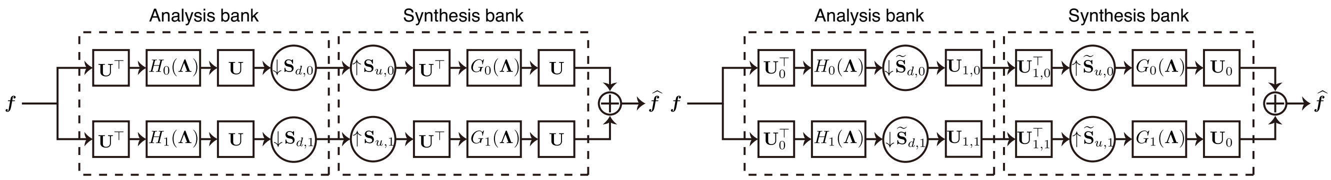

Fig. 3. CS graph filter banks. Left: Graph filter bank with vertex domain sampling. Right: Graph filter bank with spectral domain sampling.

vertices in G are divided into two disjoint sets $\mathcal{L}$ and $\mathcal{H}$. We call the vertices in $\mathcal{L}$ the lowpass channel and those in $\mathcal{H}$ the highpass channel, for the sake of convenience. The number of signals in each channel is determined on the basis of the graph-coloring result. Fig. 3 illustrates the entire transformation for one bipartite graph.

Similar to the regular signals, the downsampling and upsampling operators can be defined as follows:

$$\begin{aligned}\mathbf{S}_{d,0} = \mathbf{I}_{\mathcal{L}} \in \{0,1\}^{|\mathcal{L}|\times N},\ \mathbf{S}_{u,0} = \mathbf{S}_{d,0}^{\top}\\ \mathbf{S}_{d,1} = \mathbf{I}_{\mathcal{H}} \in \{0,1\}^{|\mathcal{H}|\times N},\ \mathbf{S}_{u,1} = \mathbf{S}_{d,1}^{\top}\end{aligned} \tag{5}$$

where $\mathbf{I}_{\mathcal{L}}$ ($\mathbf{I}_{\mathcal{H}}$) is a submatrix of $\mathbf{I}_N$ whose rows correspond to the indices of $\mathcal{L}$ ($\mathcal{H}$). Sampled signal can be represented as $\boldsymbol{f}_{d,0} = \mathbf{S}_{d,0}\boldsymbol{f}$ and so on.

The CS graph filter banks with vertex domain sampling [9, 10] are designed to satisfy the following perfect reconstruction condition:

$$\begin{aligned}\mathbf{T}_v &= \mathbf{G}_0\mathbf{S}_{u,0}\mathbf{S}_{d,0}\mathbf{H}_0 + \mathbf{G}_1\mathbf{S}_{u,1}\mathbf{S}_{d,1}\mathbf{H}_1\\ &= \mathbf{G}_0\mathbf{H}_0 + \mathbf{G}_1\mathbf{H}_1 - (\mathbf{G}_0\mathbf{S}_{u,1}\mathbf{S}_{d,1}\mathbf{H}_0 + \mathbf{G}_1\mathbf{S}_{u,0}\mathbf{S}_{d,0}\mathbf{H}_1)\\ &= c^2\mathbf{I}_N.\end{aligned} \tag{6}$$

where c is an arbitrary real number, $\mathbf{H}_k = \mathbf{U}H_k(\mathbf{\Lambda})\mathbf{U}^{\top}$ is the kth filter in the analysis bank, and $\mathbf{G}_k = \mathbf{U}G_k(\mathbf{\Lambda})\mathbf{U}^{\top}$ is one in the synthesis bank, in which

$$\begin{aligned}H_k(\mathbf{\Lambda}) = \mathrm{diag}(H_k(\lambda_0), H_k(\lambda_1), \dots, H_k(\lambda_{N-1}))\\ G_k(\mathbf{\Lambda}) = \mathrm{diag}(G_k(\lambda_0), G_k(\lambda_1), \dots, G_k(\lambda_{N-1})).\end{aligned} \tag{7}$$

For perfect reconstruction, the *spectral folding* term in (6) must be zero. As a result, the CS graph filter bank with the vertex domain sampling must satisfy the following conditions:

$$\begin{aligned}G_0(\lambda)H_0(\lambda) + G_1(\lambda)H_1(\lambda) = c^2\\ G_0(\lambda)H_0(2-\lambda) - G_1(\lambda)H_1(2-\lambda) = 0.\end{aligned} \tag{8}$$

Based on this perfect reconstruction condition, several methods for yielding (near) perfect reconstruction graph filter banks have been proposed [9, 10, 15, 18, 25].

4. CS GRAPH FILTER BANKS WITH SPECTRAL DOMAIN SAMPLING

4.1. Framework

The proposed CSSGFB is shown in Fig. 3. As in the graph filter banks for bipartite graphs [9, 10], it contains analysis and synthesis filters, downsampling, and upsampling. It is also similar to its classical signal processing counterpart. However, the definitions of sampling are different from the conventional approach, i.e., our method is fully designed in the graph spectral domain. Additionally, we do not make any assumptions on the graph used and the normalized graph Laplacian does not have to be used.

Sampling matrices in the spectral domain are defined as

$$\begin{aligned}\widetilde{\mathbf{S}}_{d,0} = \begin{bmatrix}\mathbf{I}_{N/2} & \mathbf{J}_{N/2}\end{bmatrix},\ \widetilde{\mathbf{S}}_{u,0} = \widetilde{\mathbf{S}}_{d,0}^{\top}\\ \widetilde{\mathbf{S}}_{d,1} = \begin{bmatrix}\mathbf{I}_{N/2} & -\mathbf{J}_{N/2}\end{bmatrix},\ \widetilde{\mathbf{S}}_{u,1} = \widetilde{\mathbf{S}}_{d,1}^{\top}.\end{aligned} \tag{9}$$

The downsampling and upsampling matrices for the lowpass branch are the same as the original spectral domain sampling (GD2) and (GU2) introduced in Section 2.2, whereas those in the highpass branch are their modulated versions. As a result, signals after the analysis and synthesis transforms are, respectively, represented as follows.

$$\begin{aligned}\boldsymbol{f}_k = \mathbf{U}_{1,k}\widetilde{\mathbf{S}}_{d,k}H_k(\mathbf{\Lambda})\mathbf{U}_0^{\top}\boldsymbol{f}\\ \widehat{\boldsymbol{f}}_k = \mathbf{U}_0G_k(\mathbf{\Lambda})\widetilde{\mathbf{S}}_{u,k}\mathbf{U}_{1,k}^{\top}\boldsymbol{f}_k,\end{aligned} \tag{10}$$

where $\mathbf{U}_{1,k}$ is an arbitrary eigenvector matrix for the kth subband. Since our framework is fully designed in the spectral domain, $\mathbf{U}_{1,k}$ is usually not required. Additionally, for multi-level transform, the transformed coefficients are not required to transform back into the vertex domain in each level. This is because the inverse GFT of the previous level and the GFT in the current level is cancelled. If one needs the vertex domain coefficients in each branch, they can be obtained by appropriately defining the graph (and the graph Laplacian) of the corresponding size.

4.2. Perfect Reconstruction Condition

From the above structure, the perfect reconstruction condition is expressed as follows.

Theorem 1. *The two-channel CSSGFB defined in the previous subsection is a perfect reconstruction transform, i.e.,* $\widehat{\boldsymbol{f}} = c^2\boldsymbol{f}$ *where* c *is an arbitrary real number, if the graph spectral responses of the filters satisfy the following relationship for all* i.

$$G_0(\lambda_i)H_0(\lambda_i) + G_1(\lambda_i)H_1(\lambda_i) = c^2 \tag{11}$$
$$G_0(\lambda_i)H_0(\lambda_{N-i-1}) - G_1(\lambda_i)H_1(\lambda_{N-i-1}) = 0. \tag{12}$$

Proof. From the definition, $\widehat{\boldsymbol{f}}$ can be written as:

$$\begin{aligned}\widehat{\boldsymbol{f}} =& \mathbf{U}_0G_0(\mathbf{\Lambda})\widetilde{\mathbf{S}}_{u,0}\widetilde{\mathbf{S}}_{d,0}H_0(\mathbf{\Lambda})\mathbf{U}_0^{\top}\boldsymbol{f}\\ &+ \mathbf{U}_0G_1(\mathbf{\Lambda})\widetilde{\mathbf{S}}_{u,1}\widetilde{\mathbf{S}}_{d,1}H_1(\mathbf{\Lambda})\mathbf{U}_0^{\top}\boldsymbol{f}.\end{aligned} \tag{13}$$

Since $\mathbf{U}_0$ is an orthogonal matrix, if the transfer matrix

$$\mathbf{T}_s := G_0(\mathbf{\Lambda})\widetilde{\mathbf{S}}_{u,0}\widetilde{\mathbf{S}}_{d,0}H_0(\mathbf{\Lambda}) + G_1(\mathbf{\Lambda})\widetilde{\mathbf{S}}_{u,1}\widetilde{\mathbf{S}}_{d,1}H_1(\mathbf{\Lambda}) \tag{14}$$

is the identity matrix (up to scaling factor), the whole transform is perfect reconstruction. By substituting (9) into (14), $\mathbf{T}_s$ is rewritten as

$$
\begin{aligned}
\mathbf{T}_s &= G_0(\mathbf{\Lambda}) \begin{bmatrix}\mathbf{I}_{N/2} & \mathbf{J}_{N/2}\end{bmatrix}^\top \begin{bmatrix}\mathbf{I}_{N/2} & \mathbf{J}_{N/2}\end{bmatrix} H_0(\mathbf{\Lambda}) \\
&\quad + G_1(\mathbf{\Lambda}) \begin{bmatrix}\mathbf{I}_{N/2} & -\mathbf{J}_{N/2}\end{bmatrix}^\top \begin{bmatrix}\mathbf{I}_{N/2} & -\mathbf{J}_{N/2}\end{bmatrix} H_1(\mathbf{\Lambda}) \\
&= G_0(\mathbf{\Lambda})(\mathbf{I}_N + \mathbf{J}_N)H_0(\mathbf{\Lambda}) + G_1(\mathbf{\Lambda})(\mathbf{I}_N - \mathbf{J}_N)H_1(\mathbf{\Lambda}) \\
&= G_0(\mathbf{\Lambda})H_0(\mathbf{\Lambda}) + G_1(\mathbf{\Lambda})H_1(\mathbf{\Lambda}) \\
&\quad + \left(G_0(\mathbf{\Lambda})H_0(\mathbf{\Lambda}') - G_1(\mathbf{\Lambda})H_1(\mathbf{\Lambda}')\right)\mathbf{J}_N,
\end{aligned} \tag{15}
$$

where $\mathbf{\Lambda}' = \mathrm{diag}(\lambda_{N-1}, \ldots, \lambda_0)$. If the filters satisfy (11) and (12),

$$
\begin{aligned}
G_0(\mathbf{\Lambda})H_0(\mathbf{\Lambda}) + G_1(\mathbf{\Lambda})H_1(\mathbf{\Lambda}) &= c^2\mathbf{I}_N \\
G_0(\mathbf{\Lambda})H_0(\mathbf{\Lambda}') - G_1(\mathbf{\Lambda})H_1(\mathbf{\Lambda}') &= \mathbf{0}_N.
\end{aligned} \tag{16}
$$

It leads to $\mathbf{T}_s = c^2\mathbf{I}_N$. □

This perfect reconstruction condition implies that the proposed CSSGFBs can be regarded as a generalized version of the conventional graph filter banks such as graph-QMF [9] and graphBior[2] [10]. Though the set of ideal filters clearly satisfies (11) and (12), the proposed CSSGFBs are more general and sometimes non-ideal filters outperform the ideal one (examples are shown in the next section). Note that the existing approaches with the vertex domain sampling cannot guarantee perfect reconstruction even when the ideal filters are used both in the analysis and synthesis banks.

5. DESIGN EXAMPLES AND APPLICATIONS

As mentioned in the previous section, we have some freedom for filters that satisfy (11) and (12). In this paper, we design an appropriate $H_0(\lambda)$ and remaining filters are set as $H_1(\lambda_i) = H_0(\lambda_{N-i-1})$ and $G_k(\lambda_i) = H_k(\lambda_i)$. Inspired by [15], we utilize frequency responses of time domain filters to design $H_0(\lambda_i)$. First, a real-valued function $H_0^{\text{freq}}(\omega)$ where $\omega \in [0, \pi]$ is obtained from the time domain filter, then $H_0(\lambda_i)$ is calculated according to the eigenvalue distribution of the graph Laplacian. That is, $H_0(\lambda_i) = H_0^{\text{freq}}(\pi i/N)$.

In this paper, we used two $H_0^{\text{freq}}(\omega)$: One is designed based on Haar wavelet (denoted as Haar-CSSGFB) and the other is designed based on Meyer kernel (denoted as Meyer-CSSGFB). The design examples are shown in Fig. 4. The graph used is community graph with $N = 400$ as shown in Fig. 5. As can be seen, the energy of the transform is constant over all λ, so the perfect reconstruction condition is satisfied.

As an example application, we perform nonlinear approximation of graph signals to validate the effectiveness of the proposed CSSGFBs. Along with the Haar- and Meyer-CSSGFBs, we also use the ideal filters (denoted as Ideal-CSSGFB) that are also guaranteed perfect reconstruction with our framework. The synthetic graph signal used is shown in Fig. 5. In the experiment, all of the transforms perform two-level octave-band decomposition, then fractions of coefficients with high magnitudes are kept and remaining coefficients are set to zero.

As previously mentioned, our CSSGFBs can be applied both to the combinatorial and symmetric normalized graph Laplacians. Therefore, we examine two graph Laplacians. Thresholding of the transformed coefficients for the proposed method is done in the spectral domain since our framework is fully designed in the spectral domain as presented in Sec. 4.1. The performance is compared with graphBior [10]. Note that bipartition of the underlying graph is needed for the graphBior, and a coloring-based bipartition [10,26] is used in this paper as suggested by the authors.

The results of the nonlinear approximation are shown in Fig. 5. It is clear that the proposed CSSGFBs with the combinatorial graph Laplacian outperform the graphBior. CSSGFBs with the symmetric normalized graph Laplacian (including the graphBior) present similar performances. In this example, Ideal-CSSGFB is not always the best among the proposed transforms. Indeed the performance depends on the signals and the graphs, but we can use the proposed transforms according to them.

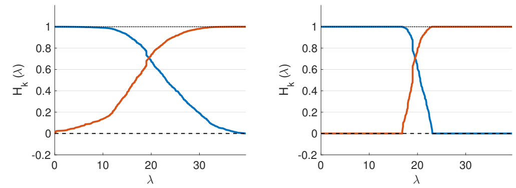

Fig. 4. Design examples of the proposed CSSGFBs. Left: Haar-CSSGFB. Right: Meyer-CSSGFB. Black dotted and dashed lines represent (11) and (12), respectively.

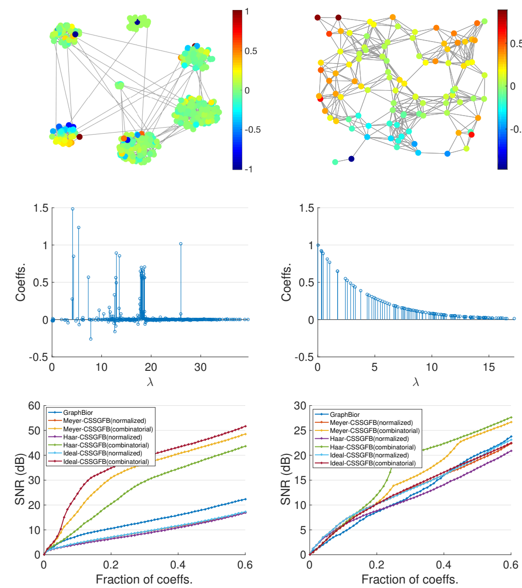

Fig. 5. Original graph signals and results of nonlinear approximation. Top row: Signals in the vertex domain. Middle row: Signals in the graph frequency domain. Bottom row: Results of nonlinear approximation. Left column: Community graph ($N = 400$). Right column: Sensor graph ($N = 100$).

6. CONCLUSIONS

In this paper, a new design method of CSSGFBs was proposed. It is based on the sampling of graph signals in the spectral domain. It is able to satisfy the perfect reconstruction condition for any type of graphs in contrast to the other graph filter banks with vertex domain sampling. Through the experiment on nonlinear approximation, some of the proposed transforms significantly outperform the conventional graph filter bank. Constructing M-channel transforms, developing faster methods, and deriving relationships with conventional graph filter banks are a list of our future work.

[2]We omit the proof due to the limitation of space. It will be available in the journal version of this paper in the future.

7. REFERENCES

[1] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, “The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains,” *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, 2013.

[2] N. Leonardi and D. Van De Ville, “Tight wavelet frames on multislice graphs,” *IEEE Trans. Signal Process.*, vol. 16, pp. 3357–3367, 2013.

[3] S. Ono, I. Yamada, and I. Kumazawa, “Total generalized variation for graph signals,” in *ICASSP’15*, 2015, pp. 5456–5460.

[4] W. Hu, G. Cheung, A. Ortega, and O. C. Au, “Multiresolution graph Fourier transform for compression of piecewise smooth images,” *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 419–433, 2015.

[5] M. Onuki, S. Ono, M. Yamagishi, and Y. Tanaka, “Graph signal denoising via trilateral filter on graph spectral domain,” *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2, pp. 137–148, 2016.

[6] H. Higashi, T. M. Rutkowski, T. Tanaka, and Y. Tanaka, “Multilinear discriminant analysis with subspace constraints for single-trial classification of event-related potentials,” *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 7, pp. 1295–1305, 2016.

[7] K. Yamamoto, M. Onuki, and Y. Tanaka, “Deblurring of point cloud attributes in graph spectral domain,” in *Proc. ICIP’16*, 2016, pp. 1559–1563.

[8] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric deep learning: Going beyond euclidean data,” *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, 2017.

[9] S. K. Narang and A. Ortega, “Perfect reconstruction two-channel wavelet filter banks for graph structured data,” *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2786–2799, 2012. [Online]. Available: http://biron.usc.edu/wiki/index.php/Graph_Filterbanks

[10] ——, “Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs,” *IEEE Trans. Signal Process.*, vol. 61, pp. 4673–4685, 2013. [Online]. Available: http://biron.usc.edu/wiki/index.php/Graph_Filterbanks

[11] D. I. Shuman, B. Ricaud, and P. Vandergheynst, “Vertex-frequency analysis on graphs,” *Applied and Computational Harmonic Analysis*, vol. 40, no. 2, pp. 260–291, 2016.

[12] Y. Tanaka and A. Sakiyama, “M-channel oversampled graph filter banks,” *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3578–3590, 2014.

[13] D. I. Shuman, C. Wiesmeyr, N. Holighaus, and P. Vandergheynst, “Spectrum-adapted tight graph wavelet and vertex-frequency frames,” *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4223–4235, 2015. [Online]. Available: http://documents.epfl.ch/users/s/sh/shuman/www/publications.html

[14] D. B. H. Tay, Y. Tanaka, and A. Sakiyama, “Near orthogonal oversampled graph filter banks,” *IEEE Signal Process. Lett.*, vol. 23, no. 2, pp. 277–281, 2015.

[15] A. Sakiyama, K. Watanabe, and Y. Tanaka, “Spectral graph wavelets and filter banks with low approximation error,” *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 3, pp. 230–245, 2016.

[16] N. Tremblay and P. Borgnat, “Subgraph-based filterbanks for graph signals,” *IEEE Trans. Signal Process.*, vol. 64, no. 15, pp. 3827–3840, 2016.

[17] A. Sakiyama and Y. Tanaka, “Oversampled graph Laplacian matrix for graph filter banks,” *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6425–6437, 2014.

[18] D. B. H. Tay, Y. Tanaka, and A. Sakiyama, “Critically sampled graph filter banks with polynomial filters from regular domain filter banks,” *Signal Processing*, vol. 131, pp. 66–72, 2017.

[19] A. Anis and A. Ortega, “Critical sampling for wavelet filterbanks on arbitrary graphs,” in *Proc. ICASSP’17*, 2017, pp. 3889–3893.

[20] Y. Jin and D. I. Shuman, “An M-channel critically sampled filter bank for graph signals,” in *Proc. ICASSP’17*, 2017, pp. 3909–3913.

[21] V. N. Ekambaram, G. C. Fanti, B. Ayazifar, and K. Ramchandran, “Spline-like wavelet filterbanks for multiresolution analysis of graph-structured data,” *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 4, pp. 268–278, 2015.

[22] M. S. Kotzagiannidis and P. L. Dragotti, “Splines and wavelets on circulant graphs,” *arXiv preprint arXiv:1603.04917*, 2016.

[23] Y. Tanaka, “Spectral domain sampling of graph signals,” *arXiv preprint arXiv: 1706.05147*, 2017.

[24] D. I. Shuman, M. J. Faraji, and P. Vandergheynst, “A multiscale pyramid transform for graph signals,” *IEEE Trans. Signal Process.*, vol. 64, no. 8, pp. 2119–2134, 2016.

[25] D. B. H. Tay, Y. Tanaka, and A. Sakiyama, “Almost tight spectral graph wavelets with polynomial filters,” *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 812–824, 2017.

[26] F. Harary, D. Hsu, and Z. Miller, “The biparticity of a graph,” *J. Graph Theory*, vol. 1, no. 2, pp. 131–133, 1977.