M-Channel Oversampled Graph Filter Banks

Yuichi Tanaka, *Member, IEEE*, and Akie Sakiyama, *Student Member, IEEE*

***Abstract*—This paper proposes M-channel oversampled filter banks for graph signals. The filter set satisfies the perfect reconstruction condition. A method of designing oversampled graph filter banks is presented that allows us to design filters with arbitrary parameters, unlike the conventional critically sampled graph filter banks. The oversampled graph Laplacian matrix is also introduced with a discussion of the entire redundancy of the oversampled graph signal processing system. The practical performance of the proposed filter banks is validated through graph signal denoising experiments.**

***Index Terms*—Graph filter banks, graph signal denoising, graph signal processing, graph wavelets, oversampled filter banks.**

I. Introduction

GRAPH signal processing is one of the emerging topics in signal processing [1]–[12]. Unlike regular signal processing[1], graph signal processing must explicitly consider the structure of the signal. In this context, well-studied signals, e.g., acoustic, image, and video signals, can be considered to be graph signals with very simple structures, and for this reason, tools for graph signal processing have received much attention.

As we describe later, graph signal processing in graph spectral domain is based on a transformation using the eigenvectors of the *graph Laplacian matrix* (GLM), which determines the signal structure. Spectral graph theory [13] has been used to study the relationships among the eigenvalues (or eigenvectors) of the GLM, and the signal processing community has focused on features that relate to *diagonalizing* a given matrix. For example, the diagonalization power of the discrete cosine transform (DCT) was summarized in [14]. Moreover, it was proven that DCT is the optimal basis for image/video processing with graphs of simple intra/inter-prediction [11].

Graph wavelet transforms have been developed for signals with structures. Some transforms on graph vertex domain require simplifications to be made to the graph, such as decomposition into even and odd nodes [5], [6]. Recently, wavelet transforms on graph spectral domain have been proposed [2]–[4]. A key technique of (digital) signal processing is down and upsampling, and it has been also studied for graph signals in the context of the *spectral folding phenomenon* [2], [3], which is analogous to the *aliasing* effect of regular signal processing. Since the spectral folding phenomenon affects bipartite graphs, the filter banks are designed to be *two-channel critically sampled* ones. Unfortunately, these designs have strong limitations imposed on them if they are to be used to obtain critically sampled perfect reconstruction filter banks. In contrast, M-channel ($M > 2$) filter banks would be useful for graph signals, since oversampled filter banks for regular signals have more freedom in their design and it has been shown that they outperform critically sampled systems in several applications [15]–[20].

Here, we study such graph filter banks and show that they do not have the limitations of critically sampled ones. For instance, perfect reconstruction is possible even if we use an *arbitrary* lowpass filter, and the filters we design have good stopband attenuation. Furthermore, we discover where the graph signals are oversampled and derive the perfect reconstruction condition for the oversampled case. We also introduce an oversampling scheme that uses an *oversampled graph Laplacian matrix*. To the best of our knowledge, it is the first attempt at using an oversampled graph Laplacian matrix for graph signal processing. Moreover, our recent work [21]–[23] describes graph expansion techniques in more detail.

As a possible application, we show how our oversampled graph filter bank can be used to *denoise graph signals*. A comparison of implementations of a simple hard-thresholding technique shows that the proposed filter bank outperforms the existing graph filter banks and the wavelet transform for regular signals.

The remaining part of this paper is organized as follows. Preliminaries and notations are summarized in the rest of this section. Section II reviews the existing studies. The oversampled graph Laplacian matrix and redundancy of transforms are studied in Section III. Section IV gives the perfect reconstruction condition of oversampled graph filter banks, and Section V presents a detailed design method. Design examples and experimental results on graph signal decomposition are shown in Section VI. Section VII concludes the paper.

A. Preliminaries and Notations

In this paper, we consider a finite undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V}$ and $\mathcal{E}$ represent sets of nodes and edges in the graph, respectively. Similar to [2], [3], we assume a graph without self-loops or multiple connections. The number of nodes is $N = |\mathcal{V}|$, unless specified otherwise. The (m, n)-th element in the $N \times N$ adjacency matrix $\mathbf{A}$ is defined as follows:

$$a_{mn} = \begin{cases} w_{mn} & \text{if nodes } m \text{ and } n \text{ are connected,} \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

Manuscript received September 04, 2013; revised December 27, 2013 and revised March 24, 2014; accepted May 24, 2014. Date of publication June 04, 2014; date of current version June 25, 2014. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Trac D. Tran. This work was supported in part by JSPS KAKENHI Grant Number 24760288 and MEXT Tenure Track Promotion Program. *Yuichi Tanaka and Akie Sakiyama contributed equally to this work.*

The authors are with the Graduate School of BASE, Tokyo University of Agriculture and Technology, 2-24-16 Naka-cho, Koganei, Tokyo, 184-8588 Japan (e-mail: ytnk@cc.tuat.ac.jp; sakiyama@msp-lab.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2014.2328983

[1]In this paper, we often use the word *regular* signals/transforms to distinguish discrete signals on uniform points without explicit structures from the *graph* signals discussed in this paper.

where w_{mn} is a weight on the edge. The diagonal degree matrix $\mathbf{D}$ contains $d_{mm} = \sum_n a_{mn}$. With $\mathbf{A}$ and $\mathbf{D}$, the unnormalized GLM is defined as $\mathbf{L} := \mathbf{D} - \mathbf{A}$. We will consider the symmetric normalized GLM (SNGLM) $\boldsymbol{\mathcal{L}} := \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$, since the normalized version is always restricted to having eigenvalues in $[0, 2]$ and it is required to correctly reflect the current graph downsampling phenomenon [2], [3]. Our filter bank is also applicable without any changes to the random-walk GLM $\boldsymbol{\mathcal{L}}_r := \mathbf{D}^{-1}\mathbf{L}$ [3].

The important symbols for the paper are listed below:

1) $\boldsymbol{f}$: Graph signal ($\boldsymbol{f} \in \mathbb{R}^N$).
2) $\boldsymbol{u}_{\lambda_i}$: i-th Eigenvector of $\boldsymbol{\mathcal{L}}$.
3) λ_i: i-th eigenvalue of $\boldsymbol{\mathcal{L}}$ ($\boldsymbol{\mathcal{L}}\boldsymbol{u}_{\lambda_i} = \lambda_i \boldsymbol{u}_{\lambda_i}$), where $0 = \lambda_0 < \lambda_1 \leq \cdots \leq \lambda_{N-2} \leq \lambda_{N-1} \leq 2$, and $\lambda_{N-1} = 2$ for bipartite graphs.
4) $\sigma(\boldsymbol{\mathcal{L}})$: Spectrum of the graph, i.e., $\sigma(\boldsymbol{\mathcal{L}}) := \{\lambda_0, \ldots, \lambda_{N-1}\}$.

II. Review

Here, we briefly review the existing approaches using graph wavelets and filter banks.

A. Graph Fourier Transform and Spectral Domain Graph Filter

Analogous to regular signal processing, the graph Fourier transform (GFT) of $\boldsymbol{f}$ is defined as follows [4]:

$$\bar{f}(\lambda_i) = \langle \boldsymbol{u}_{\lambda_i}, \boldsymbol{f} \rangle = \sum_{n=0}^{N-1} u^*_{\lambda_i}(n) f(n). \tag{2}$$

where $\cdot^*$ is the complex conjugate. Filtering in the spectral domain, i.e., the GFT domain, is able to control the spectral and vertex domain spread. As a result, recent studies on graph filters have focused on spectral domain designs.

The eigenspace projection matrix is defined as follows:

$$\mathbf{P}_{\lambda_i} := \sum_{\lambda=\lambda_i} \boldsymbol{u}_\lambda \boldsymbol{u}_\lambda^T \tag{3}$$

where $\cdot^T$ is the transpose of a matrix or a vector. Note that the $\boldsymbol{u}_{\lambda_i}$ are orthogonal to each other. That is,

$$\mathbf{P}_{\lambda_i}\mathbf{P}_{\lambda_j} = \delta(\lambda_i - \lambda_j)\mathbf{P}_{\lambda_i}, \tag{4}$$

where $\delta(\cdot)$ is the Kronecker delta function, or, equivalently,

$$\mathbf{U}\mathbf{U}^T = \mathbf{I}_N \tag{5}$$

where $\mathbf{U} = [\boldsymbol{u}_{\lambda_0}, \ldots, \boldsymbol{u}_{\lambda_{N-1}}]$ and $\mathbf{I}_N$ is an identity matrix of size N. By using $\mathbf{P}_{\lambda_i}$, a spectral domain filter for graph signals can be defined as follows:

$$\mathbf{H}_c = \sum_{\lambda_i \in \sigma(\boldsymbol{\mathcal{L}})} h_c(\lambda_i)\mathbf{P}_{\lambda_i} \tag{6}$$

where $h_c(\lambda)$ is the kernel of $\mathbf{H}_c$.

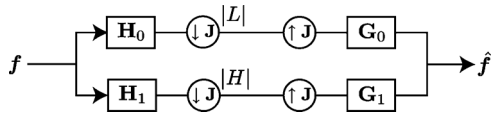

Fig. 1. Two-channel critically sampled graph filter bank.

B. Critically Sampled Graph Filter Banks

The downsampling and upsampling effects of the SNGLM of a bipartite graph have been studied in [2]. Note that an arbitrary graph can be decomposed into $K_{\mathcal{G}}$ disjoint bipartite graphs. Let us define a bipartite graph $\mathcal{G} = \{L, H, \mathcal{E}\}$ where nodes in $\mathcal{G}$ are divided into two disjoint sets L and H. We call the nodes in L the lowpass channel and those in H the highpass channel, for the sake of convenience.

Similar to the case of regular signals, the downsampling-then-upsampling operation can be defined as follows:

$$\begin{aligned} \mathbf{D}_{du,L} &= \frac{1}{2}(\mathbf{I}_N - \mathbf{J}) \\ \mathbf{D}_{du,H} &= \frac{1}{2}(\mathbf{I}_N + \mathbf{J}) \end{aligned} \tag{7}$$

where $\mathbf{J}$ is a diagonal matrix with *antipodal* coefficients, i.e., $+1$ or -1, with the following form:

$$\mathbf{J}_{mm} = \begin{cases} +1 & \text{if } f(m) \text{ belongs to } H, \\ -1 & \text{if } f(m) \text{ belongs to } L. \end{cases} \tag{8}$$

The following proposition is the key to designing the perfect reconstruction graph filter banks.

Proposition 1: (Downsampling Phenomenon of Bipartite Graph [2, Proposition 1]) The eigenspace projection matrix $\mathbf{P}_{\lambda_i}$ and the downsampling matrix $\mathbf{J}$ are related as follows:

$$\mathbf{J}\mathbf{P}_{\lambda_i} = \mathbf{P}_{2-\lambda_i}\mathbf{J}. \tag{9}$$

Two sorts of critically sampled perfect reconstruction graph filter banks have been proposed. One is *graph-QMF* [2], which is the orthogonal solution, and the other is *graphBior* [3], which guarantees perfect reconstruction by relaxing the orthogonal property of the graph-QMF.

Fig. 1 illustrates the entire transformation for one bipartite graph. It is similar to the regular filter banks [24]–[26], but the number of signals in the lowpass and highpass channels are no longer $N/2$ (or, more formally, $\lfloor N/2 \rfloor$ or $\lceil N/2 \rceil$): the lowpass channel contains $|L|$ signals whereas the highpass channel retains $|H|$ signals, where $|L| + |H| = N$. The number of signals in each channel is determined on the basis of the graph-coloring result.

Both graph-QMF and graphBior are designed to satisfy the following perfect reconstruction condition:

$$\begin{aligned} \mathbf{T} &= \frac{1}{2}\mathbf{G}_0(\mathbf{I} - \mathbf{J})\mathbf{H}_0 + \frac{1}{2}\mathbf{G}_1(\mathbf{I} + \mathbf{J})\mathbf{H}_1 \\ &= \frac{1}{2}(\mathbf{G}_0\mathbf{H}_0 + \mathbf{G}_1\mathbf{H}_1) + \frac{1}{2}(\mathbf{G}_1\mathbf{J}\mathbf{H}_1 - \mathbf{G}_0\mathbf{J}\mathbf{H}_0) = \mathbf{I}_N. \end{aligned} \tag{10}$$

where $\mathbf{H}_k = \sum_{\lambda_i \in \sigma(\boldsymbol{\mathcal{L}})} h_k(\lambda_i)\mathbf{P}_{\lambda_i}$ is a filter in the analysis bank and $\mathbf{G}_k = \sum_{\lambda_i \in \sigma(\boldsymbol{\mathcal{L}})} g_k(\lambda_i)\mathbf{P}_{\lambda_i}$ is one in the synthesis bank. In (10), the second term is called the *spectral folding* term, and it corresponds to *aliasing* in regular signals. Therefore, this

spectral folding term must be zero. As a result, a critically sampled perfect reconstruction graph filter bank must satisfy the following conditions:

$$\begin{aligned} g_0(\lambda)h_0(\lambda) + g_1(\lambda)h_1(\lambda) &= 2 \\ -g_0(\lambda)h_0(2-\lambda) + g_1(\lambda)h_1(2-\lambda) &= 0. \end{aligned} \tag{11}$$

For graph-QMF, one prototype lowpass filter $h_0(\lambda)$ is used to yield the remaining $h_1(\lambda)$, $g_0(\lambda)$, and $g_1(\lambda)$. Moreover, $h_0(\lambda)$ has to satisfy $h_0^2(\lambda)+h_0^2(2-\lambda) = c^2$, where c is some constant. However, such an $h_0(\lambda)$ cannot be an exact polynomial: hence, perfect reconstruction and orthogonality are not possible [2], [3]. Instead, graphBior is based on the spectral factorizations of a maximally flat filter pair, which is a similar approach to that of the Cohen-Daubechies-Feauveau wavelet transform [27], and the designed filter bank satisfies (10).

III. Oversampling Graph Signals

In this section, we describe the actual oversampled position in the signal processing flow of spectral graph signal processing and explicitly show the redundancy of the oversampled graph filter bank. In this section, we assume that the original arbitrary graph has already been decomposed into K_G disjoint bipartite graphs. The oversampling process can be done independently of each bipartite subgraph.

A. Oversampled SNGLM

Let us define the original GLM of a bipartite graph as $\mathbf{L}_0$ and its corresponding adjacency matrix $\mathbf{A}_0$ whose size is $N_0 \times N_0$. Without loss of generality, an *oversampled SNGLM* $\widetilde{\boldsymbol{\mathcal{L}}}$ can be represented as follows:

$$\widetilde{\boldsymbol{\mathcal{L}}} = \widetilde{\mathbf{D}}^{-1/2}\widetilde{\mathbf{L}}\widetilde{\mathbf{D}}^{-1/2} \tag{12}$$

where

$$\widetilde{\mathbf{L}} = \widetilde{\mathbf{D}} - \widetilde{\mathbf{A}} \tag{13}$$

$$\widetilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_0 & \mathbf{A}_{01} \\ \mathbf{A}_{01}^T & \mathbf{0}_{N_1-N_0} \end{bmatrix}, \tag{14}$$

in which $\widetilde{\mathbf{A}}$ is the oversampled adjacency matrix whose size is $N_1 \times N_1$ and $\widetilde{\mathbf{D}}$ is a degree matrix that normalizes the new GLM. Additionally, $\mathbf{A}_{01}$ contains information on the connection between the original graph and the appended nodes so that $\widetilde{\boldsymbol{\mathcal{L}}}$ is still a bipartite graph.

With the oversampled SNGLM, we can transform the oversampled signal $\widetilde{\boldsymbol{f}}$, which is represented as

$$\widetilde{\boldsymbol{f}} = \begin{bmatrix} \boldsymbol{f}_0 \\ \boldsymbol{f}_1 \end{bmatrix}, \tag{15}$$

where $\boldsymbol{f}_0$ is the original signal and $\boldsymbol{f}_1$ is the signal for additional nodes and its length is $N_1 - N_0$. Since the expansion process of the signal (and its corresponding SNGLM) is independent of the filter selection, the entire transform is a perfect reconstruction only if the graph filter bank satisfies the perfect reconstruction condition. Filtering in the spectral domain is defined similarly to the critically sampled case [1]–[3]:

$$\mathbf{H}_o = \sum_{\lambda_i \in \sigma(\widetilde{\boldsymbol{\mathcal{L}}})} h_o(\lambda_i)\widetilde{\mathbf{P}}_{\lambda_i}, \tag{16}$$

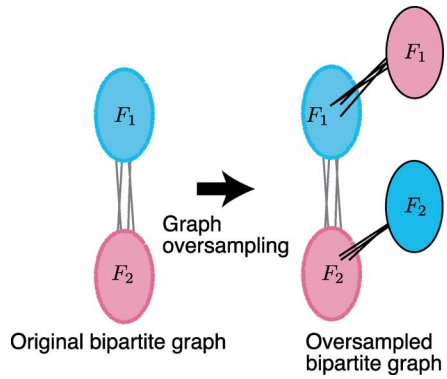

Fig. 2. Graph oversampling of a bipartite graph. Gray lines indicate edges in the original graph and black lines represent appended edges.

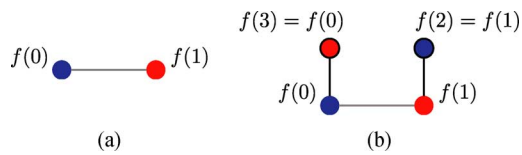

Fig. 3. Toy example of graph oversampling. (a) Scenario 1: Two-node-graph. (b) Scenario 2: Oversampled two-node-graph. The black lines are appended edges.

where $\widetilde{\mathbf{P}}_{\lambda_i}$ is the eigenspace projection matrix corresponding to $\widetilde{\boldsymbol{\mathcal{L}}}$ in (12).

B. Examples of Graph Oversampling

An effective graph expansion method is beneficial for analyzing graph signals. Note that the expanded signal $\boldsymbol{f}_1$ can be freely chosen and the choice of $\mathbf{A}_{01}$, i.e., how to connect $\boldsymbol{f}_1$ with $\boldsymbol{f}_0$, is also arbitrary. However, an inappropriate choice of $\mathbf{A}_{01}$ and/or $\boldsymbol{f}_1$ will cause a performance loss compared to the original GLM. One possible recommendation, studied in our recent papers [21], [22], is that $\boldsymbol{f}_1$ is to be (a part of) *duplicated copy* of $\boldsymbol{f}_0$ and $\mathbf{A}_{01}$ is determined according to the edges in the original graph.

An example of graph oversampling for a bipartite graph is illustrated in Fig. 2. In the figure, F_1 and F_2 represent sets of colored nodes and gray lines indicate the original edges. The original bipartite graph is oversampled by adding the nodes *just above* F_1 and F_2 and connecting the appended nodes with corresponding original nodes. The effective oversampling methods for arbitrary (non-bipartite) graphs are studied in [21], [22]. Its strategy is used for the experiments in Section VI of the oversampled graph filter banks proposed in this paper.

Furthermore, we would like to mention that *the output signal after filtering with the oversampled SNGLM is completely different from that obtained with undecimated graph filtering*. To introduce the fact, we assume two following scenarios:

Scenario 1: Filtering using the *original* graph *without* downsampling.

Scenario 2: Filtering using the *oversampled* graph *with* downsampling.

Here, a toy example is considered: a two-node-graph illustrated in Fig. 3(a). All weights on edges are 1 and the original graph signal is $\boldsymbol{f}_0 = [f(0)\ f(1)]^T$. We use this graph for the scenario 1. For the scenario 2, the simplest oversampled graph

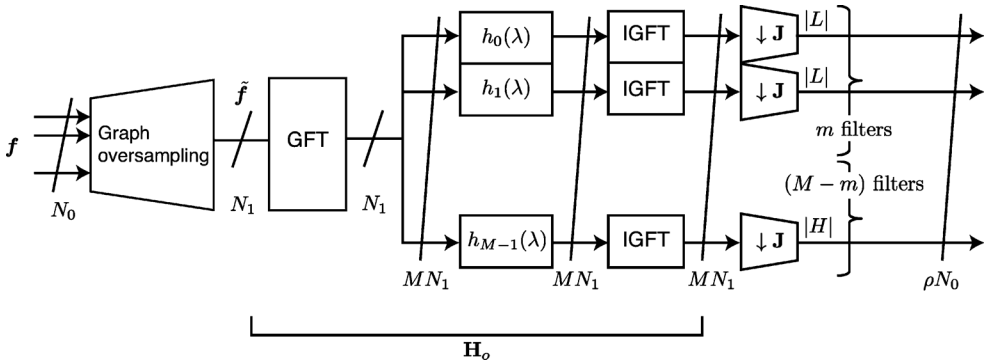

Fig. 4. Signal processing flow of oversampled graph filter bank. Symbols just below the skewed lines indicate the number of signals at typical positions.

corresponding to Fig. 2 is used and it is shown in Fig. 3(b). Obviously, the oversampled graph is still a bipartite graph. The oversampled graph signal in this example is defined as the duplicated copy of $\boldsymbol{f}_0$, i.e., $\widetilde{\boldsymbol{f}} = [f(0)\ f(1)\ f(1)\ f(0)]^T$.

The original GLM is represented as

$$\mathbf{L}_0 = \boldsymbol{\mathcal{L}}_0 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{17}$$

Its eigenvalues and eigenvectors are clearly $\{\lambda_0, \lambda_1\} = \{0, 2\}$ and

$$\mathbf{U}_0 = [\,\boldsymbol{u}_{\lambda_0}\ \boldsymbol{u}_{\lambda_1}\,] = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The oversampled GLM is

$$\widetilde{\mathbf{L}} = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}, \tag{18}$$

and its normalized version is

$$\widetilde{\boldsymbol{\mathcal{L}}} = \begin{bmatrix} 1 & -1/2 & 0 & -1/\sqrt{2} \\ -1/2 & 1 & -1/\sqrt{2} & 0 \\ 0 & -1/\sqrt{2} & 1 & 0 \\ -1/\sqrt{2} & 0 & 0 & 1 \end{bmatrix}. \tag{19}$$

Its eigenvalues are $\{\widetilde{\lambda}_0, \widetilde{\lambda}_1, \widetilde{\lambda}_2, \widetilde{\lambda}_3\} = \{0, 1/2, 3/2, 2\}$ and

$$\widetilde{\mathbf{U}} = \frac{1}{\sqrt{6}}\begin{bmatrix} \sqrt{2} & 1 & -1 & \sqrt{2} \\ \sqrt{2} & -1 & -1 & -\sqrt{2} \\ 1 & -\sqrt{2} & \sqrt{2} & 1 \\ 1 & \sqrt{2} & \sqrt{2} & -1 \end{bmatrix}.$$

For the scenario 1, the output signal after filtering with $h_0(\lambda)$ can be represented as

$$\begin{aligned} \boldsymbol{f}_0' &= \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} h_0(0) & \\ & h_0(2) \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} f(0) \\ f(1) \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} h_0(0) & h_0(2) \\ h_0(0) & -h_0(2) \end{bmatrix}\begin{bmatrix} f(0)+f(1) \\ f(0)-f(1) \end{bmatrix}. \end{aligned} \tag{20}$$

As a result, it captures the signal characteristics at $\lambda_i = 0, 2$ (the minimum and maximum eigenvalues).

Whereas for the scenario 2, we can observe the different filter response. The filtered and downsampled signal for the scenario 2 can be represented as follows:

$$\begin{aligned} \widetilde{\boldsymbol{f}}' &= \frac{1}{6}\begin{bmatrix} \sqrt{2} & 1 & -1 & \sqrt{2} \\ 1 & -\sqrt{2} & \sqrt{2} & 1 \end{bmatrix} \\ &\quad\times \begin{bmatrix} h_0(0) & & & \\ & h_0(\frac{1}{2}) & & \\ & & h_0(\frac{3}{2}) & \\ & & & h_0(2) \end{bmatrix} \\ &\quad\times \begin{bmatrix} \sqrt{2} & \sqrt{2} & 1 & 1 \\ 1 & -1 & -\sqrt{2} & \sqrt{2} \\ -1 & -1 & \sqrt{2} & \sqrt{2} \\ \sqrt{2} & -\sqrt{2} & 1 & -1 \end{bmatrix}\begin{bmatrix} f(0) \\ f(1) \\ f(1) \\ f(0) \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{2}c_0h_0(0) - c_1h_0(\frac{3}{2}) & c_0h_0(\frac{1}{2}) + \sqrt{2}c_1h_0(2) \\ c_0h_0(0) + \sqrt{2}c_1h_0(\frac{3}{2}) & -\sqrt{2}c_0h_0(\frac{1}{2}) + c_1h_0(2) \end{bmatrix} \\ &\quad\times \begin{bmatrix} f(0)+f(1) \\ f(0)-f(1) \end{bmatrix}, \end{aligned} \tag{21}$$

where $c_0 = \frac{1}{6}(\sqrt{2}+1)$ and $c_1 = \frac{1}{6}(\sqrt{2}-1)$. In this example, the extra eigenvalues $\widetilde{\lambda}_i = 1/2, 3/2$ are used for the signal analysis. Roughly speaking, finer characteristics of the graph signal can be captured by oversampling the graph signal and the underlying graph.

C. M-Channel Filter Banks for Bipartite Graph and Redundancy of Oversampled Graph Filter Banks

In summary, there are two possible ways to oversample $\boldsymbol{f}_0$: oversampling at 1) the oversampled SNGLM in (12) and/or 2) M spectral filters $h_k(\lambda)$ ($k = 0, \ldots, M-1$). The situation is illustrated in Fig. 4. Let $m < M$ be the number of filters which keep $|L|$ signals after downsampling. The overall redundancy ρ can be expressed as

$$\begin{aligned} \rho &= \frac{m|L| + (M-m)|H|}{N_0} \\ &= \frac{m|L| + (M-m)(N_1 - |L|)}{N_0}, \end{aligned} \tag{22}$$

where $|H| = N_1 - |L|$. For example, when $M = 4, m = 2$, and $|L| = |H| = N_0/2$, i.e., using a four-channel filter bank with the original SNGLM ($N_1 = N_0$), $\rho = 2$.

IV. Perfect Reconstruction Condition of Oversampled Graph Filter Banks

The details of the perfect reconstruction condition are discussed in this section. For clearer understanding, we present the case of $M = 4$ first, and extend it to any value of M afterwards.

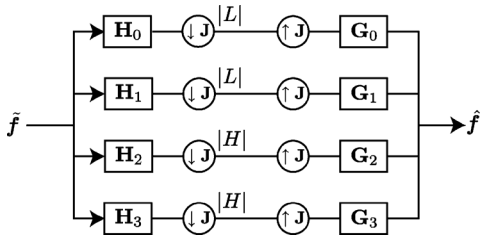

Fig. 5. Oversampled graph filter bank.

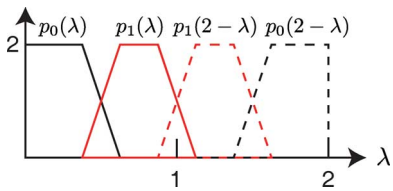

Fig. 6. Four-channel product filter example.

A. Four-Channel Case

Consider the four-channel graph filter bank shown in Fig. 5. After filtering with $\mathbf{H}_k$, the zeroth and first channels pass $|L|$ signals, whereas the second and third ones keep $|H|$ signals. $\widehat{\boldsymbol{f}}_k$ is represented as

$$\begin{aligned}\widehat{\boldsymbol{f}}_k &= \frac{1}{2}\mathbf{G}_k(\mathbf{I}-\mathbf{J})\mathbf{H}_k\widetilde{\boldsymbol{f}} \quad k=0,1\\ \widehat{\boldsymbol{f}}_k &= \frac{1}{2}\mathbf{G}_k(\mathbf{I}+\mathbf{J})\mathbf{H}_k\widetilde{\boldsymbol{f}} \quad k=2,3.\end{aligned} \tag{23}$$

where

$$\begin{aligned}\mathbf{G}_k &= \sum_{\lambda_i\in\sigma(\widetilde{\boldsymbol{\mathcal{L}}})} g_k(\lambda_i)\widetilde{\mathbf{P}}_{\lambda_i}\\ \mathbf{H}_k &= \sum_{\lambda_i\in\sigma(\widetilde{\boldsymbol{\mathcal{L}}})} h_k(\lambda_i)\widetilde{\mathbf{P}}_{\lambda_i}.\end{aligned}$$

Therefore, the overall transfer function $\mathbf{T}$ is

$$\begin{aligned}\mathbf{T} = &\frac{1}{2}\sum_{\lambda_i}\sum_{k=0}^{3} g_k(\lambda_i)h_k(\lambda_i)\widetilde{\mathbf{P}}_{\lambda_i}\\ &+\frac{1}{2}\sum_{\lambda_i,\gamma_j}\{g_2(\lambda_i)h_2(\gamma_j)+g_3(\lambda_i)h_3(\gamma_j)\\ &-g_0(\lambda_i)h_0(\gamma_j)-g_1(\lambda_i)h_1(\gamma_j)\}\widetilde{\mathbf{P}}_{\lambda_i}\mathbf{J}\widetilde{\mathbf{P}}_{\gamma_j}.\end{aligned} \tag{24}$$

From Proposition 1, $\widetilde{\mathbf{P}}_{\lambda_i}\mathbf{J}\widetilde{\mathbf{P}}_{\gamma_j} = \widetilde{\mathbf{P}}_{\lambda_i}\widetilde{\mathbf{P}}_{2-\gamma_j}\mathbf{J}$ and the orthogonality of $\widetilde{\mathbf{P}}_{\lambda_i}$, we get

$$\begin{aligned}\mathbf{T} = &\frac{1}{2}\sum_{\lambda_i}\sum_{k=0}^{3} g_k(\lambda_i)h_k(\lambda_i)\widetilde{\mathbf{P}}_{\lambda_i}\\ &+\frac{1}{2}\sum_{\lambda_i}\{g_2(\lambda_i)h_2(2-\lambda_i)+g_3(\lambda_i)h_3(2-\lambda_i)\\ &-g_0(\lambda_i)h_0(2-\lambda_i)-g_1(\lambda_i)h_1(2-\lambda_i)\}\widetilde{\mathbf{P}}_{\lambda_i}\mathbf{J}.\end{aligned} \tag{25}$$

As a result, the perfect reconstruction condition becomes

$$\sum_{k=0}^{3} g_k(\lambda)h_k(\lambda) = 2 \tag{26}$$

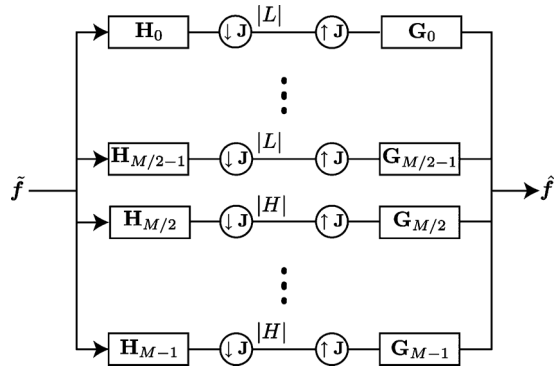

Fig. 7. M-channel oversampled graph filter bank.

and

$$\begin{aligned}&g_2(\lambda)h_2(2-\lambda)+g_3(\lambda)h_3(2-\lambda)\\ &\quad -g_0(\lambda)h_0(2-\lambda)-g_1(\lambda)h_1(2-\lambda)=0\end{aligned} \tag{27}$$

for any λ. Equation (27) is satisfied if we use the constraints $g_0(\lambda)=h_2(2-\lambda), g_1(\lambda)=h_3(2-\lambda), g_2(\lambda)=h_0(2-\lambda)$, and $g_3(\lambda)=h_1(2-\lambda)$ (similar to what is done in [3]). Accordingly, (26) becomes

$$\begin{aligned}&g_0(\lambda)h_0(\lambda)+g_0(2-\lambda)h_0(2-\lambda)\\ &\quad +g_1(\lambda)h_1(\lambda)+g_1(2-\lambda)h_1(2-\lambda)=2.\end{aligned} \tag{28}$$

Let us define a product filter as $p_k(\lambda)=g_k(\lambda)h_k(\lambda)$. Finally, (28) can be rewritten as

$$p_0(\lambda)+p_0(2-\lambda)+p_1(\lambda)+p_1(2-\lambda)=2. \tag{29}$$

By using this perfect reconstruction condition, we can select *four-channel* product filters instead of two-channel systems of the critically sampled graph filter bank. The situation is shown in Fig. 6.

B. General M-Channel Case

Let us assume that an oversampled graph filter bank has M channels, where M is even. Additionally, we assume that $M/2$ filters keep $|L|$ signals and the other ones keep $|H|$ signals, as shown in Fig. 7. Similar to (25), the transfer function can be calculated as

$$\begin{aligned}\mathbf{T} = &\frac{1}{2}\sum_{\lambda_i}\sum_{k=0}^{M-1} g_k(\lambda_i)h_k(\lambda_i)\widetilde{\mathbf{P}}_{\lambda_i}\\ &+\frac{1}{2}\sum_{\lambda_i}\sum_{k=0}^{M/2-1}\{-g_k(\lambda_i)h_k(2-\lambda_i)\\ &+g_{k+M/2}(\lambda_i)h_{k+M/2}(2-\lambda_i)\}\widetilde{\mathbf{P}}_{\lambda_i}\mathbf{J}.\end{aligned} \tag{30}$$

The perfect reconstruction condition is

$$\sum_{k=0}^{M-1} g_k(\lambda)h_k(\lambda) = 2 \tag{31}$$

$$\sum_{k=0}^{M/2-1} -g_k(\lambda)h_k(2-\lambda)+g_{k+M/2}(\lambda)h_{k+M/2}(2-\lambda) = 0 \tag{32}$$

for any λ. The latter equation is valid if we choose $g_k(\lambda) = h_{k+M/2}(2-\lambda)$ and $g_{k+M/2}(\lambda) = h_k(2-\lambda)$. Accordingly, (31) becomes

$$\sum_{k=0}^{M/2-1} g_k(\lambda)h_k(\lambda) + g_k(2-\lambda)h_k(2-\lambda) = 2. \tag{33}$$

As a result, the product filter $p_k(\lambda)$ must satisfy the following condition:

$$\sum_{k=0}^{M/2-1} p_k(\lambda) + p_k(2-\lambda) = 2. \tag{34}$$

V. Design of M-Channel Oversampled Graph Filter Bank

First, we will consider the case of $M = 4$. Let us define $q(\lambda) = p_0(\lambda) + p_1(\lambda)$. Equation (29) can be rewritten as

$$q(\lambda) + q(2-\lambda) = 2. \tag{35}$$

This equation is the same as that of a two-channel biorthogonal graph filter bank [3]. Therefore, the design problem boils down to separating the critically sampled product filter $q(\lambda)$ into lowpass and bandpass (Fig. 6) filters $p_0(\lambda)$ and $p_1(\lambda)$ such that the *sum* of filters is $q(\lambda)$.

Let us assume that a lowpass product filter $p_0(\lambda)$ is arbitrarily chosen so that $h_0(\lambda)$ and $g_0(\lambda)$ are "good" lowpass filters. By changing the variable of $\lambda = 1+l$ [3], $p_0(1+l)$ can be expressed as

$$p_0(1+l) = (1+l)^K \left(\sum_{m=0}^{J_0} \alpha_m l^m \right), \tag{36}$$

where α_m is an arbitrary parameter.

Following [3], $q(1+l)$ has the degree $2K-1$ and its even degree must be zero from the halfband condition.[2] Hence, $q(1+l)$ is represented as

$$q(1+l) = (1+l)^K \left(1 + \sum_{m=1}^{K-1} r_m l^m \right) = 1 + \sum_{n=0}^{K-1} c_{2n+1} l^{2n+1}, \tag{37}$$

which is, of course, the same as that in [3] and has a unique solution satisfying (35). Finally, the remaining product filter $p_1(\lambda)$ can be defined as follows:

$$p_1(1+l) = q(1+l) - p_0(1+l) = (1+l)^K \left(\sum_{m=0}^{J_1} \beta_m l^m \right). \tag{38}$$

Example: $K = 2$ zeros at $l = -1$

This is the same example as in [3]. We assume $J_0 = 1$. As in (36) and (38), $p_0(1+l)$ and $p_1(1+l)$ are

$$p_0(1+l) = (1+l)^2 (\alpha_0 + \alpha_1 l) \tag{39}$$

$$p_1(1+l) = (1+l)^2 (\beta_0 + \beta_1 l), \tag{40}$$

where α_0 and α_1 are arbitrarily chosen parameters. Then, the sum of the product filter $q(1+l)$ is defined as

$$q(1+l) = \frac{1}{2}(1+l)^2(2-l), \tag{41}$$

[2]Although *Proposition 1* in [3] restricts $q(\lambda)$ to being a *product* of two kernels, it is nonetheless applicable to the *sum* of two kernels assumed in this paper.

which is an odd-order polynomial and it is the same product filter as that in [3]. To guarantee the perfect reconstruction, β_0 and β_1 must be

$$\beta_0 = 1 - \alpha_0$$
$$\beta_1 = -\left(\alpha_1 + \frac{1}{2} \right). \tag{42}$$

That is, we can add free parameters (α_0 and α_1) to design a halfband filter, and this will lead to better filter characteristics.

A similar derivation is possible for general M-channel graph filter banks. In that case, the parameters for $(M-2)/2$ product filters can be freely chosen, and the last product filter can be designed so that the entire product filter $q(\lambda)$ is a maximally flat halfband filter.

Remark 1: The above design method yields perfect reconstruction graph filter banks. Unfortunately, the filter selection similar to graph-QMF [2] cannot obtain the perfect reconstruction filter set with the real-valued exact polynomial filters even for this oversampled case. This is easily confirmed by examining the transfer function in (30). If the filter bank is chosen similar to graph-QMF, the condition $g_k(\lambda) = h_k(\lambda)$ must be satisfied. Therefore, the perfect reconstruction condition (34) becomes

$$\sum_{k=0}^{M/2-1} h_k^2(\lambda) + h_k^2(2-\lambda) = 2. \tag{43}$$

Here, let us focus on the highest degree of $\sum_{k=0}^{M/2-1} h_k^2(\lambda)$. Indeed, each $h_k^2(\lambda)$ has an even highest degree. However, as previously mentioned, $\sum_{k=0}^{M/2-1} h_k^2(\lambda)$ must be an odd degree polynomial. Hence, at least one $h_k^2(\lambda)$ has a complex coefficient to cancel the highest degree. This means real-valued oversampled graph filter banks with the filter selection similar to graph-QMF cannot be designed.

Remark 2: Let us define $\mathbf{T}_a$ to be the matrix form of the analysis transform. Riesz bounds, which give the lower and upper bounds A and B such that $A\|\boldsymbol{f}\|^2 \le \|\mathbf{T}_a \boldsymbol{f}\|^2 \le B\|\boldsymbol{f}\|^2$, of the analysis filter bank can be calculated similarly to what is shown in [3], [4], as

$$\begin{aligned}\mathbf{T}_a^T \mathbf{T}_a = {} & \frac{1}{2} \sum_{\lambda_i} \sum_{k=0}^{M-1} h_k^2(\lambda_i) \widetilde{\mathbf{P}}_{\lambda_i} \\ & + \frac{1}{2} \sum_{\lambda_i} \sum_{k=0}^{M/2-1} \{ -h_k(\lambda_i) h_k(2-\lambda_i) \\ & + h_{k+M/2}(\lambda_i) h_{k+M/2}(2-\lambda_i) \} \widetilde{\mathbf{P}}_{\lambda_i} \mathbf{J}. \end{aligned} \tag{44}$$

Practically speaking, the second term in (44) can be ignored since the first term is much larger than the second one. That is, the Riesz bounds can be approximated as

$$A \sim \inf_{\lambda} \frac{1}{2} \sum_{k=0}^{M-1} h_k^2(\lambda)$$
$$B \sim \sup_{\lambda} \frac{1}{2} \sum_{k=0}^{M-1} h_k^2(\lambda). \tag{45}$$

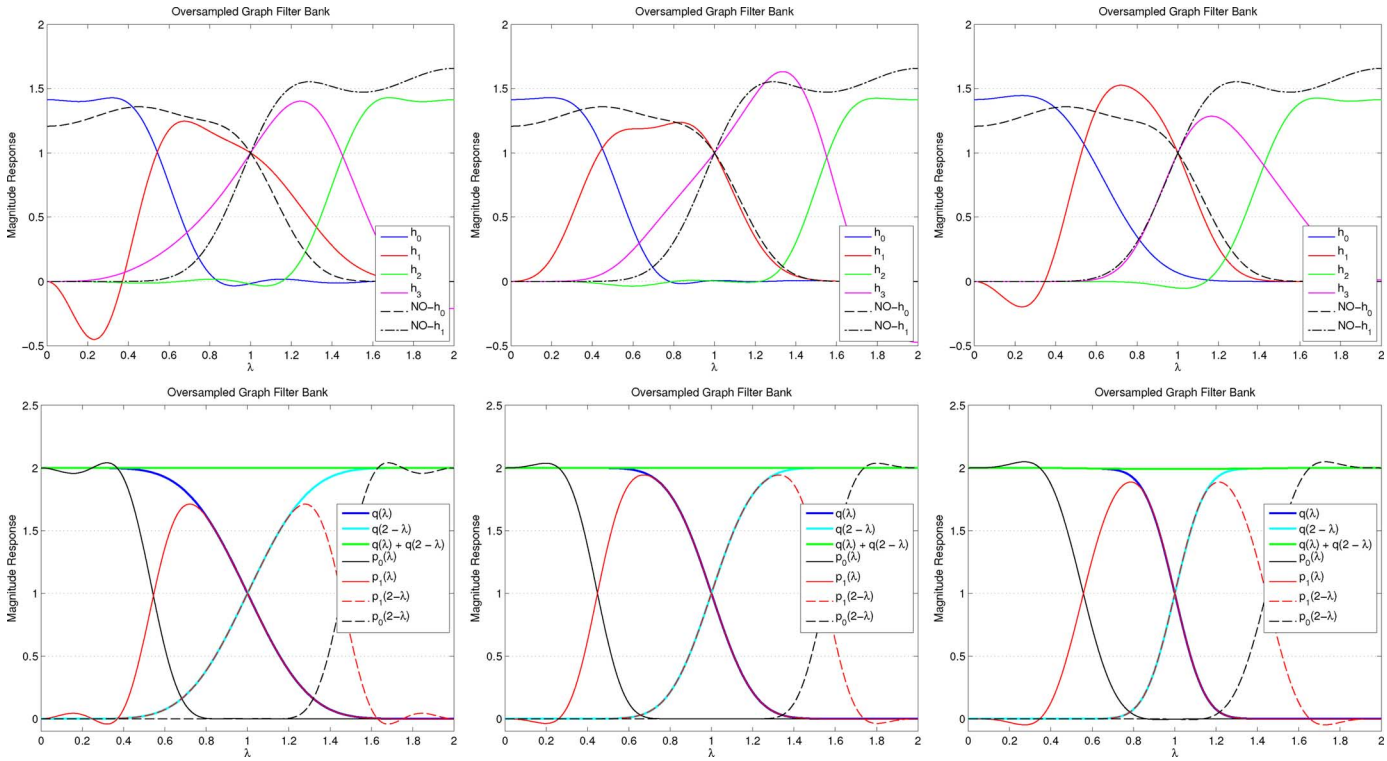

Fig. 8. Four-channel oversampled graph filter banks. From left to right: $(k_0, k_1) = (2, 2)$, $(4, 4)$, and $(8, 8)$. Top row: analysis filter bank. Black lines indicate graphBior(6,6) [3]. Bottom row: halfband filters.

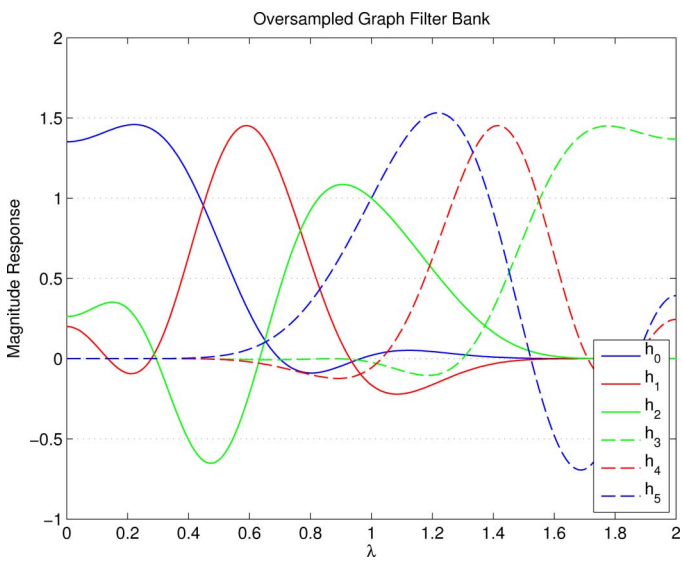

Fig. 9. Six-channel oversampled graph filter bank: analysis bank.

This result is the similar to what is shown in [3].

VI. Design Examples and Experimental Results

In this section, we show the design methodology of M-channel oversampled graph filter banks and a few design examples.

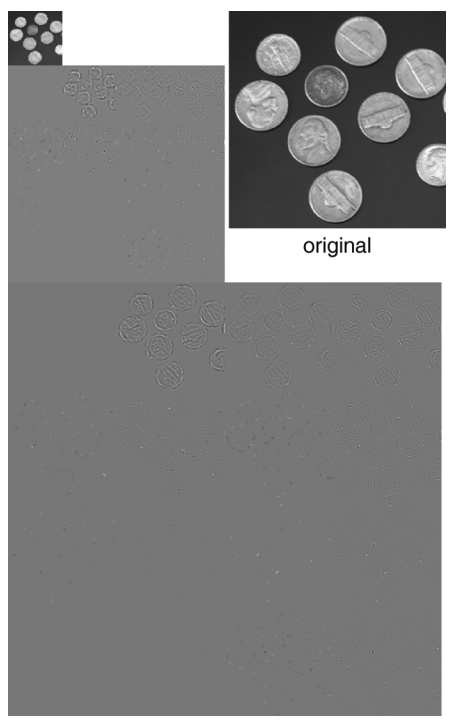

Fig. 10. Multiresolution *Coins* image after three-level decomposition using the oversampled graph filter bank. The original image on the same scale is shown at the top right. The values of the transformed coefficients are scaled to be in the range $[0, 1]$ for the sake of visualization.

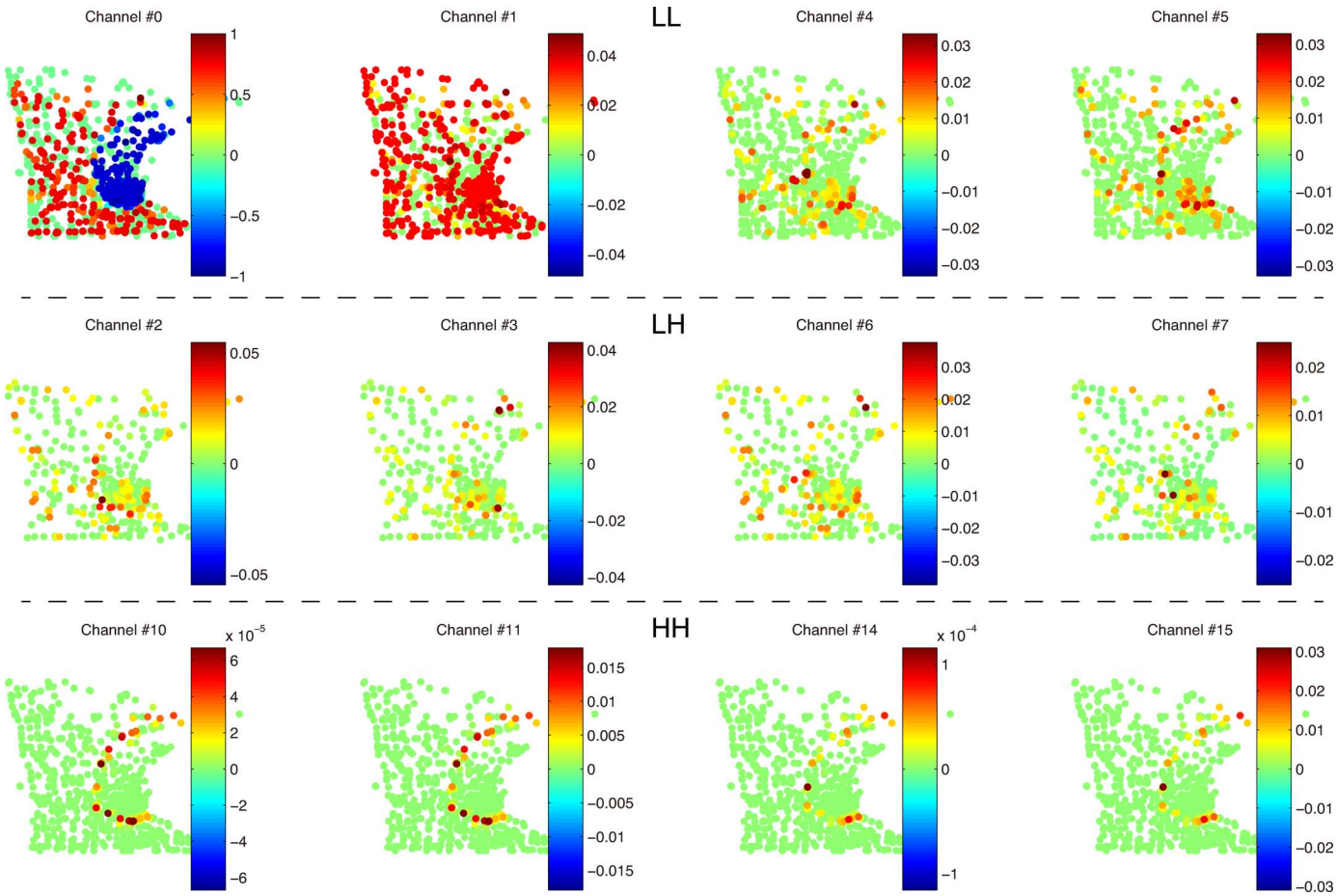

Fig. 11. Graphs decomposed by the proposed oversampled graph filter bank. (Colors are adjusted according to each channel for the sake of visualization.) Original signal is shown in Fig. 13(a). We use a two-dimensional four-channel filter bank leading to $4^2 = 16$ channels. Note that the graph is three-colorable: therefore, channels 8, 9, 12, and 13 (corresponding to the HL channel for the critically sampled filter banks) are empty.

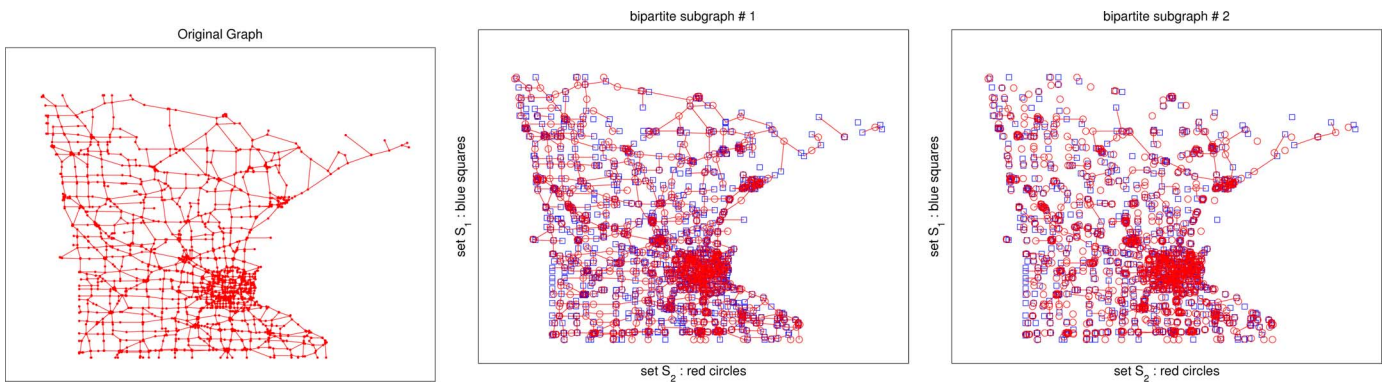

Fig. 12. Structure of the *Minnesota Traffic Graph*. It was reproduced from the MATLAB code of Narang and Ortega [3], and Harary's algorithm [2], [28] was used to yield two bipartite subgraphs. From left to right: Original graph, bipartite graph #1, and bipartite graph #2.

A. Design Methodology

As mentioned above, we can use arbitrary parameters to design filters. In what follows, we will use a sequential design method to obtain good filter banks:

1) Design $h_k(1-l)$ and $g_k(1-l)$ $(k = 0, \ldots, M/2-2)$ with k_0 and k_1 zeros (where $K = k_0 + k_1$ in (36)–(37)) at $l = 1$ $(\lambda = 2)$. They are represented as follows:

$$h_k(1-l) = (1-l)^{k_0} \sum_{m=0}^{J_k^{(h)}} s_{h,k,m} l^m$$

$$g_k(1-l) = (1-l)^{k_1} \sum_{m=0}^{J_k^{(g)}} s_{g,k,m} l^m \tag{46}$$

where $s_{h,k,m}$ and $s_{g,k,m}$ are filter coefficients. i.e., the product filter $p_k(1-l) = g_k(1-l)h_k(1-l)$ can be represented as

$$p_k(1-l) = (1-l)^K \left(\sum_{m=0}^{J_k^{(h)}+J_k^{(g)}} \alpha_{k,m} l^m \right). \tag{47}$$

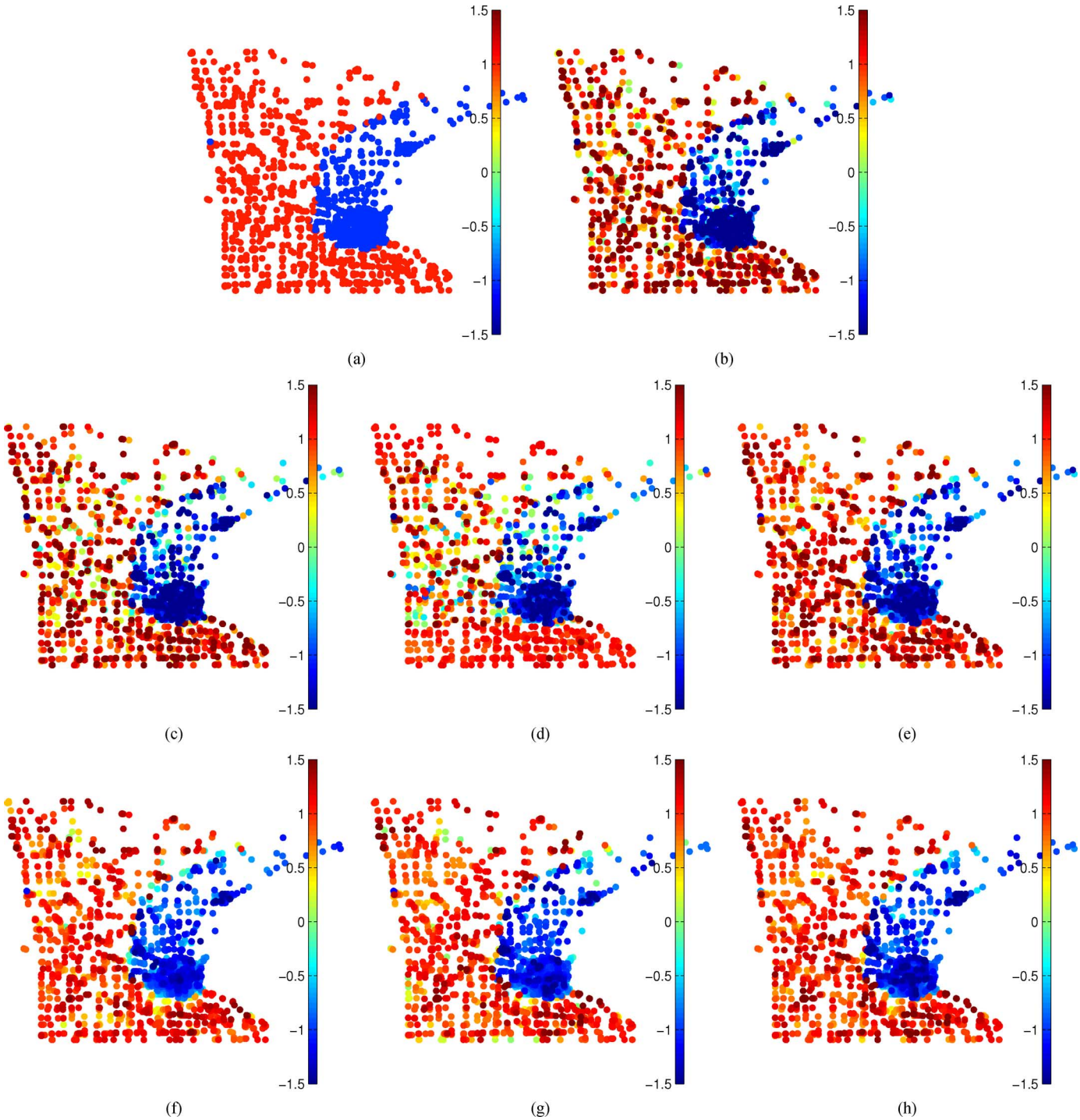

Fig. 13. Denoising results of Example 1: (a) input signal; (b) noisy observation; (c) sym8 (1 level); (d) sym8 (5 levels); (e) graphBior(6, 6); (f) SGWT; (g) OSGFB with CSGLM; and (h) OSGFB with OSGLM.

The numbers of arbitrary parameters in $h_k(1-l)$ and $g_k(1-l)$ are $J_k^{(h)}$ and $J_k^{(g)}$, respectively.

The filters are optimized by using the cost function of the stopband attenuation shown below:

$$C(h_k) = w_0 \int_{l\in\omega_p} (\sqrt{2} - h_k(1-l))^2 dl + w_1 \int_{l\in\omega_s} h_k^2(1-l) dl, \quad (48)$$

where w_0 and w_1 are weights and ω_p and ω_s are defined as the passband and stopband $(-1 \le \omega_p, \omega_s \le 1)$, respectively.

2) Calculate the two-channel halfband filter pair $q(1-l) = q(\lambda)$ and $q(1+l) = q(2-\lambda)$ with K zeros at $l = 1$ so that the pair satisfies (35).

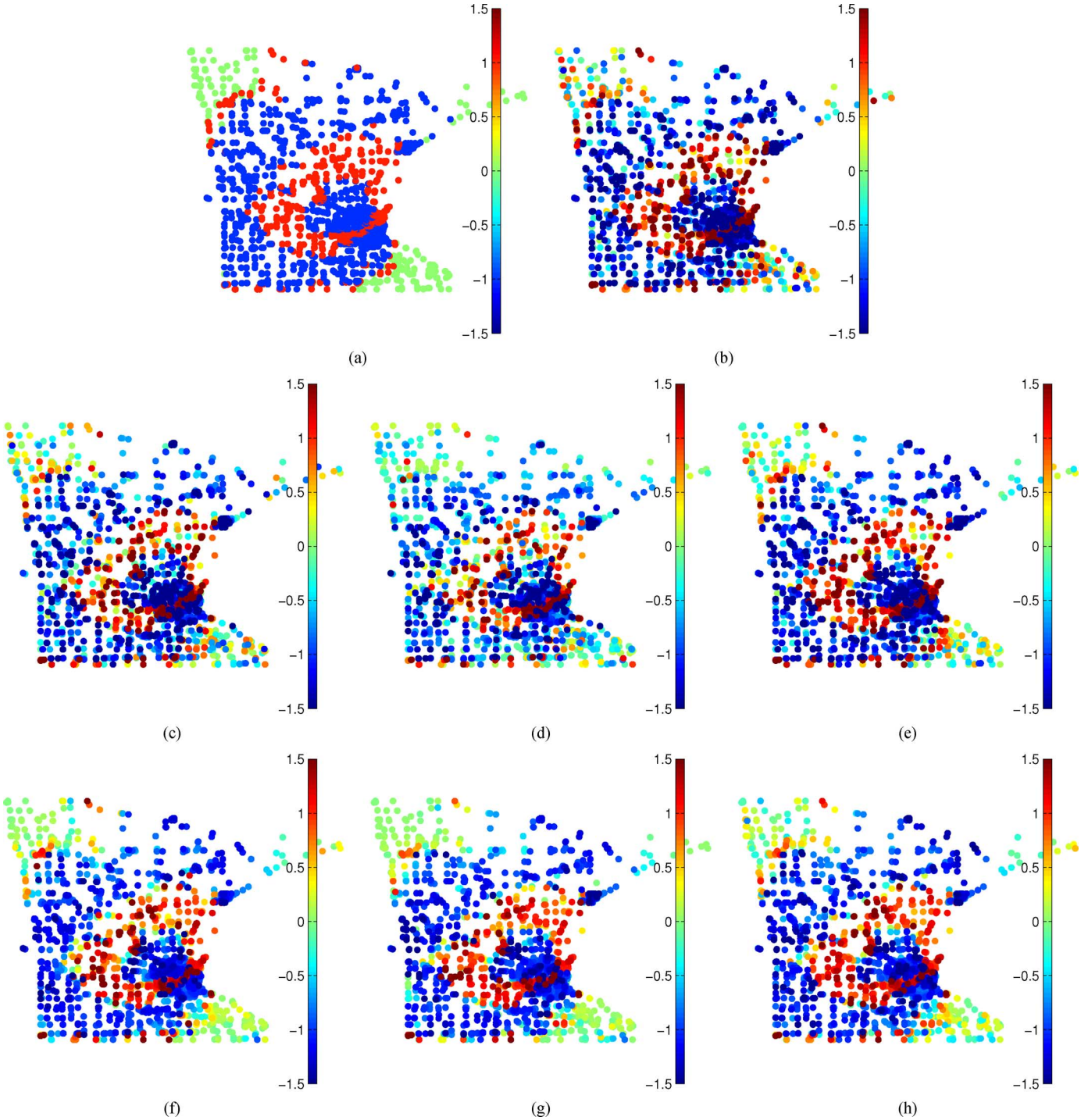

Fig. 14. Denoising results of Example 2: (a) Input signal; (b) Noisy observation; (c) sym8 (1 level); (d) sym8 (5 levels); (e) graphBior(6, 6); (f) SGWT; (g) OSGFB with CSGLM; and (h) OSGFB with OSGLM.

3) Calculate the bandpass product filter[3]

$$p_{\frac{M}{2}-1}(1-l) = q(1-l) - \sum_{k=0}^{M/2-2} p_k(1-l)$$

$$= (1-l)^K \tilde{p}_{\frac{M}{2}-1}(1-l). \quad (49)$$

4) Factorize $p_{\frac{M}{2}-1}(1-l)$ into two bandpass filters $h_{\frac{M}{2}-1}(1-l)$ and $g_{\frac{M}{2}-1}(1-l)$. Test all combinations of roots as long as both bandpass filters have real-valued coefficients, and select the best combination, i.e., the filters minimizing $C\left(h_{\frac{M}{2}-1}\right) + C\left(g_{\frac{M}{2}-1}\right)$.

Fig. 8 shows an example of oversampled graph filter banks. The arbitrary lowpass filters $h_0(\lambda)$ and $g_0(\lambda)$ are designed to

[3]There always exists $q(1-l)$ which satisfies the perfect reconstruction condition (35) [3]. Therefore, $p_{\frac{M}{2}-1}(1-l)$ also has a unique solution with real coefficients as long as all of the arbitrary design filters have real coefficients. Additionally, since the perfect reconstruction condition is only imposed on $p_{\frac{M}{2}-1}(1-l)$, $J_k^{(h)}$ and $J_k^{(g)}$ in (46) can be set arbitrarily regardless of K.

TABLE I
DENOISED RESULTS OF MINNESOTA TRAFFIC GRAPH (AVERAGE OF TEN EXECUTIONS): SNR (dB)

	σ	noisy	sym8 (1 level)	sym8 (5 levels)	graphBior	SGWT	OSGFB (CSGLM)	OSGFB (OSGLM)
Example 1	1/32	30.04	30.02	30.06	31.25	33.24	**34.64**	33.89
	1/16	24.08	24.24	24.03	25.66	27.73	**28.77**	28.52
	1/8	18.07	18.59	17.96	20.16	22.22	21.82	**22.99**
	1/4	12.06	12.00	11.02	14.25	15.19	15.36	**17.03**
	1/2	6.05	6.26	5.74	8.51	10.29	10.31	**11.42**
	1	0.00	1.60	3.05	2.77	**8.82**	4.18	6.01
Example 2	1/32	29.45	28.69	28.24	29.13	29.17	**30.30**	30.10
	1/16	23.44	22.81	22.00	23.45	23.08	24.54	**24.84**
	1/8	17.41	16.64	15.31	17.91	17.27	18.09	**19.26**
	1/4	11.44	9.99	8.52	11.90	11.59	11.99	**13.08**
	1/2	5.40	4.40	3.31	6.49	7.35	7.32	**8.15**
	1	-0.64	0.27	0.94	1.54	**4.72**	3.22	4.16
Redundancy	–		1.00	1.00	1.00	4.00	4.00	3.21

have degree 10 and 11, respectively. We used $-1 \leq \omega_p \leq -0.84$ and $-0.75 \leq \omega_s \leq 1$. For comparison, the frequency responses of the critically sampled graphBior(6, 6) [3] are also plotted. They have 13-taps for the lowpass filter and 12-taps for the highpass filter. It is clear that our oversampled lowpass filter has a sharper transition band and a more uniform response in the passband than the critically sampled graph filter banks have. In the following experiments, we use the oversampled filter bank with $(k_0, k_1) = (4, 4)$ zeros.

Additionally, Fig. 9 presents a six-channel oversampled graph filter bank with $(k_0, k_1) = (4, 4)$ zeros and $\{J_h, J_g\} = \{7, 8\}$. The filter lengths are 11 or 12 taps. For lowpass filters, we used $-1 \leq \omega_p \leq -0.93$ and $-0.83 \leq \omega_s \leq 1$. For bandpass ones, $-0.83 \leq \omega_p \leq -0.75$ and $-1 \leq \omega_s \leq -0.95 \wedge -0.65 \leq \omega_s \leq 1$ are used. It is clear that our sequential design methodology can be used for the general M-channel case and the frequency responses of the filters are well localized.

B. Graph Signal Decomposition

In this experiment, we assume that $\boldsymbol{f}_0$ is oversampled only at the M-channel filter kernels: i.e., we use $\widetilde{\boldsymbol{f}} = \boldsymbol{f}_0$ and $\widetilde{\boldsymbol{\mathcal{L}}} = \boldsymbol{\mathcal{L}}$ for the sake of clarity. The experiment using an oversampled SNGLM is presented in the next subsection. Since the filters designed in this paper are exact polynomials, the filtered signal can be efficiently computed by using Chebychev polynomials [4]. Therefore, an explicit computation of the entire set of eigenvalues and eigenvectors of $\boldsymbol{\mathcal{L}}$ or $\widetilde{\boldsymbol{\mathcal{L}}}$ is not required.

Figs. 10 and 11 show the decomposed results of the graph signals using the proposed oversampled graph filter bank. For the *Coins* image shown in Fig. 10, the eight-connected image graph is decomposed into a rectangular bipartite subgraph and a diagonal bipartite subgraph. Its further details can be found in [2], [3]. The structure of the *Minnesota Traffic Graph*, whose decomposition is presented in Fig. 11, is shown in Fig. 12. It is three-colorable: thus, the HL channel is empty and not shown. It is clear that the decomposed graph signals are well localized and different channels extract different signal characteristics.

C. Denoising of Graph Signal

Here, we show the potential ability of using oversampled graph filter banks to remove additive white Gaussian noise from graph signals of the *Minnesota Traffic Graph*. The oversampled graph filter bank is compared with graphBior(6, 6) [3], the spectral graph wavelet transform (SGWT) with three scales [4] and a regular one-dimensional wavelet *sym8*, which can be found in the Wavelet Toolbox in MATLAB. For the regular wavelet transform, the input signal $\boldsymbol{f}_0$ is treated as a vector, and one-level and five-level dyadic decompositions are performed. Only one level transform is used for the graph filter banks. All methods retain the lowest-frequency subband and the remaining high-frequency subbands are hard-thresholded with $T = 3\sigma$, where σ is the standard deviation of noise.

We tested two setups for the oversampled graph filter bank. One uses the critically sampled SNGLM $\boldsymbol{\mathcal{L}}$ (abbreviated as OSGFB with CSGLM): the multidimensional decomposition introduced in [2] is performed on the two bipartite subgraphs shown in Fig. 12. The other uses the oversampled SNGLM $\widetilde{\boldsymbol{\mathcal{L}}}$ (abbreviated as OSGFB with OSGLM): the oversampled SNGLM can be *one bipartite graph containing all edges in the original graph* [21], [22]. Therefore, by using the latter approach, a multidimensional decomposition does not have to be performed on the graph. Furthermore, two signal examples, shown in Figs. 13(a) and 14(a) were tested. Both structures, i.e., the original GLM, are the same, but the values on the nodes are different.

Table I summarizes the denoising performances together with the redundancies of the transforms. As expected, the graph filter banks perform much better than the regular wavelet transform. Furthermore, our oversampled graph filter bank outperforms graphBior by 1–3 dB in SNR. The SGWT performs better for the strong noise case $\sigma = 1$, whereas the proposed oversampled graph filter banks are better than the SGWT for the other σ. Interestingly, the OSGFB with OSGLM sometimes outperformed the OSGFB with CSGLM and the SGWT in spite of it having less redundancy. The effectiveness of the OSGLM is thus experimentally validated.

Moreover, the OSGFB with OSGLM is comparable to the SGWT for Example 2 even for the strong noise case. Both transforms have four filters: however, the frequency partitions are different. The SGWT belongs to a class of *nonuniform* graph filter banks, whereas our filter bank is a *uniform* one. Indeed, the performance depends on the signal characteristics. The graph Fourier spectra of both examples are shown in Fig. 15. Obviously, the spectrum of Example 1 is more concentrated at the

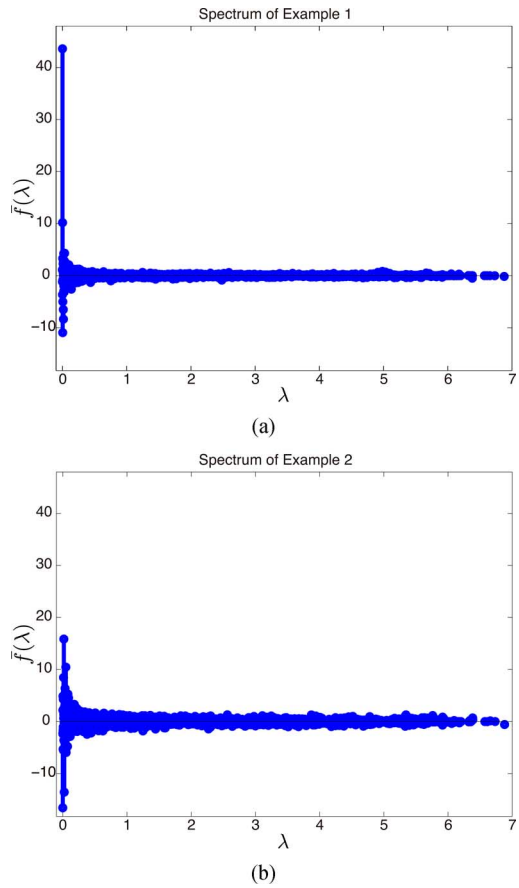

Fig. 15. Graph Fourier spectra of *Minnesota Traffic Graph*. Since the experiment uses the unnormalized GLM, the maximum value of λ is not restricted to be 2. We utilized the code by Shuman *et al.* in [12]. (a) Example 1. (b) Example 2.

low λ values than that of Example 2. This characteristic is responsible for the good denoising performance of the SGWT with $\sigma = 1$ in Example 1.

The denoised signals of Example 1 and 2 for $\sigma = 1/2$ are shown in Figs. 13 and 14, respectively. Since the regular wavelet transform does not consider the structure of signals explicitly, the signals are over-smoothened across the boundary of the center and surrounding areas; many blue points appear in the surrounding area. In contrast, graph filter banks preserve the solid boundary. It is clear that the proposed filter bank performs better than the critically sampled one. It is well-known in signal/image processing circles that the oversampled filter banks are better than the critically sampled ones for signal analysis. The experiment showed this to be the case for graph signal processing.

VII. Conclusions

We presented a method of designing M-channel oversampled filter banks for graph signals. It satisfies the perfect reconstruction condition and allows us to use arbitrary parameters, unlike critically sampled graph filter banks. Furthermore, it was shown to outperform other transforms, including regular wavelet transforms, in a graph signal denoising experiment.

Acknowledgment

MATLAB code examples are available at http://tanaka.msp-lab.org/software.

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] 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.

[3] S. K. Narang and A. Ortega, "Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4673–4685, 2013.

[4] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.

[5] 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. ICML*, 2010, pp. 367–374.

[6] G. Shen and A. Ortega, "Transform-based distributed data gathering," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3802–3815, 2010.

[7] W. Wang and K. Ramchandran, "Random multiresolution representations for arbitrary sensor network graphs," in *Proc. ICASSP*, 2006, pp. IV-161–IV-164.

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

[9] R. R. Coifman and M. Maggioni, "Diffusion wavelets," *Appl. Comput. Harmon Anal.*, vol. 21, no. 1, pp. 53–94, 2006.

[10] N. Leonardi and D. Van De Ville, "Tight wavelet frames on multislice graphs," *IEEE Trans. Signal Process.*, vol. 16, no. 13, pp. 3357–3367, 2013.

[11] C. Zhang and D. Florêncio, "Analyzing the optimality of predictive transform coding using graph-based models," *IEEE Signal Process. Lett.*, vol. 20, no. 1, pp. 106–109, 2012.

[12] D. I. Shuman, C. Wiesmeyr, N. Holighaus, and P. Vandergheynst, "Spectrum-adapted tight graph wavelet and vertex-frequency frames," *arXiv preprint*, 2013 [Online]. Available: http://documents.epfl.ch/users/s/sh/shuman/www/publications.html, arXiv:1311.0897

[13] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI, USA: Amer. Math. Soc., 1997, vol. 92, CBMS Regional Conference Series in Mathematics.

[14] V. Sánchez, P. Garcia, A. M. Peinado, J. C. Segura, and A. J. Rubio, "Diagonalizing properties of the discrete cosine transforms," *IEEE Trans. Signal Process.*, vol. 43, no. 11, pp. 2631–2641, 1995.

[15] Z. Cvetković and M. Vetterli, "Oversampled filter banks," *IEEE Trans. Signal Process.*, vol. 46, no. 5, pp. 1245–1255, May 1998.

[16] P. L. Dragotti, J. Kovacevic, and V. K. Goyal, "Quantized oversampled filter banks with erasures," in *Proc. IEEE Data Compress. Conf.*, 2001, pp. 173–182.

[17] L. Gan and K.-K. Ma, "Oversampled linear-phase perfect reconstruction filterbanks: Theory, lattice structure and parameterization," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 744–759, 2003.

[18] F. Labeau, L. Vandendorpe, and B. Macq, "Structures, factorizations, and design criteria for oversampled paraunitary filterbanks yielding linear-phase filters," *IEEE Trans. Signal Process.*, vol. 48, no. 11, pp. 3062–3071, 2000.

[19] T. Tanaka and Y. Yamashita, "The generalized lapped pseudo-biorthogonal transform: Oversampled linear-phase perfect reconstruction filterbanks with lattice structures," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 434–446, 2004.

[20] Y. Tanaka, M. Ikehara, and T. Q. Nguyen, "Higher-order feasible building blocks for lattice structure of oversampled linear-phase perfect reconstruction filter banks," *Signal Process.*, vol. 89, no. 9, pp. 1694–1703, 2009.

[21] A. Sakiyama and Y. Tanaka, "Oversampled graph Laplacian matrix for graph filter banks," *IEEE Trans. Signal Process.*, 2014, submitted for publication.

[22] A. Sakiyama and Y. Tanaka, "Oversampled graph Laplacian matrix for graph signals," in *Proc. EUSIPCO*, 2014, to be presented.

[23] A. Sakiyama and Y. Tanaka, "Edge-aware image graph expansion methods for oversampled graph Laplacian matrix," in *Proc. ICIP*, 2014, to be presented.

[24] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.
[25] M. Vetterli and J. Kovačevic, *Wavelets and Subband Coding*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.
[26] G. Strang and T. Q. Nguyen, *Wavelets and Filter Banks*. Cambridge, MA, USA: Wellesley-Cambridge, 1996.
[27] A. Cohen, I. Daubechies, and J.-C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 45, no. 5, pp. 485–560, 1992.
[28] F. Harary, D. Hsu, and Z. Miller, "The biparticity of a graph," *J. Graph Theory*, vol. 1, no. 2, pp. 131–133, 1977.

Yuichi Tanaka (S'06–M'07) received the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 2003, 2005 and 2007, respectively. He was a Postdoctoral Scholar at Keio University, Yokohama, Japan, from 2007 to 2008, and supported by the Japan Society for the Promotion of Science (JSPS). From 2006 to 2008, he was also a visiting scholar at the University of California, San Diego (Video Processing Group supervised by Prof. T. Q. Nguyen). From 2008 to 2012, he was an Assistant Professor in the Department of Information Science, Utsunomiya University, Tochigi, Japan. Since 2012, he has been an Associate Professor in Graduate School of BASE, Tokyo University of Agriculture and Technology, Tokyo, Japan. He was a recipient of the Yasujiro Niwa Outstanding Paper Award in 2010 and the TELECOM System Technology Award in 2011. Since 2013, he has been an Associate Editor of *IEICE Transactions on Fundamentals*. His current research interests are in the field of multidimensional signal processing which includes: Graph signal processing, image and video processing with computer vision techniques, distributed video coding, objective quality metric, and effective spatial-frequency transform design.

Akie Sakiyama (S'14) received the B.E. degree in computer and information sciences from Tokyo University of Agriculture and Technology, Tokyo, Japan, in 2014. She is currently a Ph.D. student with Graduate School of BASE, Tokyo University of Agriculture and Technology. Her current research interests are graph signal processing and high-dimensional data analysis.