

# BIPARTITE SUBGRAPH DECOMPOSITION FOR CRITICALLY SAMPLED WAVELET FILTERBANKS ON ARBITRARY GRAPHS

Jin Zeng<sup>†</sup>, Gene Cheung<sup>\*</sup>, and Antonio Ortega<sup>‡</sup>

<sup>†</sup> The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong

<sup>\*</sup> National Institute of Informatics, Chiyoda-ku, Tokyo, Japan

<sup>‡</sup> University of Southern California, Los Angeles, CA, USA

## ABSTRACT

The observation of frequency folding in graph spectrum during down-sampling for signals on bipartite graphs—analogueous to the same phenomenon in Fourier domain for regularly sampled signals—has led to the development of critically sampled wavelet filterbanks such as GraphBior. However, typical graph-signals live on general graphs that are not necessarily bipartite. To decompose a non-bipartite graph into a series of bipartite subgraphs so that two-channel filterbanks can be applied iteratively, we propose a new algorithm based on two criteria easily computed in the vertex domain aiming at compact signal representation in the wavelet domain. Given that filterbanks have minimal frequency discrimination at 1, the first criterion aims to minimize the multiplicity of mid graph frequency 1. The second criterion aims to preserve the edge structure of the original graph, which may reflect correlations among signal samples, so that a signal projected on approximated bipartite subgraphs can nonetheless be well represented using low frequency components. Experimental results show that our proposed bipartite subgraph decomposition outperforms competing proposals in terms of energy compaction.

**Index Terms**— graph signal processing, bipartite subgraph decomposition, graph wavelet filterbanks

## 1. INTRODUCTION

One of the key problems in graph signal processing (GSP) [1] is the design of critically sampled wavelets for graph-signals. Unlike regularly sampled signals on 1D time line (*e.g.*, audio) or on 2D grid (*e.g.*, images), graph-signals typically live on structured graphs such as social networks or wireless sensor networks, which reflect relationships among nodes. Designing appropriate wavelet filters to account for all these correlations is a difficult task. Towards this goal, one recent contribution is GraphBior in [2], which are critically sampled two-channel wavelet filterbanks, resulting in compact representation of bipartite graph-signals. However, graph-signals typically live on general graphs that are not necessarily bipartite.

In order to apply two-channel filterbanks on non-bipartite graphs, [3] first decomposed the input graph into a series of edge-disjoint bipartite subgraphs, and then applied filtering and down-sampling separately on each subgraph. In [3], the decomposition was accomplished using Harary's algorithm [4]: for a  $c$ -colorable graph, at each step the vertices are separated into two independent sets according to the  $i$ -th bit of its color index, where  $i = 1, 2, \dots, \lceil \log_2 c \rceil$ . Harary's algorithm is adopted in many graph wavelet implementations [2, 5]. Since the method performs graph coloring first, the performance will depend largely on the coloring algorithm. However, graph coloring is an NP-hard problem to solve optimally [6] and NP-hard to approximate within  $n^{1-\epsilon}$  [7].

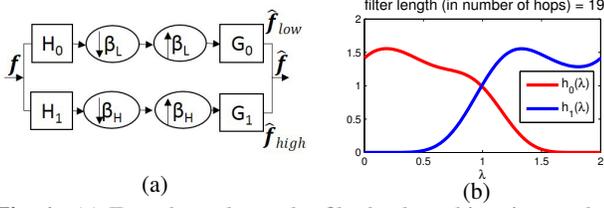
[8] proposed a decomposition called *min-cut weighted max-cut* (MCWMC), claiming that a desired criterion of the decomposition is to maximally separate the neighborhoods in each bipartite subgraph. The authors also argued that the subgraphs should maximally retain edges in the original graph to avoid isolated vertices that cannot be filtered during wavelet transform. This approach assigns a weight to each edge according to its influence in changing the neighborhood structure, and then performs max-cut at each iteration. This work discussed the relationship between graph decomposition and the subsequent filtering operations, but did not analyze how its proposed properties can affect the compactness of wavelet domain signal representation directly.

The state-of-the-art method is the *maximum spanning tree* (MST) based down-sampling [9], which approximates the original graph with a MST at each iteration in order to efficiently achieve max-cut in the resulting subgraphs. Even though the method outputs a bipartite subgraph decomposition, it actually aims at reconstructing the original signal from one set of vertices, so even if max-cut is justified to minimize the linear interpolation error, it is not proved to benefit signal decomposition.

In summary, a common problem among existing methods is the lack of criteria directly related to compact signal representation to guide bipartite subgraph decomposition. In contrast, in this paper we develop two criteria that can be easily computed in the vertex domain to guide a bipartite subgraph decomposition that leads to compact signal representation in the graph wavelet domain. The first criterion minimizes the multiplicity of mid graph frequency 1, where a high-dimension subspace would lead to non-compact signal representation at both low-pass and high-pass channels due to lack of frequency discrimination at this cutoff frequency. The second criterion preserves the edge structure of the original graph, which may reflect correlations among signal samples, so that signal projected on approximated bipartite subgraphs can nonetheless be well represented using low frequency components. Experimental results show that our proposed bipartite subgraph decomposition outperforms competing proposals in terms of energy compaction.

## 2. CRITICALLY SAMPLED WAVELET FILTERBANKS

Consider an undirected, weighted graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  composed of a vertex set  $\mathcal{V}$  of size  $N$  and an edge set  $\mathcal{E}$  specified by  $(i, j, w_{i,j})$ , where  $i, j \in \mathcal{V}$ ,  $w_{i,j} \in \mathbb{R}^+$  is the edge weight between vertices  $i$  and  $j$ . Thus a weighted graph can be characterized by its adjacency matrix  $\mathbf{W}$  with  $\mathbf{W}(i, j) = w_{i,j}$ .  $\mathbf{D}$  denotes the diagonal degree matrix where entry  $d_{i,i} = \sum_j w_{i,j}$ . Laplacian matrix is defined as  $\mathbf{L} = \mathbf{D} - \mathbf{W}$ , and the normalized form is given by  $\mathcal{L} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2}$  [10]. A graph signal is a mapping that assigns a value to each vertex, denoted as  $\mathbf{f} = [f_1, \dots, f_N]^T$ .



**Fig. 1:** (a) Two-channel wavelet filterbank on bipartite graph; (b) Kernels of  $\mathbf{H}_0, \mathbf{H}_1$  in graphBior [2] with filter length of 19.

$\mathcal{L}$  has real eigenvalues  $\{\lambda_i\}_{i=1,2,\dots,N}$  within range  $[0, 2]$  and corresponding eigenvectors  $\{\mathbf{u}_{\lambda_i}\}_{i=1,2,\dots,N}$ . The eigenvalues can be interpreted as graph frequencies, and they form the spectrum of the graph  $\mathcal{G}$  [10], denoted as  $\sigma(\mathcal{L}) = \{\lambda_1, \dots, \lambda_N\}$ . Then a spectral domain filter  $\mathbf{H}$  for graph signal can be defined in terms of the spectral kernel  $h(\lambda)$ .

A bipartite graph  $\mathcal{B} = (L, H, \mathcal{E})$  is a graph whose vertices can be divided into two disjoint sets  $L$  and  $H$ , such that each edge connects a vertex in  $L$  to one in  $H$ . A graph is called  $c$ -colorable if each vertex in a graph  $\mathcal{G}$  can be assigned one of  $c$  colors, such that no edge connects two vertices of the same color.

A two-channel wavelet filterbank on a bipartite graph decomposes a signal on the graph into low-pass and high-pass components [3]. As shown in Fig. 1(a),  $\mathbf{H}_i, \mathbf{G}_i$  are the respective analysis and synthesis wavelet filters. Down-sampling functions  $\beta_L(n), \beta_H(n)$  keep only the signal samples in vertex set  $L$  and  $H$  respectively, and subsequently the down-sampled signal is up-sampled to original size by inserting zero at discarded vertices. Thus the overall output is critically sampled, *i.e.*,  $|L| + |H| = N$ .

The “down-sample then up-sample” (DU) operation in the bipartite graph is shown in [11] to produce a graph-frequency spectral folding phenomenon in DU signal, analogous to folding of Fourier frequencies on regularly sampled signal. GraphBior filterbanks [2] exploit this observation to satisfy perfect reconstruction by designing spectral filters  $\mathbf{H}_i$  and  $\mathbf{G}_i$  satisfying spectral-folding cancellation and perfect reconstruction conditions. Note that  $\lambda = 1$  is the point where the kernels,  $h_i(\lambda)$  and  $g_i(\lambda)$ , of  $\mathbf{H}_i, \mathbf{G}_i$  have the same response as illustrated in Fig. 1(b) [2]. Thus at the mid-frequency there is minimal energy discrimination, with equal amount of energy going to low- and high-pass. This lack of discrimination can be found in regular filterbanks.

The above mentioned filterbanks are applicable only for bipartite graphs. So given a general graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , [3] proposed to first decompose it into  $k$  edge-disjoint bipartite subgraphs  $\mathcal{B}_i = (L_i, H_i, \mathcal{E}_i)$ , defined on the same vertex set  $L_i \cup H_i = \mathcal{V}$ , and each edge in  $\mathcal{G}$  belongs to only one bipartite subgraph  $\mathcal{B}_i$ , *i.e.*,  $\mathcal{E}_i \cap \mathcal{E}_j = \emptyset$ ,  $i \neq j$ ,  $\cup_i \mathcal{E}_i = \mathcal{E}$ . Then the filterbank is implemented separably in  $k$  steps, restricting the operation to only one subgraph at each step, similar to separable filter for images where filters in  $x$ - and  $y$ -dimension are applied separately.

### 3. BIPARTITE SUBGRAPH DECOMPOSITION

We seek a bipartite subgraph decomposition  $\mathcal{G}'$  of the original non-bipartite graph  $\mathcal{G}$  that achieves compact representation of signals in  $\mathcal{G}$  projected to wavelet domain of  $\mathcal{G}'$ . To accomplish this, we have two criteria. First, we seek  $\mathcal{G}'$  with minimal mid-frequency multiplicity so as to minimize the dimension of subspace at  $\lambda = 1$  where the filterbanks have least energy discrimination. Second, assuming that the edges  $\mathcal{E}$  in  $\mathcal{G}$  capture the intrinsic relations among vertices—leading to low-frequency representation of signals in graph frequen-

cies of  $\mathcal{G}$  [12]—we seek to maximally preserve edge structure in the original graph during bipartite approximation. We will next discuss how to obtain a bipartition satisfying these two criteria.

#### 3.1. Bipartition with Minimum Mid-frequency Multiplicity

For purpose of compact representation in wavelet domain,  $\mathcal{G}'$  should have minimal multiplicity of eigenvalue  $\lambda = 1$ . Specifically, when  $\lambda = 1$  has high multiplicity, leading to a high-dimension subspace corresponding to this frequency, a large signal energy projected at  $\lambda = 1$  due to the high-dimension subspace will be extracted to both low- and high-pass channels, jeopardizing the goal of compact representation. However, if one fails to take this into account, the resulting bipartite graph can have very high multiplicity at  $\lambda = 1$ , as observed in real-world cases in Table 1. Hence it is critical to reduce the multiplicity of  $\lambda = 1$ .

Given our approximated graph structure  $\mathcal{G}'$  is bipartite with partition  $(\mathcal{V}_1, \mathcal{V}_2)$ , we can rearrange the vertices in  $\mathcal{V}_1$  to have smaller indices than those in  $\mathcal{V}_2$ , so that  $\mathbf{W}$  and  $\mathbf{D}$  can be written as:

$$\mathbf{W} = \begin{bmatrix} \mathbf{0} & \mathbf{W}_{1,2} \\ \mathbf{W}_{1,2}^T & \mathbf{0} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{bmatrix}.$$

The normalized Laplacian  $\mathcal{L}$  can then be written as:

$$\begin{aligned} \mathcal{L} &= \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{W} \mathbf{D}^{-1/2} \\ &= \mathbf{I} - \begin{bmatrix} \mathbf{0} & \mathbf{D}_1^{-1/2} \mathbf{W}_{1,2} \mathbf{D}_2^{-1/2} \\ \mathbf{D}_2^{-1/2} \mathbf{W}_{1,2}^T \mathbf{D}_1^{-1/2} & \mathbf{0} \end{bmatrix}. \end{aligned} \quad (1)$$

Eigenvector  $\mathbf{v} = [\mathbf{v}_1; \mathbf{v}_2]$  for  $\lambda = 1$  satisfies  $(\mathcal{L} - \mathbf{I})\mathbf{v} = \mathbf{0}$ :

$$\begin{bmatrix} \mathbf{0} & \mathbf{D}_1^{-1/2} \mathbf{W}_{1,2} \mathbf{D}_2^{-1/2} \\ \mathbf{D}_2^{-1/2} \mathbf{W}_{1,2}^T \mathbf{D}_1^{-1/2} & \mathbf{0} \end{bmatrix} \mathbf{v} = \mathbf{0}$$

Expanding the above, we get:

$$\begin{cases} \mathbf{W}_{1,2} \mathbf{v}'_2 = \mathbf{0} \\ \mathbf{W}_{1,2}^T \mathbf{v}'_1 = \mathbf{0} \end{cases},$$

where  $\mathbf{v}'_1 = \mathbf{D}_1^{-1/2} \mathbf{v}_1$  and  $\mathbf{v}'_2 = \mathbf{D}_2^{-1/2} \mathbf{v}_2$ . Hence the multiplicity of  $\lambda = 1$  is the sum of nullity of  $\mathbf{W}_{1,2}$  and  $\mathbf{W}_{1,2}^T$ , *i.e.*  $2 \text{null}(\mathbf{W}_{1,2})$ . So to reduce the multiplicity of  $\lambda = 1$  for more compact spectral decomposition, we minimize  $\text{null}(\mathbf{W}_{1,2})$ , or maximize  $\text{rank}(\mathbf{W}_{1,2})$ . Obviously, a balanced bipartition is preferred, since  $\text{rank}(\mathbf{W}_{1,2}) \leq \min\{|\mathcal{V}_1|, |\mathcal{V}_2|\}$ . Hence if an algorithm does not balance the sizes of the two partitions, the multiplicity of  $\lambda = 1$  will be large as exemplified in Table 1.

**Table 1:** Multiplicity of  $\lambda = 1$  for first level bipartite subgraph using Harary’s decomposition algorithm.

Graph	Number of vertices	Multiplicity
Minnesota Traffic Graph	2642	428
Yale Coat of Arms [13]	1059	103
China Temperature Graph (Fig 3(a))	208	32

#### 3.2. Bipartition with Structure Preservation

The bipartite subgraph should well approximate the original graph, because the original graph has edges consistent with the inter-node

correlation structure of typical signals on the graph, leading to signal smoothness with respect to the graph, and energy compaction of signals in low graph frequencies only [12]. Both [8] and [9] propose to approximate original graph through bipartition with max-cut, but do not show how max-cut relates directly to compact signal representation in wavelet domain.

Intuitively, edges in densely connected areas have less influence on the overall graph structure than those in sparsely connected areas; removing an edge in a sparse area can lead to disconnected local patches or even isolated vertices, weakening the effectiveness of filtering. [8] shows similar concern about edge importance and experimentally validates this intuition, but its goal is to optimize neighborhood separation in subgraphs and not to improve compact representation. In the following, we will show how the *Kullback-Leibler divergence* (KLD) [14] agrees with this intuition, and thus is a reasonable metric that promotes compact representation.

We model the signal  $\mathbf{f}$  on the original graph  $\mathcal{G}$  as a Gaussian random field (GRF), following the multivariate normal distribution, denoted as  $\mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  where  $\boldsymbol{\mu}$  is the mean vector, and  $\boldsymbol{\Sigma}$  is the covariance matrix. The inverse covariance matrix (precision matrix) is specified by the graph, written as  $\boldsymbol{\Sigma}^{-1} = \mathbf{L} + \delta\mathbf{I}$  where  $1/\delta$  is interpreted as the variance of DC component of  $\mathbf{f}$  [15]. The bipartition thus will generate a new graph specifying a new signal distribution, so for compact representation, the distance between the two distributions measured by KLD should be minimized.

To examine how the importance of an edge is affected by the connection density, or in other words, the degrees of its connecting nodes, we assume that the graph is unweighted, *i.e.*,  $w_{i,j} = 1, i, j \in \mathcal{V}$ . We remove the edge connecting vertices  $i$  and  $j$ , then the new graph  $\mathcal{R}$ , with Laplacian matrix  $\mathbf{L}_R$ , specifies a new signal distribution  $\mathcal{N}_R(\boldsymbol{\mu}_R, \boldsymbol{\Sigma}_R)$  where  $\boldsymbol{\mu}_R$  is assumed to equal  $\boldsymbol{\mu}$ , and  $\boldsymbol{\Sigma}_R^{-1} = \mathbf{L}_R + \delta\mathbf{I}$ . The difference between  $\mathcal{N}$  and  $\mathcal{N}_R$  is measured by KLD:

$$\begin{aligned} D_{KL}(\mathcal{N}||\mathcal{N}_R) &= \frac{1}{2} \left( \text{tr}(\boldsymbol{\Sigma}_R^{-1}\boldsymbol{\Sigma}) + (\boldsymbol{\mu}_R - \boldsymbol{\mu})^T \boldsymbol{\Sigma}_R^{-1}(\boldsymbol{\mu}_R - \boldsymbol{\mu}) \right) \\ &\quad - N + \ln \left( \frac{|\boldsymbol{\Sigma}_R|}{|\boldsymbol{\Sigma}|} \right) \\ &= \frac{1}{2} \left( \text{tr}(\boldsymbol{\Sigma}_R^{-1}\boldsymbol{\Sigma}) - \text{tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}) - \ln |\boldsymbol{\Sigma}_R^{-1}\boldsymbol{\Sigma}| \right) \\ &= \frac{1}{2} \left( \underbrace{\text{tr}((\mathbf{L}_R - \mathbf{L})\boldsymbol{\Sigma})}_A - \underbrace{\ln |(\mathbf{L}_R + \delta\mathbf{I})\boldsymbol{\Sigma}|}_B \right). \end{aligned} \quad (2)$$

For illustration, we label the two vertices linked by the removed edge as 1 and 2 with respective degrees  $d_1$  and  $d_2$ , and denote the edge as  $e_{1,2}$ . In  $\boldsymbol{\Sigma}$ , denote the variance of  $f_i$  as  $\sigma_i$ , and covariance between  $f_i$  and  $f_j$  as  $\sigma_{i,j}$ . In  $R$ , the degrees of the two vertices are reduced to  $d_1 - 1$  and  $d_2 - 1$ . So the first term in (3) is rewritten as:

$$A = \text{tr}((\mathbf{L}_R - \mathbf{L})\boldsymbol{\Sigma}) = -(\sigma_1 + \sigma_2 - 2\sigma_{1,2}). \quad (4)$$

Similarly, the second term in (3) is rewritten as:

$$\begin{aligned} B &= \ln \det((\mathbf{L}_R + \delta\mathbf{I})\boldsymbol{\Sigma}) = \ln(b_{11} * b_{22} - b_{12} * b_{21}) \\ &= \ln(1 - (\sigma_1 + \sigma_2 - 2\sigma_{1,2})), \end{aligned} \quad (5)$$

where (5) follows from the fact that  $(\mathbf{L}_R + \delta\mathbf{I})\boldsymbol{\Sigma}$  differs from  $(\mathbf{L} + \delta\mathbf{I})\boldsymbol{\Sigma}$  only in the  $(i, j)$ th element,  $i, j \in \{1, 2\}$ , so that only the first and second rows of matrix  $(\mathbf{L}_R + \delta\mathbf{I})\boldsymbol{\Sigma}$  differ from  $(\mathbf{L} + \delta\mathbf{I})\boldsymbol{\Sigma} = \mathbf{I}$ .

Hence its matrix determinant depends only on the upper-left four elements  $b_{ij}, i, j \in \{1, 2\}$ . (3) can now be written as:

$$\begin{aligned} D_{KL}(\mathcal{N}||\mathcal{N}_R) &= \\ &= \frac{1}{2} \left( -(\sigma_1 + \sigma_2 - 2\sigma_{1,2}) - \ln(1 - (\sigma_1 + \sigma_2 - 2\sigma_{1,2})) \right). \end{aligned} \quad (7)$$

Note that the first row of  $(\mathbf{L} + \delta\mathbf{I})$  multiplied by the first column of  $\boldsymbol{\Sigma}$  equals 1, so we have,

$$(d_1 + \delta)\sigma_1 - \sum_{i \in \mathcal{N}_1} \sigma_{1,i} = 1, \quad (8)$$

where  $\mathcal{N}_1$  is the set of neighboring vertices for vertex 1. For clearer illustration, assume that the covariances between the current vertex and its neighbors are approximately same, *i.e.*  $\sigma_{i,m} \approx \sigma_{i,n}, m, n \in \mathcal{N}_i$  which is the normal case when the neighboring vertices are of similar degrees. Then (8) becomes,

$$(d_1 + \delta)\sigma_1 - d_1\sigma_{1,2} \approx 1. \quad (9)$$

By definition  $(\mathbf{L} + \delta\mathbf{I})\boldsymbol{\Sigma} = \mathbf{I}$ , whose first column is  $(\mathbf{L} + \delta\mathbf{I})[\sigma_1, \sigma_{1,2}, \dots, \sigma_{1,N}]^T = [1, 0, \dots, 0]^T$ , so by summing up the array elements, we have  $\delta\mathbf{1}^T[\sigma_1, \sigma_{1,2}, \dots, \sigma_{1,N}]^T = 1$ , *i.e.*,  $\delta(\sigma_1 + \sum_{i=2}^N \sigma_{1,i}) = 1$ . Assume  $\sigma_1 + \sum_{i \in \mathcal{N}_1} \sigma_{1,i} \gg \sum_{i \notin \{1, \mathcal{N}_1\}} \sigma_{1,i}$ , we have

$$\delta(\sigma_1 + \sum_{i \in \mathcal{N}_1} \sigma_{1,i}) \approx 1. \quad (10)$$

(9) minus (10) gives

$$\sigma_1 \approx (1 + \delta)\sigma_{1,2}. \quad (11)$$

Similarly  $\sigma_2 \approx (1 + \delta)\sigma_{1,2}$ , so we have  $\sigma_1 - \sigma_{1,2} = \frac{\delta}{1+\delta}\sigma_1$  and  $\sigma_2 - \sigma_{1,2} = \frac{\delta}{1+\delta}\sigma_2$ , with which (7) can be written as,

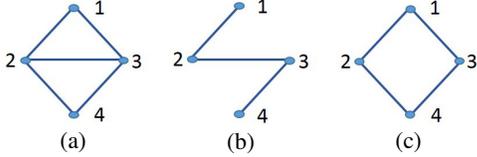
$$\begin{aligned} D_{KL}(\mathcal{N}||\mathcal{N}_R) &\approx \\ &= \frac{1}{2} \left( -\frac{\delta}{1+\delta}(\sigma_1 + \sigma_2) - \ln\left(1 - \frac{\delta}{1+\delta}(\sigma_1 + \sigma_2)\right) \right). \end{aligned} \quad (12)$$

With (11), (9) becomes  $(d_1 \frac{\delta}{1+\delta} + \delta)\sigma_1 \approx 1$ . When vertex 1 has many connections, *i.e.*,  $d_1$  is large, then  $\sigma_1$  will be small. Then if both vertices 1 and 2 are of relatively high degrees, then  $\sigma_1$  and  $\sigma_2$  will be small, resulting in a smaller  $D_{KL}$  as can be seen from (12). We see now that KLD indeed agrees with the intuition that an edge at a densely connected area is less important than one in a sparsely connected area. So given the Laplacian matrices of the two graphs, the difference can be measured using KLD defined in (2), and to preserve the structure of original graph,  $D_{KL}$  need to be minimized.

### 3.3. Bipartite Subgraph Decomposition

While maximizing  $\text{rank}(\mathbf{W}_{1,2})$  and minimizing  $D_{KL}(\mathcal{N}||\mathcal{N}_B)$  both promote compact signal representation, optimizing one does not necessarily imply optimizing the other one. An example to illustrate this is the bipartition of the four-vertex graph in Fig. 2(a). To achieve the maximum  $\text{rank}(\mathbf{W}_{1,2}) = 2$ , vertices 1 and 3 are grouped in the same set as shown in Fig. 2(b), thus edges  $e_{1,3}$  and  $e_{2,4}$  are removed, resulting in a relatively large  $D_{KL}$ . However, to minimize  $D_{KL}$ , vertices 1 and 2 should be grouped together as shown in Fig. 2(c), so that only edge  $e_{2,3}$  need to be removed, but resulting in a low  $\text{rank}(\mathbf{W}_{1,2}) = 1$ . Hence both of the two criteria need to be incorporated into a well-designed bipartition algorithm.

In particular, we design a heuristic algorithm for bipartite subgraph decomposition. For a given graph  $\mathcal{G}$ , we build the bipartite



**Fig. 2:** Trade-off between two criteria: (a) original graph; (b) bipartition favoring rank; (c) bipartition favoring  $D_{KL}$ .

graph  $\mathcal{B}_1$  by adding the vertices one by one into two sets and removing the edges within each set. Specifically, for each connected component, we start from one random vertex and put it in set 1, and use the breadth-first search (BFS) [16] to explore other vertices<sup>1</sup>. To decide which set a discovered vertex should be allocated,  $\text{rank}(\mathbf{W}_{1,2})$  and  $D_{KL}$  are calculated assuming the vertex is allocated to set 1 or 2. We then choose the option with a higher  $\text{rank}(\mathbf{W}_{1,2})$ . If  $\text{rank}(\mathbf{W}_{1,2})$  are the same, the one with a smaller  $D_{KL}$  will be chosen. Note that the bipartite graph grows gradually as more vertices are discovered, and the calculation of  $\text{rank}(\mathbf{W}_{1,2})$  and  $D_{KL}$  is based on discovered vertices only. In addition, the two criteria are combined in this way to avoid any weighting parameter, making the algorithm simpler.

After the first-level bipartite graph  $\mathcal{B}_1$  is obtained, its edges  $\mathcal{E}_1$  are removed from  $\mathcal{G}$ , which gives the updated  $\mathcal{G}_1 = (\mathcal{V}, \mathcal{E} - \mathcal{E}_1)$  for bipartition in the next level. Then  $\mathcal{G}$  is iteratively decomposed into edge-disjoint bipartite subgraphs  $\mathcal{B}_1, \dots, \mathcal{B}_k$  up to  $k$  levels, where  $k$  is decided by the chromatic number  $c$ , i.e.  $k = \lceil \log_2 c \rceil$ . The proposed bipartite subgraph decomposition is named *Bipartite Subgraph Decomposition Optimizing Mid-frequency and Structure* (MFS), and summed up in Algorithm 1.

**Algorithm 1** Bipartite Subgraph Decomposition Optimizing Mid-frequency and Structure

**Input:** graph  $\mathcal{G}$ , decomposition level  $k$

**Output:** edge-disjoint bipartite graphs  $\mathcal{B}_1, \dots, \mathcal{B}_k$

- 1: **for**  $i = 1:k$  **do**
- 2: Find connected components in  $\mathcal{G}$ .
- 3: For each component, put the starting vertex in set 1.
- 4: Use breadth-first search to explore other vertices, and choose the proper set by jointly comparing  $\text{rank}(\mathbf{W}_{1,2})$  and  $D_{KL}$ .
- 5: After all vertices are discovered, bipartite graph  $\mathcal{B}_i$  is given.
- 6: Update  $\mathcal{G}$  by removing edges in  $\mathcal{B}_i$ .
- 7: **end for**

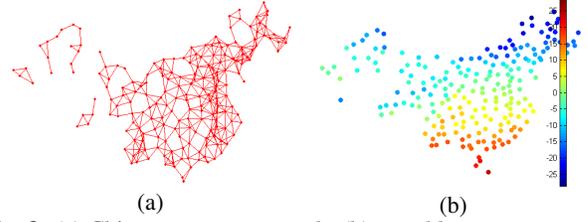
#### 4. EXPERIMENTAL RESULTS

We now compare our proposed MFS algorithm with existing schemes: Harary's Decomposition [3], MCWMC algorithm [8], and MST-based Construction of Bipartite Graph Multiresolution (MST) [9]. China temperature graph [17] is used for testing, where each vertex represents a weather station. Given the longitude and latitude vectors,  $\mathbf{x}$  and  $\mathbf{y}$ , for all the vertices  $\mathcal{V}$ , the graph is constructed by connecting the vertices whose distance is under the threshold:

$$T = \sqrt{\frac{(\mathbf{x}_{\max} - \mathbf{x}_{\min})(\mathbf{y}_{\max} - \mathbf{y}_{\min})}{N}}, \quad (13)$$

where  $N$  is the vertex number. This results in a 6-colorable graph, so a 3-level decomposition is needed. The signals are the monthly

<sup>1</sup>Alternatively using an exhaustive search increases complexity significantly without bringing noticeable improvement in performance in our experiments, thus BFS is adopted.



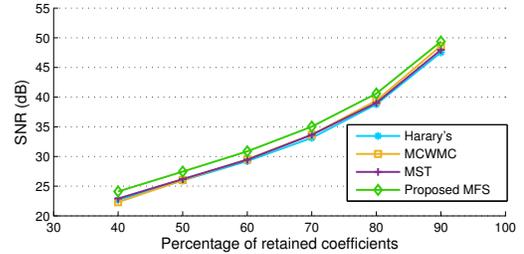
**Fig. 3:** (a) China temperature graph; (b) monthly average temperature in Jan. 2010.

average temperature from Oct. 2009 to May. 2012; 32 datasets in total. Fig. 3 shows this unweighted graph and its signal in Jan. 2010.

First, the graph is decomposed into three bipartite subgraphs by each of the four algorithms.  $\text{rank}(\mathbf{W}_{1,2})$  and  $D_{KL}$  of the results are shown in Table 2. Note that  $D_{KL}$  is between the subgraph and the corresponding updated original graph. It is clear that MFS outperforms the three existing methods in terms of producing the highest  $\text{rank}(\mathbf{W}_{1,2})$  and the smallest  $D_{KL}$  in average.

**Table 2:**  $\text{rank}(\mathbf{W}_{1,2})$  and  $D_{KL}$  of each bipartition and average.

		Harary's	MCWMC	MST	MFS
<b>rank</b>	Subgraph 1	88	92	77	100
	Subgraph 2	73	63	36	87
	Subgraph 3	20	9	13	38
	<b>Average</b>	60.3	54.7	42	75
<b><math>D_{KL}</math></b>	Subgraph 1	6.46	6.21	11.79	6.66
	Subgraph 2	2.24	2.96	2.06	1.76
	Subgraph 3	0	4.06	0.27	0
	<b>Average</b>	2.90	4.41	4.71	2.81



**Fig. 4:** Non-linear approximation curve.

After bipartition, we apply zeroDC graphBior [2] on the signal, then we reconstruct the signal using the largest  $n\%$  wavelet coefficients, and the average SNR of 32 datasets are plotted in Fig. 4. In addition, we compute average SNR gains of MFS over the three other methods (calculated using G. Bjontegaard's metric [18]). We also induce different graphs by connecting vertices differently: i) change the connecting threshold to  $0.8T, 1.2T, 1.4T$ , ii) connect each node to its  $k$  nearest neighbors, where  $k = 7, 8, 9$ . The average SNR gains for different resulting graphs are shown in Table 3. It is clear that MFS outperforms existing schemes in all these different graphs as well.

**Table 3:** Average gain of MFS over competing schemes in SNR(dB) for graphs with different connections.

	T	0.8T	1.2T	1.4T	k=7	k=8	k=9
<b>Harary's</b>	1.65	1.43	0.82	0.82	0.76	0.64	1.34
<b>MCWMC</b>	1.35	0.74	1.17	1.24	1.56	1.62	2.06
<b>MST</b>	1.35	0.16	2.24	1.38	0.93	0.64	1.91

## 5. 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," in *IEEE Signal Processing Magazine*, May 2013, vol. 30, no.3, pp. 83–98.
- [2] S.K. Narang and A. Ortega, "Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs," *Signal Processing, IEEE Transactions on*, vol. 61, no. 19, pp. 4673–4685, Oct 2013.
- [3] S.K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *Signal Processing, IEEE Transactions on*, vol. 60, no. 6, pp. 2786–2799, June 2012.
- [4] F. Harary, D. Hsu, and Z. Miller, "The biparticity of a graph," *Journal of Graph Theory*, vol. 1, no. 2, pp. 131–133, 1977.
- [5] Y. Tanaka and A. Sakiyama, "M-channel oversampled graph filter banks," *Signal Processing, IEEE Transactions on*, vol. 62, no. 14, pp. 3578–3590, July 2014.
- [6] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, pp. 85–103. Springer US, 1972.
- [7] D. Zuckerman, "Linear degree extractors and the inapproximability of max clique and chromatic number," in *Proceedings of the Thirty-eighth Annual ACM Symposium on Theory of Computing*, 2006, STOC '06, pp. 681–690.
- [8] S.K. Narang and A. Ortega, "Multi-dimensional separable critically sampled wavelet filterbanks on arbitrary graphs," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, March 2012, pp. 3501–3504.
- [9] H.Q. Nguyen and M.N. Do, "Downsampling of signals on graphs via maximum spanning trees," *Signal Processing, IEEE Transactions on*, vol. 63, no. 1, pp. 182–191, Jan 2015.
- [10] F. RK Chung, *Spectral graph theory*, American Mathematical Soc., 1997.
- [11] S.K. Narang and A. Ortega, "Downsampling graphs using spectral theory," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, May 2011, pp. 4208–4211.
- [12] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Laplacian matrix learning for smooth graph signal representation," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, 2015, pp. 3736–3740.
- [13] D. Spielman, "Yale coat of arms graph," <http://www.cs.yale.edu/homes/spielman/561/>, 2015-09-23. Retrieved: 2015-09-24.
- [14] J. Duchi, "Derivations for linear algebra and optimization," 2014.
- [15] A. Gadde and A. Ortega, "A probabilistic interpretation of sampling theory of graph signals," *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pp. 3257–3261, 2015.
- [16] M. Bóna, *A Walk Through Combinatorics: An Introduction to Enumeration and Graph Theory*, World Scientific, 2011.
- [17] The National Centers for Environmental Information (NCEI), "China temperature data," <http://www.ncdc.noaa.gov/cdo-web/>, 2015-02-01. Retrieved: 2015-04-30.
- [18] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," *Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA*, April 2001.