Adaptive Multiscale Decomposition of Graph Signals

Xianwei Zheng, *Student Member, IEEE*, Yuan Yan Tang, *Life Fellow, IEEE*, Jianjia Pan, *Member, IEEE*, and Jiantao Zhou, *Member, IEEE*

***Abstract*—This paper proposes an adaptive multiscale decomposition algorithm for graph signals. We develop two types of graph signal cost functions: α-sparsity functional and graph signal entropies, to capture the energy compaction of the signal components. The adaptive decomposition can then be constructed by applying a minimum cost constraint during the full subband decomposition. The proposed adaptive decomposition is shown to outperform graph wavelet decomposition in compressing nonpiecewise constant graph signals.**

***Index Terms*—α-sparsity, entropy, graph Fourier transform, graph signal cost function, graph wavelet decomposition (GWD).**

I. Introduction

GRAPHS are very flexible for representing many types of modern real-world data which naturally reside on irregular domains, such as sensor networks and social networks [1], [2]. In these occasions, data sources are represented as vertices on a graph, and the edges are assigned weights for measuring relationships of the vertices. Extending classical signal processing techniques for handling the graph signals have recently received a great deal of attention [2]–[10].

Discrete wavelet transform (DWT) is one of the most successful techniques in classical signal processing for time-frequency analysis [12]. To establish a framework similar to the DWT, graph wavelet filter banks (GWFBs), and graph downsampling need to be carefully designed to produce multiscale decomposition for the graph signals. Recently, GWFBs were constructed in [6]–[8] by applying spectral graph theory [11]. In addition, three types of graph signal downsampling methods were proposed: coloring-based [6], singular value decomposition (SVD)-based [4], and maximum spanning tree (MST)-based [10] downsampling. The coloring-based downsampling method colors the vertices of a graph to generate a sequence of bipartite subgraphs, and then applies downsampling according to the bipartiteness. However, the coloring-based downsampling does not generate downsampled subgraphs, making it impossible to generate a graph signal multiscale decomposition. The SVD-based downsampling method subtly generates subgraphs by polarizing the eigenvector associated with the largest eigenvalue of the graph Laplacian. Unfortunately, the SVD-based downsampling method does not necessarily generate connected bipartite subgraphs, and hence, the existing GWFBs cannot be directly applied. In contrast, MST-based downsampling ensures that the vertices in every downsampled subset can be reconnected as a graph. Such desirable property makes the existing GWFBs directly applicable in each scale to construct a multiscale graph signal decomposition. As shown in [10], a graph wavelet decomposition (GWD) can be constructed by applying MST-based downsampling and GWFBs. It was also demonstrated that GWD performs well in compressing piecewise constant graph signals, because it produces a decomposition with superior energy compaction for this type of signal. However, for nonpiecewise constant graph signals, GWD may not generate decomposition with good energy compaction, due to the fact that only low-pass components are decomposed.

In this paper, we address the problem of adaptively decomposing graph signals in such a way that energy compaction is well preserved. To this end, we develop two types of graph signal cost functions: α-sparsity functional and graph signal entropies, to capture the energy compaction of the signal components. We then integrate the graph signal cost function into the full subband decomposition constructed by GWFBs and MST-based downsampling. Specifically, we construct the adaptive decomposition of graph signals by deleting some components in the full subband decomposition under a minimum cost constraint. Our results show that the proposed adaptive decomposition algorithms outperform GWD in compressing some nonpiecewise constant graph signals.

The rest of this paper is organized as follows. Section II reviews the related work. In Section III, we present definitions of graph signal cost functions and the adaptive decomposition algorithm. Section IV provides experiment results and Section V concludes this paper.

Notations: Consider a weighted undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, where $\mathcal{V}$ and $\mathcal{E}$ denote the sets of vertices and edges, respectively. $\mathbf{W} = (w_{i,j})$ is the weighted adjacency matrix, where $w_{i,j}$ represents the weight assigned to the edge $e(i,j) \in \mathcal{V}$. A graph $\mathcal{G}$ is called unweighted if $w_{i,j} = 1$ for $e(i,j) \in \mathcal{V}$. A bipartite graph is a graph whose vertices can be partitioned into two disjoint subsets $\mathcal{V}_1$ and $\mathcal{V}_2$, such that each edge of $\mathcal{G}$ connects one vertex in $\mathcal{V}_1$ to one in $\mathcal{V}_2$. The degree matrix $\mathbf{D}$ associated with $\mathcal{G}$ is a diagonal matrix whose ith diagonal element equals to the degree of vertex i, i.e., $D_{ii} = d_i = \sum_j w_{ij}$. The normalized Laplacian matrix of $\mathcal{G}$ is defined as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$. If $\mathcal{G}$ is a unweighted cycle graph, the classical discrete Fourier transform matrix is an eigenmatrix of $\mathbf{L}$. In this case, the graph signal processing is equivalent to the classical signal processing.

Manuscript received April 5, 2016; revised June 4, 2016; accepted July 27, 2016. Date of publication August 10, 2016; date of current version August 24, 2016. This work was supported in part by the Macau Science and Technology Development Fund under Grants FDCT/009/2013/A1, FDCT/046/2014/A1, and FDCT/008/2014/AMJ and in part by the Research Committee at the University of Macau under Grants MYRG2014-00031-FST and MYRG2015-00056-FST. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Antonio García Marques. *(Corresponding author: Dr. Jiantao Zhou.)*

The authors are with the Department of Computer and Information Science, University of Macau, Taipa 999078, China (e-mail: yb47430@umac.mo; yytang@umac.mo; jianjiapan@gmail.com; jtzhou@umac.mo).

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

Digital Object Identifier 10.1109/LSP.2016.2598750

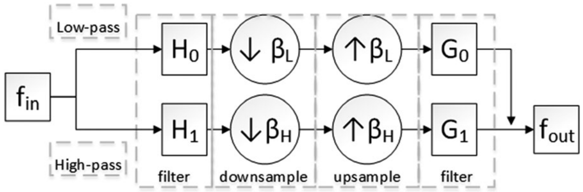


Fig. 1. Two-channel analysis and synthesis wavelet filter banks on bipartite graphs.

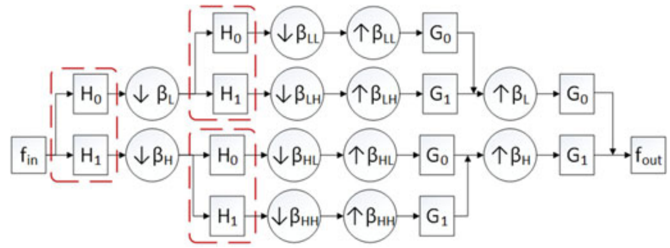


Fig. 2. Two-channel two-scale full subband wavelet filter bank.

II. Related Work

In this section, we give a brief introduction to the GWFBs and MST-based downsampling, which are the two most important ingredients of GWD.

A. GWFBs on Bipartite Graphs

In [7], Narang and Ortega proposed to construct two-channel biorthogonal compact support GWFBs with zero response (zeroDC) and nonzero response (nonzeroDC) for constant signals on bipartite graphs. These GFWBs can be represented by GraphBior(k_0, k_1), where k_0 and k_1 denote the number of roots at 0 of the low-pass filters. When $k_0 = k_1 = k$, these filters are simply written as GraphBior(k). Such a two-channel graph filter bank contains four filters: the low-pass and high-pass analysis filters $\mathbf{H}_0, \mathbf{H}_1$, and the low-pass and high-pass synthesis filters $\mathbf{G}_0, \mathbf{G}_1$, as shown in Fig. 1. These filters work in a two-channel graph filter bank to map an input graph signal $\mathbf{f}_{\text{in}}$ to an output signal $\mathbf{f}_{\text{out}}$. Here, the downsampling operators $\downarrow \beta_L$ and $\downarrow \beta_H$, respectively, keep the signal samples at low-pass and high-pass vertices, defined by the two independent sets of the underlying bipartite graph. Similarly, $\uparrow \beta_L$ and $\uparrow \beta_H$ are the upsampling operators for the low-pass and high-pass channels.

B. MST-Based Downsampling

In order to apply the GraphBior GWFBs to establish a graph signal multiscale decomposition, Nguyen and Do proposed a downsampling method based on MSTs of the graphs [10]. A spanning tree of graph $\mathcal{G}$ is a tree that contains all the vertices and a subset of edges of $\mathcal{G}$. A spanning tree is called a MST of $\mathcal{G}$ if its sum of weights is maximum among all the spanning trees of $\mathcal{G}$. Suppose that $\mathcal{T}_0 = (\mathcal{V}_0, \mathcal{E}_0, \mathbf{W}_{\mathcal{T}_0})$ is a MST of graph $\mathcal{G} = (\mathcal{V}_0, \mathcal{E}, \mathbf{W}_{\mathcal{G}})$. Define tree distance $d_{\mathcal{T}_0}(i,j)$ as the number of edges of the shortest path in $\mathcal{T}_0$ connecting vertices i and j. Let $r \in \mathcal{V}_0$ be the root vertex of $\mathcal{T}_0$. Downsampling is implemented by dividing $\mathcal{V}_0$ into $\mathcal{V}_1$ and $\mathcal{V}_0 \backslash \mathcal{V}_1$, where $\mathcal{V}_1$ includes all the vertices with even tree distance from r, and $\backslash$ is the set subtraction operator. Namely, $\mathcal{V}_1 := \{i \in \mathcal{V}_0 : d_{\mathcal{T}_0}(i,r) \text{ is even}\}$. The vertex set $\mathcal{V}_1$ is assigned edges by connecting a vertex $i \neq r$ to its grandparent vertex with the weight given by

$$w_{\mathcal{T}_1}(i, g_0(i)) = \frac{2}{\frac{1}{w_{\mathcal{T}_0}(i,p_0(i))} + \frac{1}{w_{\mathcal{T}_0}(p_0(i),g_0(i))}} \tag{1}$$

where $p_0(i)$ and $g_0(i)$ are the parent and grandparent vertices of i in $\mathcal{T}_0$. Repeating the vertex set bipartition and edge weights assignments, we can produce a tree multiresolution $\mathcal{T}_0 \supset \mathcal{T}_1 \supset \cdots \supset \mathcal{T}_{L-1}$, with edge weights

$$w_{\mathcal{T}_{l+1}}(i, g_l(i)) = \frac{2}{\frac{1}{w_{\mathcal{T}_l}(i,p_l(i))} + \frac{1}{w_{\mathcal{T}_l}(p_l(i),g_l(i))}} \tag{2}$$

for $i \in \mathcal{V}_{l+1}$, where $p_l(i)$ and $g_l(i)$ are the parent and grandparent vertices of i in $\mathcal{T}_l$.

III. Adaptive Decomposition of Graph Signals

Since DWT in classical signal processing only decomposes the low-pass components in each scale [13], it often does not lead to optimal decomposition in terms of energy compaction. For the same reason, GWD works poorly when applying to compress many oscillatory graph signals. Motivated by the wavelet packet theory [14] and [15], in this paper, we construct an adaptive multiscale decomposition algorithm based on GWD for graph signals by taking the energy compaction into account. Specifically, our contribution is to introduce the definition of cost functions for graph signals to characterize the energy compaction of the signal components. Incorporating these cost functions into the full subband decomposition framework, we design an adaptive decomposition scheme for graph signals, potentially achieving better energy compaction performance. Fig. 2 gives the two-scale full subband decomposition, where the red-dashed rectangles indicate the constraints designed through the cost functions for controlling the decomposition. Here, $\mathbf{H}_0, \mathbf{H}_1, \mathbf{G}_0, \mathbf{G}_1$, the downsampling, and the upsampling operators can be similarly defined as in Fig. 1. The graph signal cost functions and the detailed adaptive decomposition algorithm will be discussed below.

A. Graph Signal Cost Functions

We first discuss the graph signal cost functions to characterize the energy compaction, which is important to decide whether or not a signal component should be decomposed in the full subband decomposition.

Definition III.1. A map $\mathcal{M}$ from a graph signal $\mathbf{f} = \{f_i\}$ on graph $\mathcal{G}$ to $\mathbb{R}$ is called an additive cost function if $\mathcal{M}(\mathbf{0}, \mathcal{G}) = 0$ and $\mathcal{M}(\mathbf{f}, \mathcal{G}) = \sum_i \mathcal{M}(f_i, \mathcal{G})$.

Specifically, we define two classes of cost functions: α-sparsity functional and graph signal entropies.

Definition III.2. For a graph signal $\mathbf{f}$ on graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, $\alpha > 0$, functional $\mathcal{M}_\alpha(\mathbf{f}, \mathcal{G})$ defined by

$$\mathcal{M}_\alpha(\mathbf{f}, \mathcal{G}) := \#\{i : |f(i)| > \alpha, i \in \mathcal{V}\} \tag{3}$$

is called the α-sparsity of $\mathbf{f}$, where $\#\{\cdot\}$ denotes the cardinality of a given set.

Essentially, the α-sparsity counts the number of nonnegligible elements, reflecting the energy compaction property.

In addition to α-sparsity functional, we also propose to use Shannon entropy as a cost function, noticing the fact that entropy relates to signal compressibility and hence energy compaction. For a classical one-dimensional signal $\mathbf{x}$, its Shannon entropy is defined as [15]: $H(\mathbf{x}) = -\sum_i p(x_i)\log p(x_i)$, where $\log(\cdot)$ is logarithm with base 2, and $p(x_i)$ is the probability of $x = x_i$. Here, we define $0 \cdot \log 0 = 0$. Similarly, the joint Shannon entropy [16] of two random variables X and Y is defined by $H(X,Y) = -\sum_{i,j} p(x_i, y_j)\log p(x_i, y_j)$, where $p(x,y)$ is the joint probability distribution of X and Y.

To extend the definition of entropy to the graph signal setting, we need to design the graph signal entropy in such a way that both the signal and the graph it resides on are appropriately characterized. One strategy to this end is to exploit the joint entropy. Specifically, for a graph signal $(\mathbf{f}, \mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W}))$, we define graph signal entropy with joint and separable forms as follows.

a) *Graph signal Shannon entropy*:

$$H(\mathbf{f}, \mathcal{G}) = -\sum_{i,j} \frac{d_j}{2} p(f_i, d_j)\log p(f_i, d_j) \tag{4}$$

where $p(f_i, d_j)$ is the probability that sample value f_i and vertex degree d_j occur at the same vertex.

b) *Graph signal separable Shannon entropy*:

$$H_S(\mathbf{f}, \mathcal{G}) = H(\mathbf{f})\Big(-\sum_j \frac{d_j}{2} p(d_j)\log p(d_j) + 1\Big) \tag{5}$$

where $H(\mathbf{f})$ is the classical Shannon entropy of $\mathbf{f}$, $p(d_j)$ is the probability of vertex degree $d = d_j$.

The term $\frac{d_j}{2}$ in (4) and (5) is introduced to ensure that the graph signal entropies reduce to classical entropies, if the underlying graph is the cycle graph, i.e., $d_j = 2$ for all j. In addition, as can be seen from (5), signal with high (low) entropy $H(\mathbf{f})$ also leads to high (low) graph entropy.

B. Adaptive Decomposition Algorithm

We now construct the adaptive decomposition algorithm for graph signals. For a given graph signal $\mathbf{f} \in \mathbb{R}^N$ defined on graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, let $\mathcal{T}$ be the MST of $\mathcal{G}$. By using the downsampling and graph reduction technique presented in Section II-B, we can decompose $\mathcal{V}$ into low-pass and high-pass vertex sets $\mathcal{I}_l^k$, where l is the level index, and $1 \leq k \leq 2^l$. For each vertex set $\mathcal{I}_l^k$, we apply the MST-based downsampling method to produce the low-pass vertex set $\mathcal{I}_{l+1}^{2k-1}$ and the high-pass vertex set $\mathcal{I}_{l+1}^{2k}$. We then employ two-channel GraphBior GWFBs on the signal components defined on $\mathcal{I}_l^k$: $f(\mathcal{I}_l^k)$, to generate the low- and the high-pass signal children components $f(\mathcal{I}_{l+1}^{2k-1})$ and $f(\mathcal{I}_{l+1}^{2k})$. If $f(\mathcal{I}_l^k)$ satisfies the following minimum cost constraint:

$$\mathcal{M}(f(\mathcal{I}_l^k), \mathcal{G}) < \mathcal{M}(f(\mathcal{I}_{l+1}^{2k-1}), \mathcal{G}) + \mathcal{M}(f(\mathcal{I}_{l+1}^{2k}), \mathcal{G}) \tag{6}$$

then its children components are deleted. Otherwise, the corresponding children components are kept. Here, the cost function $\mathcal{M}(\cdot, \cdot)$ can be either the α-sparsity functional or the graph signal Shannon entropy defined in Section III-A. In other words, this constraint controls whether or not $f(\mathcal{I}_l^k)$ should be further decomposed. The rationale behind such selections is as follows. If the value of the cost function associated with a parent component becomes even bigger after being decomposed into two children, then there is no need to perform such decomposition, from the perspective of energy compaction. This could also be treated as a greedy algorithm to approximate the optimal decomposition under the criterion of the specified cost function. Such process is repeated until the required decomposition levels are achieved. The details for the adaptive decomposition are presented in Algorithm 1.

Algorithm 1: Adaptive graph signal decomposition algorithm

Input: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$; Graph signal $\mathbf{f}$; Number of decomposition level L.

```
Initialization: l ← 1, generate the MST of G: T.
Set I_0^1 = T.
for l = 0, 1, 2, ..., L do
    for k = 1, 2, ..., 2^l do
        — compute cost value for signal component on
        each existing vertex set I_l^k: M(f(I_l^k), G).
        — decompose I_l^k into two children components,
        I_{l+1}^{2k-1}, I_{l+1}^{2k}, and compute the cost for each component
        M(f(I_{l+1}^{2k-1}), G), M(f(I_{l+1}^{2k}), G):
        if M(f(I_l^k), G) < M(f(I_{l+1}^{2k-1}), G) + M
        (f(I_{l+1}^{2k}), G)
            — retain the parent component and eliminate the
            children components.
        else
            — retain both parent and children components.
        end if
    end for
end for
```

Output: The adaptive decomposition of $\mathbf{f}$ and $\mathcal{G}$.

IV. Experiments

In this section, we present the performance of the proposed adaptive decomposition algorithm, and compare with GWD proposed in [10]. The experiments are conducted in MATLAB R2010b with Graph_Downsampling_Toolbox [17], which includes MatlabGBL [18] and GraphBior-Filterbanks [19] toolboxes. The GWFBs used in our experiments are the GraphBior(3) zeroDC GWFBs. Note that the downsampling method in both our proposed method and GWD are the MST-based method.

We first apply the α-sparsity functional as the cost function in Algorithm 1. The resulting α-sparsity-based adaptive decomposition is called SAD in the sequel. In Fig. 3, we show an example of the 2-scale SAD for a signal on Minnesota road graph. As can be seen, SAD only decomposes the high-pass component in the second scale, rather than the low-pass component. This is quite different to GWD, in which only the low-pass components are decomposed.

We then apply the proposed adaptive decomposition algorithm to graph signal compression. We here adopt the n-term nonlinear approximation, in which the original signal is

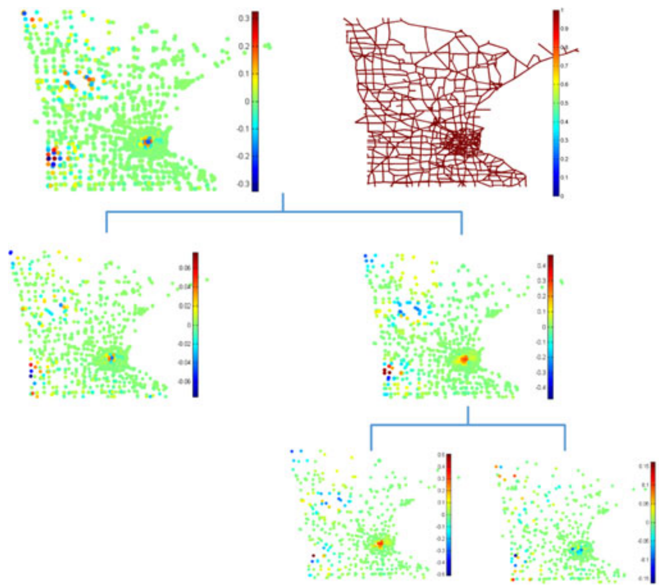
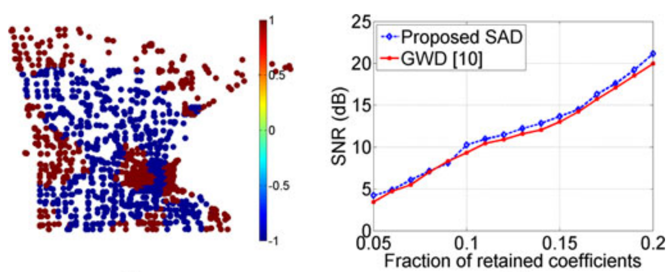


Fig. 3. Signal on Minnesota road graph and its two-scale SAD with $\alpha = 0.0077$. The signal and the graph structure are shown in the first row.

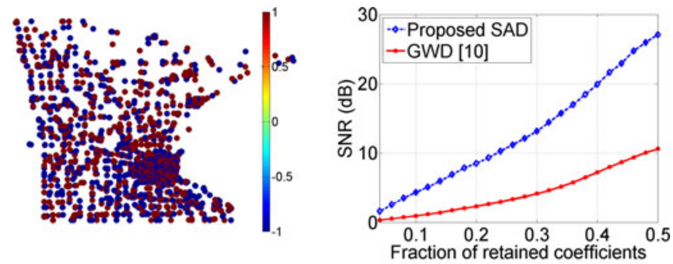


(a) (b)

Fig. 4. Reconstructions of a piecewise-constant signal by 6-level SAD and GWD. (a) Original signal. (b) Reconstruction performance of SAD and GWD.

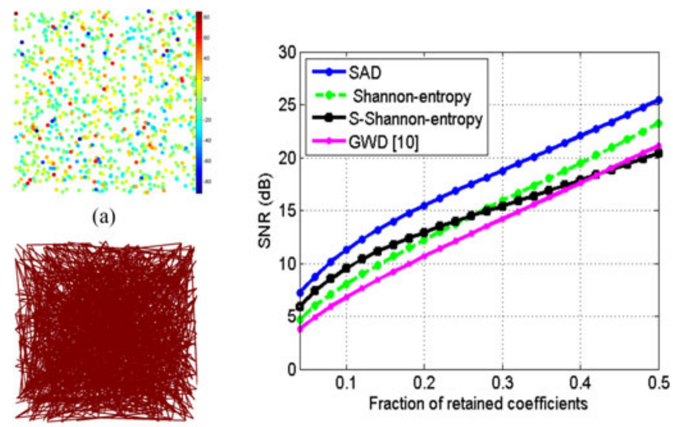
reconstructed by its n largest decomposed coefficients. For the proposed SAD algorithm, we empirically set $\alpha = \beta \cdot \overline{|\mathbf{f}|}$, where $\overline{|\mathbf{f}|} = \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} |f(i)|$ is the average absolute value of the signal, and $\beta \in (0, 1)$ is a scaling parameter. In our experiments, we select $\beta = 0.5$. We now compare the reconstruction performance of SAD and GWD in [10] for 6-level decomposition. As shown in Fig. 4, for the piecewise constant graph signals on unweighted Minnesota road graph, the performance of both methods is very close. This is because the energy of the piecewise constant signal is mostly concentrated in the low-pass components, which leads to a SAD similar to GWD. However, for nonpiecewise constant binary graph signals on unweighted Minnesota road graph, as illustrated in Fig. 5, SAD produces better decomposition in terms of energy compaction, compared with GWD. For instance, when 30% of the coefficients are available, the SNR value of SAD is 12.56 dB, which is significantly better than 4.13 dB given by the GWD. In addition, the reconstruction performance gain of our proposed method over the GWD becomes larger with the increasing number of coefficients.

We also consider the graph signals defined on unweighted Erdös–Rényi random graphs. The edges of random graphs are first generated according to a Bernoulli distribution, with a



(a) (b)

Fig. 5. Reconstructions of a nonpiecewise constant binary signal by 6-level SAD and GWD. (a) Original signal. (b) Reconstruction performance of SAD and GWD.



(a) (b) (c)

Fig. 6. Reconstruction of signals defined on random graphs. (a) Example signal. (b) Example random graph. (c) Reconstructions performance of 6-level SAD, entropy-based adaptive decomposition, and GWD.

parameter $p \in [0, 1]$ controlling the sparsity of the graph. Here, we set $p = 0.001$ and the number of vertices $N = 1500$. An example of such graph signal is demonstrated in Fig. 6(a) and (b). We then apply the proposed adaptive decomposition algorithm using α-sparsity, graph signal Shannon-entropy, and separable Shannon (S-Shannon) entropy as the cost functions to compress this type of graph signal. Fig. 6(c) presents the reconstruction performance of different decomposition methods. Each data point in this figure is the result averaging over 100 random graphs. As can be observed, SAD leads to the best performance among all the methods. Furthermore, both SAD and entropy-based adaptive decomposition methods outperform GWD with respect to reconstruction performance.

V. Conclusion

In this paper, we propose an adaptive decomposition algorithm for graph signals by considering energy compaction. We introduce the definitions of graph signal cost functions, which enable us to construct adaptive decomposition algorithms by applying the minimum cost constraint in the full subband decomposition. Specifically, we design two types of cost functions, α-sparsity functional and graph signal entropies. Experiment results show that our proposed adaptive decomposition algorithm outperforms the GWD for several representative types of graph signals.

REFERENCES

[1] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80–90, Sep. 2014.

[2] 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, May 2013.

[3] D. I. Shuman, B. Ricaud, and P. Vandergheynst, "A windowed graph Fourier transform," in *Proc. IEEE Stat. Signal Process. Workshop*, Ann Arbor, MI, USA, 2012, pp. 133–136.

[4] 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, Apr. 2016.

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

[6] 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, Jun. 2012.

[7] 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, Oct. 2013.

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

[9] D. B. H. Tay and J. Zhang, "Techniques for constructing biorthogonal bipartite graph filter banks," *IEEE Trans. Signal Process.*, vol. 63, no. 21, pp. 5772–5783, Nov. 2015.

[10] H. Q. Nguyen and M. N. Do, "Downsampling of signals on graphs via maximum spanning trees," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 182–191, Jan. 2015.

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

[12] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd ed. New York, NY, USA: Academic, 2008.

[13] F. G. Meyer, A. Z. Averbuch, and J. O. Strömberg, "Fast adaptive wavelet packet image compression," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 792–800, May 2000.

[14] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 713–718, Mar. 1992.

[15] M. V. Wickerhauser, *Adapted Wavelet Analysis: From Theory to Software*. Natick, MA, USA: A.K. Peters, 1994.

[16] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[17] H. Q. Nguyen, Graph_Downsampling_Toolbox, Jul. 2015. [Online]. Available: http://bigwww.epfl.ch/nguyen/publications/Graph%20Downsampling%20Toolbox.zip

[18] D. Gleich, MatlabBGL, Oct. 2008. [Online]. Available: https://www.cs.purdue.edu/homes/dgleich/packages/matlab_bgl

[19] S. K. Narang, GraphBior-Filterbanks, Nov. 2012. [Online]. Available: http://biron.usc.edu/kumarsun/Codes/graphBior-Filterbanks