RECOVERY OF GRAPH SIGNALS FROM SIGN MEASUREMENTS

Wenwei Liu[1], *Hui Feng*[1,2], *Kaixuan Wang*[1], *Feng Ji*[3], *Bo Hu*[1,2]

[1] School of Information Science and Technology, Fudan University, Shanghai 200433, China
[2] Shanghai Institute of Intelligent Electronics & Systems, Shanghai 200433, China
[3] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
{wwliu21,kaixuanwang21}@m.fudan.edu.cn, {hfeng, bohu}@fudan.edu.cn, jifeng@ntu.edu.sg

ABSTRACT

Sampling and interpolation of continuous graph signals have been extensively studied, in order to reconstruct or estimate the entire graph signal from the signal values on a subset of vertices. Whereas in a lot of real-world scenarios, only the signs of signals are available. For example, a rating system may only provide simple options such as "like" or "dislike". We are interested in whether it is possible to recover the original signal from such coarse information. In this paper, the reconstruction of bandlimited graph signals based on sign measurements is discussed and a greedy sampling strategy is proposed. The simulation experiments are presented, and the greedy sampling algorithm is compared with the random sampling algorithm, which verifies the feasibility of the proposed approach.

Index Terms— Graph signal processing, sign information, sampling and recovery

1. INTRODUCTION

In general, many types of data in life can be modeled as graph signals, such as transportation networks, social networks, and sensor networks [1–3]. As one of the emerging analyzing tools in the signal processing community in recent years [4,5], graph signal processing (GSP) has a wide range of application scenarios in semantic segmentation, behavior recognition, community discovery, traffic prediction and other aspects [6]. Based on the graph Fourier Transform (GFT), techniques analogous to those in the classical Fourier theory have been developed in the GSP framework [7]. Common graph signals in life often show bandlimited characteristics, i.e., signal values associated with neighboring vertices are similar [8]. For instance, in social networks, people who are closely connected tend to share similar preferences.

It is a fundamental problem of GSP to recover signals from partial samples [9, 10]. At present, there are two main sampling schemes, deterministic and random [9, 11]. The deterministic approach attempts to find a sampling set that minimizes a preset loss function. The random approach usually calculates the sampling probability distribution of each vertex according to its importance. In practice, random selection is of low computational complexity, but may need more samples than a deterministic approach to achieve equivalent sampling effect for subsequent reconstruction.

In many cases, it may be difficult to obtain continuous values of a graph signal. For instance, in a goods rating system, customers may feel difficult to give specific scores to a product range from 0 to 100, but it is easier to provide simple evaluations for goods such as "like","dislike","indifference". It will be valuable to dig fine-grained scores from the coarse information gathered, which will help the merchants to make more precise recommendations.

The problem raised here is related to the recovery from 1-bit quantized signals, which can date back to early works on reconstruction of continuous signals from zero-crossing information [12]. Leveraged by the compressed sensing theory, a binary iterative hard thresholding (BIHT) algorithm has been proposed accordingly to recover the spectral sparse signal in discrete domain [13]. However, previous works do not consider the impact of the graph topology in recovering process. In specific, different choice of vertices to be sampled will result in different recovery precision.

In this paper, we investigate the problem of recovering bandlimited graph signals from the signs of limited samples. We provide a reconstruction algorithm based on sign information, whose convergence performance is analyzed theoretically. Furthermore, we propose an effective sampling scheme and expound its implementation details. Finally, we present simulation experiments to test the performance of the proposed algorithm, whose results verify the validity of our work.

2. MODEL

A connected undirected graph $\mathcal{G}$ with N vertices can be represented as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \boldsymbol{W}\}$, where $\mathcal{V}$ is the set of vertices, $\mathcal{E}$ is the set of edges, and $\boldsymbol{W}$ is the weighted adjacency matrix

This work was supported by CURE (Hui-Chun Chin and Tsung-Dao Lee Chinese Undergraduate Research Endowment) (19930), and National University Student Innovation Program (201910246154).

with $W_{ij} > 0$ if the edge $(i, j) \in \mathcal{E}$, otherwise $W_{ij} = 0$. The degree of vertex i is $d_i = \sum_j W_{ij}$, and $\boldsymbol{D}$ is the diagonal degree matrix whose ith diagonal element is d_i. The graph Laplacian $\boldsymbol{L}$ is defined as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, with eigendecomposition $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, where $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$ and $\boldsymbol{U} = [\boldsymbol{u}_1, ..., \boldsymbol{u}_N]$ is an orthonormal matrix containing the corresponding eigenvectors.

A graph signal $\boldsymbol{x} : \mathcal{V} \rightarrow \mathbb{R}$ is a vector defined on $\mathcal{G}$. The GFT of $\boldsymbol{x}$ is defined as $\hat{\boldsymbol{x}} = \boldsymbol{U}^T\boldsymbol{x}$. Each component of $\hat{\boldsymbol{x}}$ is the corresponding frequency coefficient. Let f_L, f_U be positive integers, and $f_L \leq f_U \leq N$. A graph signal is called bandlimited if $\hat{\boldsymbol{x}}_k = 0$ for $k \notin [f_L, f_U]$, with bandwidth $B = f_U - f_L + 1$. Such signals exist in the constraint space

$$\mathcal{C}_b = \left\{ \boldsymbol{x} \in \mathbb{R}^N \,|\, \forall i \notin [f_L, f_U], \boldsymbol{u}_i^T \boldsymbol{x} = 0 \right\}, \tag{1}$$

where $\mathcal{C}_b$ is a closed convex cone in $\mathbb{R}^N$.

Consider obtaining M samples on a subset $\mathcal{V}'$ of $\mathcal{V}$, i.e. $|\mathcal{V}'| = M$, the sampling process can be described as

$$\boldsymbol{y} = \boldsymbol{\Psi}_v \boldsymbol{x},$$

where $\boldsymbol{\Psi}_v \in \mathbb{R}^{M \times N}$ is the vertex-domain sampling matrix, with elements $(\boldsymbol{\Psi}_v)_{ij} = 1$ if the ith sample is vertex j, and 0 otherwise.

The sign information at a vertex comes from a sign operation of the signal at the corresponding vertex. For example, the sign on vertex j is sign(x_j), where

$$\text{sign}(x) = \begin{cases} -1 & x < 0 \\ 1 & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The sampling process of sign information on vertices can be described as

$$\boldsymbol{s}_x = \text{sign}\left(\boldsymbol{\Psi}_v \boldsymbol{x}\right). \tag{3}$$

As the sign of a signal does not depend on the magnitude of the signal itself, we assume $\|\boldsymbol{x}\| = 1$[1].

For a signal with sign $\boldsymbol{s}_x$ on vertices, from (3) we can derive that the signal should be confined to a constraint space

$$\mathcal{C}_s = \bigcap_{i=1}^{M} \left\{ \boldsymbol{x} \in \mathbb{R}^N \,|\, \boldsymbol{a}_i^T \boldsymbol{x} < 0 (> 0, \, = 0) \right\}. \tag{4}$$

In (4), $\boldsymbol{a}_i$ is nearly an all zero vector with its jth component being 1. When $(\boldsymbol{s}_x)_i$ is -1, 1, 0 respectively, the relational operator in (4) is $<$, $>$, $=$ correspondingly.

[1] In practice, the magnitude of the signal may lie in a prior range such as the daily temperature. In application such as rating system, we are more interested in the ordinal of the signals on vertices, not the actual magnitude.

3. RECONSTRUCTION ALGORITHM

Through sign information in the sampling set, the original bandlimited graph signal lies in space: $\mathcal{C}_b \cap \mathcal{C}_s$ Here, we relax the $\mathcal{C}_s$ to its closed convex hull for the sake of subsequent recovery as follows

$$\mathcal{C}_v = \bigcap_{i=1}^{M} \left\{ \boldsymbol{x} \in \mathbb{R}^N \,|\, \boldsymbol{a}_i^T \boldsymbol{x} \leq 0 (\geq 0, = 0) \right\}. \tag{5}$$

It is not hard to prove that $\mathcal{C}_v$ is a closed convex cone in $\mathbb{R}^N$. The feasible region $\mathcal{C}_b \cap \mathcal{C}_v$, denoted as $\mathcal{C}$, is also a closed convex cone. Thus, we can define the projection operators on $\mathcal{C}_b$ and $\mathcal{C}_v$ [14].

Projecting any signal onto $\mathcal{C}_b$ requires only one band-pass filtering operation

$$\boldsymbol{P}_b = \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{U}^T, \tag{6}$$

where $\boldsymbol{\Gamma}$ is a diagonal matrix whose diagonal elements are 1s inside the passband and 0s outside the passband.

The projection operation onto $\mathcal{C}_v$ can be defined as

$$\left(\boldsymbol{P}_v \boldsymbol{x}\right)_j := \begin{cases} 0 & j \in \mathcal{V}', \text{sign}\left(x_j\right) \neq \left(\boldsymbol{s}_x\right)_i \\ x_j & \text{otherwise} \end{cases} \tag{7}$$

According to (6) and (7), the signal can be reconstructed by continuously projecting onto these convex sets (POCS) [15]. Specifically, for any initial signal $\boldsymbol{x}_0$, the applied algorithm is a simple iterative process

$$\boldsymbol{x}_{n+1} = \boldsymbol{P}_b \boldsymbol{P}_v \boldsymbol{x}_n. \tag{8}$$

We prove that (8) satisfies the following convergence properties.

Theorem 1. *The iterative sequence $\{\boldsymbol{x}_n\}$ converges to some point $\boldsymbol{x}^*$ in $\mathcal{C}$, and the convergence rate is independent of the selection of the initial point $\boldsymbol{x}_0$.*

Proof. See Appendix in [16] for details. □

4. DESIGN OF SAMPLING SET

Since the recovery sequence of the graph signal will converge onto the feasible region, reducing the feasible region can improve the recovery performance. From (5), the feasible region of the signal depends on the sampling results, we aim to design the sampling set carefully given the sampling budget, to obtain a small feasible region.

4.1. Feasible Region Analysis

From sections 2 and 3, we see that the feasible region is a finite-dimensional closed convex cone. If we want to make it smaller, it is necessary to determine a metric for its size.

Denote the graph Fourier basis in the passband by $\boldsymbol{U}_B = [\boldsymbol{u}_{f_L}, ..., \boldsymbol{u}_{f_U}]$. Since the columns of $\boldsymbol{U}_B$ are orthonormal, for any two signals $\boldsymbol{\beta}_1$, $\boldsymbol{\beta}_2$ in $\mathcal{C}$ with coordinates $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$, we have $\langle \boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \rangle = \langle \boldsymbol{U}_B \boldsymbol{\alpha}_1, \boldsymbol{U}_B \boldsymbol{\alpha}_2 \rangle = \langle \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2 \rangle$. In other words, denoting the feasible region of coordinates as $\hat{\mathcal{C}}$, we can transform the exploration in $\mathcal{C}$ into the exploration in $\hat{\mathcal{C}}$, because it is of one-to-one correspondence between vectors in $\mathcal{C}$ and $\hat{\mathcal{C}}$. In view of (1) and (5), $\hat{\mathcal{C}}$ can be written as

$$\hat{\mathcal{C}} = \bigcap_{i=1}^{M} \left\{ \boldsymbol{\alpha} \in \mathbb{R}^B \,|\, \boldsymbol{a}_i^T \boldsymbol{U}_B \boldsymbol{\alpha} \leq 0 (\geq 0, = 0) \right\}. \tag{9}$$

At the same time, as a closed convex cone, $\hat{\mathcal{C}}$ can also be written as

$$\hat{\mathcal{C}} = \left\{ \sum_{i=1}^{r} \omega_i \boldsymbol{\varphi}_i \,|\, \omega_i \geq 0 \right\}.$$

The vectors $\{\boldsymbol{\varphi}_i\}_{i=1}^r$ are called extreme vectors [17]. In other words, any vector in $\hat{\mathcal{C}}$ can be represented linearly by extreme vectors with non-negative coefficients.

We want the recovered signal to be as close to the original signal as possible, i.e., the angle between them is as small as possible. Hence, we try to use the biggest angle among extreme vectors to quantify the size of $\hat{\mathcal{C}}$. Let $\mathcal{Z}$ be the set of normalized extreme vectors of $\hat{\mathcal{C}}$, then the metric for $\hat{\mathcal{C}}$ can be represented as

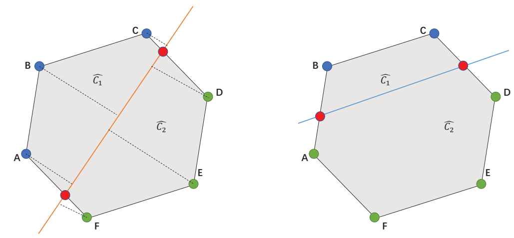
$$\theta(\hat{\mathcal{C}}) = \min_{\boldsymbol{\gamma}, \boldsymbol{\mu} \in \mathcal{Z}} \langle \boldsymbol{\gamma}, \boldsymbol{\mu} \rangle. \tag{10}$$

4.2. Greedy Sampling Algorithm

According to (9), different sampling sets correspond to different feasible regions, so an intuitive idea is to find a sampling set that makes the feasible region as small as possible.

Note that a cone in $\mathbb{R}^B$ has at least B extreme vectors, which means there is no extreme vector until the number of samples reaches $B-1$. Here, we sort the norms of the row vectors of $\boldsymbol{U}_B$ in descending order, and select the first $B-1$ linearly independent indices as the first $B-1$ samples, i.e. the initial sampling set $\mathcal{S}_0$. In this way, the first extreme vector can be calculated uniquely. This is one of the initialization schemes we use, not the only one. Once $\mathcal{S}_0$ is determined, there is only one extreme vector in current $\mathcal{Z}$.

Assume the current sampling set is $\mathcal{S} \subset \mathcal{V}$, the current feasible region is $\hat{\mathcal{C}}$, the next sample is $\xi \in \mathcal{S}^c$, where $\mathcal{S}^c = \mathcal{V} \backslash \mathcal{S}$. For every additional sample, we're adding a constraint to our current feasible region. For example, if we choose ξ as the next sample, of which the sign is negative, in that way the new feasible region becomes $\hat{\mathcal{C}} \cap \{\boldsymbol{\alpha} | \boldsymbol{U}_B(\xi, :)\boldsymbol{\alpha} \leq 0\}$, where $\boldsymbol{U}_B(\xi, :)$ is the ξth row vector of $\boldsymbol{U}_B$. The boundary of the new constraint is a hyperplane $\mathcal{H} = \{\boldsymbol{\alpha} | \boldsymbol{U}_B(\xi, :)\boldsymbol{\alpha} = 0\}$. For example, in Fig.1(a), the gray polygon area is a cross section of this cone, i.e. the feasible region, in down direction towards the origin. The solid orange line stands for the hyperplane $\mathcal{H}$. The blue and green solid dots represent the extreme vectors in the current feasible region, and the red solid dots represent the new extreme vectors associated with $\mathcal{H}$. The whole feasible region is divided into two parts $\hat{\mathcal{C}}_1$, $\hat{\mathcal{C}}_2$ by $\mathcal{H}$. Without loss of generality, assume $\hat{\mathcal{C}}_1$ is $\hat{\mathcal{C}} \cap \{\boldsymbol{\alpha} | \boldsymbol{U}_B(\xi, :)\boldsymbol{\alpha} \leq 0\}$, which means if we choose ξ as the next sample, then the new feasible region is $\hat{\mathcal{C}}_1$. On condition that the sign is positive, $\hat{\mathcal{C}}_2$ is the new feasible region instead. Each unsampled vertex corresponds to a hyperplane, our target is to find a suitable unsampled vertex whose hyperplane can reduce $\theta(\hat{\mathcal{C}})$.



(a) (b)

Fig. 1. Examples of two different samples in $\hat{\mathcal{C}}$.

Of course, we do not know any information of each unsampled vertex until we sample it. In order to determine the next sample, we have to discuss the three possible signs $\pm 1, 0$ for each vertex in $\mathcal{S}^c$, because of the inevitable uncertainties before sampling. For the three possible signs, three different feasible regions can be obtained. We suggest the following scheme: for unsampled vertex i, compute the corresponding θ for each of the new feasible regions using (10) under the three signs, denoted as $\theta_1^i, \theta_2^i, \theta_3^i$. Let $\theta^i = \min\{\theta_1^i, \theta_2^i, \theta_3^i\}$, then the next sample ξ can be chosen as follows

$$\xi = \mathrm{argmax}_j \;\; \theta^j. \tag{11}$$

However, it is of high complexity to compute $\mathcal{Z}$ for each unsampled vertex. To address this issue, we propose a greedy sampling algorithm. When selecting the Bth sample, we can apply (11), and get the corresponding $\mathcal{Z}$ according to the sampling result $\mathrm{sign}(x_\xi)$. Now, there's more than one extreme vector in $\mathcal{Z}$. In the subsequent sampling process, when calculating θ using (10) of current $\mathcal{Z}$, the extreme vectors $\boldsymbol{t}_1, \boldsymbol{t}_2$ with the biggest angle in $\mathcal{Z}$ can be recorded, for example, A and D in Fig.1. If $\boldsymbol{t}_1, \boldsymbol{t}_2$ are not separated by the hyperplane of an unsampled vertex, then we would not choose it as the next sample. Because if $\boldsymbol{t}_1, \boldsymbol{t}_2$ are still in the new feasible region, θ of the the new feasible region might be as big as the previous one. We call such sampling invalid. Refer to Fig.1, the hyperplane in (a) separates A and D, so this vertex is a valid sample. Similarly, the hyperplane in (b) is invalid.

How to evaluate all the valid samples is the following question. Here we can see all the valid hyperplanes divide the feasible region into two parts. If these two parts are divided as evenly as possible, then whatever the sign is, the new feasible region would not be too bad. In other words, if these two parts are greatly different from each other, there is a high risk that the new feasible region would be almost as large as the previous one. As a result, we want all the extreme vectors to lie roughly "equidistant" on both sides of the hyperplane under consideration. Refer to Fig.1(a), each dotted line segment stands for the distance from the corresponding extreme vector to the hyperplane. For each valid unsampled vertex i, sum up such distances as d^i

$$d^i = \sum_{\gamma \in \mathcal{Z}} \frac{\boldsymbol{U}_B(i,:)\boldsymbol{\gamma}}{\|\boldsymbol{U}_B(i,:)\boldsymbol{\gamma}\|}. \tag{12}$$

If d^i is numerically close to 0, the feasible region is roughly "cut" in half. Then we can choose the next sample:

$$\xi = \text{argmin}_j \ |d^j|. \tag{13}$$

Once the sample is determined, we can update the feasible region. This process of greedy sampling is then continued until the sampling budget is reached.

To sum up, the sampling algorithm can be described as:

Algorithm 1 Greedy Sampling Algorithm

Input: Sampling budget M, passband $[f_L, f_U]$, graph $\mathcal{G}$
Output: Sampling set $\mathcal{S}$

Get initial sampling set $\mathcal{S}_0$, $\mathcal{S} \leftarrow \mathcal{S}_0$, $s \leftarrow 0$
Calculate ξ using (11), $\mathcal{S} \leftarrow \mathcal{S} \cup \{\xi\}$
Determine optimal points $\boldsymbol{t}_1$,$\boldsymbol{t_2}$ of (10) according to acquired sign(x_ξ)
while $s < M - B$ **do**
  for $i \in \mathcal{S}^c$ and $(\boldsymbol{U}_B(i,:)\boldsymbol{t}_1)\,(\boldsymbol{U}_B(i,:)\boldsymbol{t}_2) < 0$ **do**
    Calculate d^i using (12)
  end for
  Calculate ξ using (13), $\mathcal{S} \leftarrow \mathcal{S} \cup \{\xi\}$, $s \leftarrow s + 1$
  Update extreme vectors
end while

5. SIMULATION

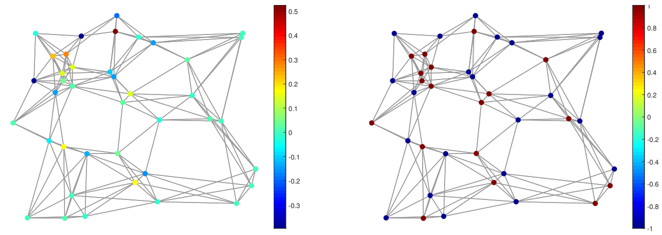
In the following experiment, we firstly apply the Algorithm 1 and random sampling to get the corresponding sampling sets, and next we choose K initial signals to recover the original signal using (8). Then we compare the reconstruction quality by the average error in angle defined as

$$\delta = \frac{1}{K} \sum_{i=1}^{K} \arccos\langle \boldsymbol{x}, \boldsymbol{x}_i^* \rangle, \tag{14}$$

where $\boldsymbol{x}_i^*$ stands for the normalized recovery signal of the ith initial signal. The larger δ is, the worse the recovery is.

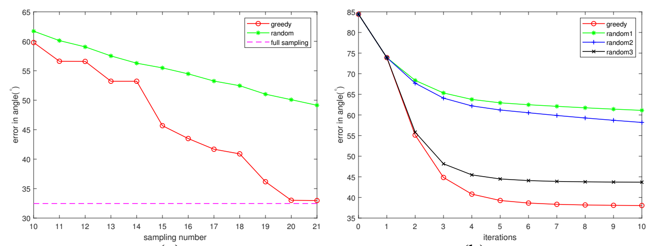
Consider a bandlimited graph signal on a sensor graph generated from gspbox in Matlab, which is shown in Fig.2 [18]. The parameters of the graph topology and the graph signal are: $N = 40$, $|\mathcal{E}| = 153$, $f_L = 29$, $f_U = 35$.

We applied our sampling algorithm and random sampling algorithm 50 times to obtain their respective sample sets, and then calculated the recovery error after 10^4 iterations with 50 initial random signals for each sampling set.



(a) (b)

Fig. 2. (a) A bandlimited graph signal of unit amplitude. (b) Sign information on vertices in (a).



(a) (b)

Fig. 3. Comparison of recovery performance between greedy sampling and random sampling (a) at different sampling numbers, and (b) during a iteration at 40% sampling rate.

We compared the average error in angle of greedy sampling and random sampling at different sampling numbers, as presented in the Fig.3(a), where the dotted pink line stands for the average error in angle when all vertices are sampled (full sampling). As we can see, with the increase of sampling numbers, the average error in angle goes down continuously. Moreover, the recovery performance of the greedy sampling is always better than the random sampling, and the error is very close to that of full sampling at 50% sampling rate.

We give the error in angle during the iterations in recovery with 16 samples in Fig.3(b), where greedy and random 1$\sim$3 stand for our greedy algorithm and three instances of random experiments with the same initial point, respectively. As expected, compared with random sampling, the error in angle of greedy sampling set descends more steeply.

6. CONCLUSION

In this paper, we put forward the idea of sign sampling and recovery of bandlimited graph signals. Based on sign information on vertices, we propose the corresponding reconstruction and greedy sampling algorithm for bandlimited graph signals. On the one hand, we guarantee the performance of the recovery algorithm by theoretical proof. On the other hand, simulation results show that greedy sampling algorithm has better performance than random sampling. In all, our work has practical significance to a certain extent and could be valuable in the field of sampling and recovery of graph signals.

7. REFERENCES

[1] Ljubisa Stankovic, Danilo P Mandic, Milos Dakovic, Ilia Kisil, Ervin Sejdic, and Anthony G Constantinides, "Understanding the basis of graph signal processing via an intuitive example-driven approach [lecture notes]," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 133–145, 2019.

[2] Dengyong Zhou and Bernhard Schölkopf, "A regularization framework for learning from graph data," in *ICML 2004 Workshop on Statistical Relational Learning and Its Connections to Other Fields (SRL 2004)*, 2004, pp. 132–137.

[3] Ziwei Zhang, Peng Cui, and Wenwu Zhu, "Deep learning on graphs: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[4] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE signal processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[5] Ljubiša Stanković, Danilo Mandic, Miloš Daković, Miloš Brajović, Bruno Scalzo, Shengxi Li, Anthony G Constantinides, et al., "Data analytics on graphs part ii: Signals on graphs," *Foundations and Trends® in Machine Learning*, vol. 13, no. 2-3, 2020.

[6] Feng Xia, Ke Sun, Shuo Yu, Abdul Aziz, Liangtian Wan, Shirui Pan, and Huan Liu, "Graph learning: A survey," *arXiv preprint arXiv:2105.00696*, 2021.

[7] Dragoš M Cvetkovic, "Applications of graph spectra: An introduction to the literature," *Appl. Graph Spectra*, vol. 13, no. 21, pp. 7–31, 2009.

[8] Aamir Anis, Akshay Gadde, and Antonio Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3775–3789, 2016.

[9] Yuichi Tanaka, Yonina C Eldar, Antonio Ortega, and Gene Cheung, "Sampling signals on graphs: From theory to applications," *IEEE Signal Processing Magazine*, vol. 37, no. 6, pp. 14–30, 2020.

[10] Benjamin Girault, Antonio Ortega, and Shrikanth S Narayayan, "Graph vertex sampling with arbitrary graph signal hilbert spaces," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 5670–5674.

[11] Mikhail Tsitsvero, Sergio Barbarossa, and Paolo Di Lorenzo, "Signals on graphs: Uncertainty principle and sampling," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4845–4860, 2016.

[12] Benjamin F Logan, "Information in the zero crossings of bandpass signals," *The Bell System Technical Journal*, vol. 56, pp. 487–510, 1977.

[13] Laurent Jacques, Jason N Laska, Petros T Boufounos, and Richard G Baraniuk, "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors," *IEEE Transactions on Information Theory*, vol. 59, no. 4, pp. 2082–2102, 2013.

[14] Sergios Theodoridis, Konstantinos Slavakis, and Isao Yamada, "Adaptive learning in a world of projections," *IEEE Signal Processing Magazine*, vol. 28, no. 1, pp. 97–123, 2010.

[15] Heinz H Bauschke and Jonathan M Borwein, "On projection algorithms for solving convex feasibility problems," *SIAM review*, vol. 38, no. 3, pp. 367–426, 1996.

[16] Wenwei Liu, Hui Feng, Kaixuan Wang, Feng Ji, and Bo Hu, "Recovery of graph signals from sign measurements," *arXiv preprint arXiv:2109.12576*, 2021.

[17] George Phillip Barker, "The lattice of faces of a finite dimensional cone," *Linear Algebra and its Applications*, vol. 7, no. 1, pp. 71–82, 1973.

[18] Nathanaël Perraudin, Johan Paratte, David Shuman, Lionel Martin, Vassilis Kalofolias, Pierre Vandergheynst, and David K Hammond, "Gspbox: A toolbox for signal processing on graphs," *arXiv preprint arXiv:1408.5781*, 2014.