ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9413561

GRAPH LEARNING UNDER SPECTRAL SPARSITY CONSTRAINTS

B. Subbareddy[*][†], *Aditya Siripuram*[*], *Jingxin Zhang*[†]

[*] Indian Institute of Technology Hyderabad, India.
[†] Swinburne University of Technology, Australia.

ABSTRACT

Graph inference plays an essential role in machine learning, pattern recognition, and classification. Signal processing based approaches in literature generally assume some variational property of the observed data on the graph. We make a case for inferring graphs on which the observed data has high variation. We propose a new signal processing based inference model and a new learning criterion that allow for wideband frequency variation in the data and derive an algorithm for graph inference. The proposed inference algorithm consists of two steps: 1) learning orthogonal eigenvectors of a graph from the data; 2) recovering the adjacency matrix of the graph topology from the given graph eigenvectors. The first step is solved by an iterative algorithm with a closed-form solution. In the second step, the adjacency matrix is inferred from the eigenvectors by solving a convex optimization problem. Numerical results on synthetic data show the proposed inference algorithm can effectively capture the meaningful graph topology from observed data under the wideband assumption.

Index Terms— Graph signal processing, Graph topology inference, Sparse reconstruction, Graph learning.

1. INTRODUCTION

Graph Signal Processing (GSP) provides a framework for processing data which is unstructured, complex, and massive. Such data is ubiquitous including the data from brain networks [1], sensor works, gene networks and transport networks [2] as examples. GSP handles such complex data by effectively capturing the underlying relationship using graphs. In most of these domains, the signal of interest is more naturally indexed by the vertices of an underlying graph. The additional information from graph topology (which possibly encodes latent domain constraints) potentially allows better signal processing techniques than the classical framework. Research over the past decade has been successful in applying conventional signal processing techniques, like rate of change or frequency properties, to graphs [3]. Among the many subfields in this research include designing graph filters, sampling, graph neural networks and graph learning from data. In the graph learning paradigm (the focus of this paper), the goal is to infer the underlying graph structure, given the data as signals (indexed by the vertices of such graph), with some assumptions on the relationship between the graph and the signal space. Most of these assumptions are based on the graph spectrum: this is indicative of how much a signal is allowed to change across an edge of the graph. Graph learning has found applications in diverse areas such as machine learning, biological network, and sensor networks [4]. The papers [5], [6] have surveyed most of the work so far on graph learning from the signal processing perspective.

Graph reconstruction from the given data requires certain assumptions on how the data is related to the graph. Global smoothness based approaches essentially assume all observed signals have low graph frequencies (i.e., vary smoothly over the edges of the graph) [7], [4], [8]. Filtering based techniques essentially assume a linear map from the unknown input to the observed signals, and hence try to force all of the observed data into the same graph frequency spectrum [9]. Diffusion based approaches assume the data is generated as a smooth diffusion process starting from a few heat sources [10] thus implicitly force a low pass filtering on the graph signal spectrum.

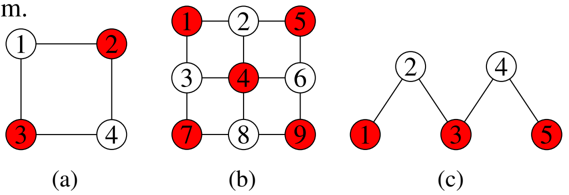


Fig. 1: Lateral inhibition system models.

However, to the best of our knowledge, these approaches do not seem to allow the data to have both low and high graph frequencies. High frequency (negative correlation/high rate of change) across graph edges also conveys a structural relationship in the data. Hence it may not be prudent to allow only smooth variation. In graph based image processing, the high frequency represents the edges and transitions in the image, and often needs to be retained and enhanced [11]. As discussed in [12, pp-881], [13], one of the reasons GSP based analysis is appealing to brain imaging techniques is its ability to capture high frequency variation. For example, in [14], [13] the graph spectrum of brain signals during visual-motor learning tasks was decomposed into the low, band and high frequency components; and it was observed that most of the information of brain signals have both low and high graph frequencies.

As an additional motivation for considering high frequencies, we consider lateral inhibition, a mechanism that occurs in neuro-biology and gene interactions. Lateral inhibition is a cell to cell or neuron to neuron signal mechanism, where the excited neuron inhibits the action of the neighboring neurons. Figure 1 illustrates the process of lateral inhibition, where the red nodes are capacity excited neurons that reduce the activity of their neighbors (white nodes) [15]. While the phenomenon of diffusion is more suitably captured by low frequencies, lateral inhibition is more suitably captured by high frequency variation on the graph. Existing methods do not seem to perform well when data has such high frequency variation (see the discussions in Sections 2 and 4).

Also, allowing the observed data to have high graph frequencies might result in a more compact and informative graph structural representation, which might be practically useful. This is the main motivation for our work in this paper.

To account for a wide range of frequencies, we propose a new sparsity based graph learning model and a new learning criterion. Based on this model and criterion, we proceed in two phases - we first provide an algorithm for estimating the eigenvectors of the graph matrix and then proceed to find the eigenvalues from the eigenvectors in the second phase [9]. In principle, our model in this paper is similar to the block sparsity models of [16], [17] where all signals are assumed to have the same frequency support and behave smoothly with respect to the underlying graph structure; however, the crucial difference is that we allow our data to have arbitrary frequency support anywhere within the entire frequency range. We refer to these as wideband graph signals.

The paper is organized as follows. In Section 2, we review graph signal processing along with notations used in the rest of the paper. In Section 2.1, we introduce the notation, problem statement, and an overview of the proposed algorithm. In Section 3 we give an overview of the proposed algorithm. Subsequently, in section 4, we discuss the results obtained using the proposed algorithm on synthetic datasets and provide a comparison with other well known graph learning algorithms. Finally, Section 5 concludes the work.

2. NOTATIONS AND PROBLEM FORMULATION

We assume the set of vertices of the underlying graph $\mathcal{G}$ is $V = \{1, 2, \ldots, N\}$, and the set of edges is E, potentially with each edge having some non-negative weight. We denote by $A_\mathcal{G}$ the adjacency matrix of the graph $\mathcal{G}$: an $N \times N$ matrix whose entries are the weights of the corresponding edges. The degree matrix $D_\mathcal{G}$ is a diagonal $N \times N$ matrix whose diagonal entries are the number of edges from a given vertex (or, for weighted graphs, the sum of weights of edges incident on a vertex). We denote the Laplacian matrix of the graph $\mathcal{G}$ as $L_\mathcal{G} = D_\mathcal{G} - A_\mathcal{G}$. The subscript $\mathcal{G}$ is omitted if the graph is apparent from the context.

A graph signal $\mathbf{x} \in \mathbb{R}^N$ is a mapping $\mathbf{x} : V \mapsto \mathbb{R}$. Similar to traditional signals, the notion of graph frequencies and graph Fourier transform of graph signals can be defined ([2], [18], [3]). For a graph signal $\mathbf{x}$ defined on a graph $\mathcal{G}$, the Graph Fourier Transform (GFT) $\mathbf{y}$ of $\mathbf{x}$ is defined as $\mathbf{y} = V_\mathcal{G}^\mathsf{T}\mathbf{x}$ where the $V_\mathcal{G}$ is the eigenvector matrix of $A_\mathcal{G}$. The quadratic form $\mathbf{x}^\mathsf{T} L_\mathcal{G}\mathbf{x}$ captures the *variation* of the signal $\mathbf{x}$ over the graph $L_\mathcal{G}$. Thus the eigenvectors of the Laplacian $L_\mathcal{G}$ have a natural rate of change interpretation: the eigenvectors corresponding to smaller eigenvalues do not vary much over the edges of the graph; and eigenvectors corresponding to larger eigenvalues have more variation. The number of nonzero coefficients in vector $\mathbf{y}$ is $\|\mathbf{y}\|_0$. In this work, we focus on adjacency-based graph transforms, though similar techniques may apply to Laplacian-based transforms as well.

2.1. Problem formulation

Given the set of observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m \in \mathbb{R}^N$ from the unknown graph, the graph learning problem is to infer the unknown underlying graph. A popular graph learning algorithm [4] tries to find a graph $\mathcal{G}$ such that the observations vary smoothly (or have low variation) on the graph:

$$\underset{L_\mathcal{G}, Z}{\arg\min} \|X - Z\|_F^2 + \alpha Tr(Z^\mathsf{T} L_\mathcal{G} Z) + \beta \|L_\mathcal{G}\|_F^2 \tag{1}$$

$$\text{s.t. } \operatorname{tr}(L_\mathcal{G}) = N, {L_\mathcal{G}}_{ij} = {L_\mathcal{G}}_{ji} \leq 0, i \neq j, L_\mathcal{G} \cdot \mathbf{1} = \mathbf{0}$$

Here X is the $N \times M$ observation matrix that has $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$ as its columns. The second in the objective function for 1 forces the signal on the learned graph to be smooth (and hence contains only low frequencies [4]). The third term is a regularization term that ensures that the entries of $L_\mathcal{G}$ are not too large.

As motivated in the introduction, we consider the scenario when the observed signals are not necessarily smooth on the underlying graph. We assume that the observed signals are a sparse combination of graph eigenvectors $V_\mathcal{G}$, as described in (2). Such a model allows for the underlying graph representation to be compact, especially in settings like lateral inhibition, where large variation on the graph is natural.

$$\mathbf{x}_i = V_\mathcal{G}\, \mathbf{y}_i + \eta, \quad \|\mathbf{y}_i\|_0 \leq k \tag{2}$$

A constraint in (2) allows the observed data to be maximum k linear combinations of the eigenvectors. In other words, this model allows the graph signal $\mathbf{x}_i$ to have at most k nonzero frequency components; but these k components could be from any region of the spectrum. Suppose the matrix Y is obtained by stacking the GFTs of $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$ in the data: $Y = (\mathbf{y}_1\, \mathbf{y}_2\, \ldots \mathbf{y}_M)$, then we have

$$X = V_\mathcal{G} Y + \eta \tag{3}$$

Here η is the noise in observation. Given X, mathematically, the problem of graph learning using spectral constraints is to solve the optimization problem (4): we try to minimize the error between the observed signal and the original signal X by imposing the sparsity constraint on the observation matrix.

$$\begin{aligned} \underset{V_{\mathcal{G}}, Y}{\text{minimize}} \quad & \|X - V_{\mathcal{G}}\, Y\|_F^2 \\ \text{subject to} \quad & V_{\mathcal{G}}^{\mathsf{T}} V_{\mathcal{G}} = I, \\ & \|\mathbf{y}_i\|_0 \leq k, \quad \forall i \in \{1, \dots, M\} \end{aligned} \tag{4}$$

The first constraint (on $V_{\mathcal{G}}$) avoids the trivial solution and forces orthogonality on the eigenvector matrix. Unlike the earlier formulation in 1, our formulation in (4) does not include the quadratic variation term, hence we do not force our solution to consist of only low frequencies. Solving (4) is not straightforward as the objective function is non-convex due to the product of optimization variables. Moreover, the feasible set formed by the above constraints are non-convex due to the sparsity and orthogonality constraints. Our approach for finding the eigenbasis $V_{\mathcal{G}}$, coefficient matrix Y and graph $\mathcal{G}$ are discussed in the following section.

3. ALGORITHMS FOR GRAPH LEARNING UNDER SPECTRAL SPARSITY CONSTRAINTS

This section present our algorithm for solving the optimization problem (4). The algorithm consists of two steps. In the first step, we find the eigevector matrix $V_{\mathcal{G}}$, and in the second step we find the eigenvalues of the adjacency matrix $A_{\mathcal{G}}$. To find the eigenvector matrix, we solve (4) by the method of alternating minimization and tackle the sub-problems individually. We first fix the eigenbasis $V_{\mathcal{G}}$ and find the coefficient matrix Y, then fix the coefficient matrix Y to find the eigenbasis $V_{\mathcal{G}}$.

3.1. Estimating the coefficient matrix Y

Fixing the eigenbasis $V_{\mathcal{G}}$ in (4) results in the convex objective function in (5) with sparsity constraint, as in (4).

$$\begin{aligned} \underset{Y}{\text{minimize}} \quad & \|X - V_{\mathcal{G}}\, Y\|_F^2 \\ \text{subject to} \quad & \|\mathbf{y}_i\|_0 \leq k, \quad \forall i \in \{1, \dots, M\} \end{aligned} \tag{5}$$

Though the constraint set in (5) is non-convex, several relaxations can be used to obtain the convex problem. Since the matrix $V_{\mathcal{G}}$ is orthogonal, the solution is obtained by taking the top absolute k coefficients in any column of $V_{\mathcal{G}}^{\mathsf{T}} X$.

However, the sparsity k may not be known in practice. We estimate the sparsity level k by using the following technique: for each potential sparsity level k (starting from $k = 1$), we find Y as described above, and compute the pseudo error $\|X - V_{\mathcal{G}} Y\|_F^2 \,/\, \|X\|_F^2$. We pick the value of k for which the pseudo error is locally minimum.

3.2. Estimating the eigenbasis

Once the coefficient matrix Y is obtained from the previous step, we try to find the eigenbasis $V_{\mathcal{G}}$. We frame the following optimization problem that requires finding the nearest orthogonal matrix to the observation data:

$$\begin{aligned} \underset{V_{\mathcal{G}}}{\text{minimize}} \quad & \|X - V_{\mathcal{G}}\, Y\|_F^2 \\ \text{subject to} \quad & V_{\mathcal{G}}^{\mathsf{T}} V_{\mathcal{G}} \, = I \end{aligned} \tag{6}$$

The optimization problem (6) is an orthogonal Procrustes-problem [19]. The solution for such a problem is obtained by evaluating the singular value decomposition of matrix XY^{T}. Thus a problem equivalent to the above is

$$\max_{V_{\mathcal{G}}} \operatorname{tr}(Y^{\mathsf{T}} V_{\mathcal{G}}^{\mathsf{T}} X) \tag{7}$$

The solution for $V_{\mathcal{G}}$ is given by $V_{\mathcal{G}} = U_1 U_2^{\mathsf{T}}$ where U_1 and U_2 are the singular vectors of XY^{T}: i.e. $XY^{\mathsf{T}} = U_1 \Sigma U_2^{\mathsf{T}}$. The process is iterated until the convergence criterion is met.

Algorithm for Topology Inference

```
Given: Observations $X$, sparsity $k$
Initialisation: Intialize $V_{\mathcal{G}}$ as a random $N \times N$ orthogonal matrix
Step1: Estimation of the eigenbasis $\mathcal{V}_{\mathcal{G}}$
currentObj = $\|X - V_{\mathcal{G}}\, Y\|_F$ , previousObj = $-\infty$
while | currentObj − previousObj |$\leq \epsilon$ do
    $Y \leftarrow V_{\mathcal{G}}^{\mathsf{T}} X$
    $Y \leftarrow$ keep the largest $k$ absolute coefficients.
    $SVD(XY^{\top}) = U_1 \Sigma U_2^{\mathsf{T}}$
    $V_{\mathcal{G}} \leftarrow U_1 U_2^{\mathsf{T}}$ to update $V_{\mathcal{G}}$
    previousObj $\leftarrow$ currentObj
    currentObj $\leftarrow \|X - V_{\mathcal{G}}\, Y\|_F$
end while
Step2: Learning Topology from
Compute the adjacency matrix $A_{\mathcal{G}}$ using equation 8
```

Even though we are applying the standard co-ordinate descent framework, it is worthwhile to note that each of the resulting sub-problems (sparse recovery problem and orthogonal Procrustes problem) is very well studied and has provable solutions. This probably explains the reasonably fast convergence of the proposed iterative technique.

3.3. Estimating the adjacency matrix

Once the eigenbasis $V_{\mathcal{G}}$ is known, the next step is to obtain the graph $\mathcal{G}$. To this end, we solve the constrained optimization problem (8) that enforces the properties of a valid adjacency matrix. This stage is similar to other such approaches in the graph learning literature [9]. The algorithm is outlined in below.

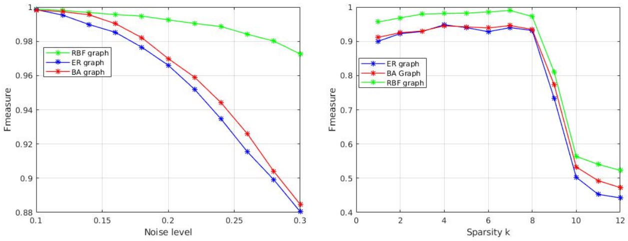
$$\begin{aligned} & \underset{A_{\mathcal{G}}, \Lambda}{\arg\min} \, \|A_{\mathcal{G}}\|_1 \\ & \quad \text{s.t. } A_{\mathcal{G}} = V_{\mathcal{G}} \Lambda V_{\mathcal{G}}^{\mathsf{T}}, A_{\mathcal{G}} \geq 0, A_{\mathcal{G}} \mathbf{1} \geq \mathbf{1} \end{aligned} \tag{8}$$

Each of our iterations in the first step of the algorithm is bottlenecked by an SVD computation of an $N \times N$ matrix (where N is the number of nodes in the graph). The computational complexity per iteration for our algorithm is $O(N^3)$, which is polynomial in the number of nodes. In practice, we observe convergence in a few iterations (about 25 iterations on average for $N = 20$ node graphs).

4. RESULTS AND DISCUSSION

We evaluate our algorithm on a synthetic dataset. We first generate a ground truth graph and synthetic dataset of graph signals for the ground truth graph. Then we apply our algorithm (and other algorithms in the literature) to compare the results. We use the following three models for generating the ground truth graph. 1) RBF: The N vertices are placed randomly in the unit square, and edges weights are set to be $exp(-d(i,j)^2/2\sigma^2)$ where $d(i,j)$ is the Euclidean distance between vertices i and j, and $\sigma = 0.5$. We only keep the edges with weights greater than 0.75. 2) ER: Erdős-Rényi model with edge probability 0.2 is used to create edges between the nodes with probability 0.2. 3) BA: Barabasi-Albert model (a scale-free random graph). We build the graph adding one vertex at a time and using the preferential attachment mechanism. When adding a new vertex, we put edges from the new vertex with probability equal to the ratio of the degree of the existing vertex to the total number of existing edges in the graph. The ER and BA graphs in our experiments have unit edge weights. Similarly to other works [9], [4], [7] we generate graphs with nodes $N = 20$.

Starting with a ground truth graph, we generate $M = 300$ graph signals according to our model in (2). Each of the graph signals is generated as a random sparse combination of eigenvectors of the graph previously generated, with the maximum sparsity $k_{max} = 5$. The non-zero coefficients are picked uniformly at random from $[-2,-1] \cup [1,2]$. Before applying the learning algorithm, random Gaussian noise is added to the graph signals to create a synthetic dataset.



(a) (b)

Fig. 2: (a) F-measure for different noise levels of different graphs. The noise level is variance of each entry of η in (2). (b) Plot of F-measure as a function of sparsity

The performance of the proposed technique is evaluated with the widely used F-measure (harmonic mean of precision and recall) [20] [4] [7]. The F-measure calculates the deviation/similarity between the edges of the learned graph and those of the ground truth graph. The higher the F-measure is, the closer the learned and ground truth graphs are. Once an algorithm is applied to the noisy synthetic dataset, the F-measure is obtained. This is then averaged over 100 independent realizations of noise, and further averaged over 100 random graphs from a given model. The sizes of the graphs and the number of graphs over which averaging is done are similar to other graph learning algorithms. Based on this framework, we perform the following experiments.

F-measure comparison on noisy dataset			
Algorithm	ER graph	BA graph	RBF graph
Proposed	$\mathbf{0.8804}$	$\mathbf{0.8964}$	$\mathbf{0.9726}$
Dong [4]	0.3256	0.3138	0.4414
Kalofolias [7]	0.3445	0.3260	0.4793
Segarra [9]	0.2953	0.2787	0.4013
Maretic [10]	0.2903	0.3089	0.4858
Chepuri [8]	0.2117	0.2005	0.3172

Table 1: F-measure comparison on existing algorithms.

The table 1 compares the F-measure of the proposed algorithm with other graph learning algorithms in the literature on the synthetic dataset with noise level 0.3. Note that the data used to evaluate the state-of-the-art algorithms conforms to our model (2), and so it may not be globally smooth. It is evident that the proposed algorithm outperforms the existing algorithms with significantly higher F-measure under the noise perturbation: this is because the proposed algorithm better captures the wideband graph frequency spectra than the existing algorithms.

Figure 2a plots the F-measure for our proposed algorithm as a function of the added noise level. It appears that our algorithm performs better for RBF than for ER and BA graphs. This is because ER and BA are unweighted graphs, and thus their inference is prone to discretization errors in our framework: the optimization problem in (8) does not impose binary constraints on the entries of $A_{\mathcal{G}}$. We observe that our proposed algorithm is fairly robust to additive noise. Figure 2b shows the impact of the spectrum sparsity of graph signals on the performance of the proposed algorithm. As shown, the F-measure decreases rapidly after about 40% (8 out of 20) of sparsity.

5. CONCLUSIONS AND FUTURE WORK

We have proposed novel model and criterion to learn graphs from wideband graph signal data, motivated from problems in neuro-imaging and lateral inhibition, and an efficient algorithm for associating a graph topology to such data. A comparison of our algorithm with existing graph learning algorithms under different graph signal spectra is presented to show the advantages of our method. Future research directions include testing the proposed algorithm on real datasets from neuro-imaging, further understanding the theoretical basis of the proposed algorithm and Laplacian based reconstruction with prior knowledge on one of the eigenvectors.

6. REFERENCES

[1] Ed Bullmore and Olaf Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature Reviews Neuroscience*, vol. 10, no. 3, pp. 186, 2009.

[2] 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.

[3] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[4] Xiaowen Dong, Dorina Thanou, Pascal Frossard, and Pierre Vandergheynst, "Learning laplacian matrix in smooth graph signal representations," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, 2016.

[5] Xiaowen Dong, Dorina Thanou, Michael Rabbat, and Pascal Frossard, "Learning graphs from data: A signal representation perspective," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 44–63, 2019.

[6] Gonzalo Mateos, Santiago Segarra, Antonio G Marques, and Alejandro Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 16–43, 2019.

[7] Vassilis Kalofolias, "How to learn a graph from smooth signals," in *Artificial Intelligence and Statistics*, 2016, pp. 920–929.

[8] Sundeep Prabhakar Chepuri, Sijia Liu, Geert Leus, and Alfred O Hero, "Learning sparse graphs under smoothness prior," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 6508–6512.

[9] Santiago Segarra, Antonio G Marques, Gonzalo Mateos, and Alejandro Ribeiro, "Network topology inference from spectral templates," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 3, pp. 467–483, 2017.

[10] Hermina Petric Maretic, Dorina Thanou, and Pascal Frossard, "Graph learning under sparsity priors," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Ieee, 2017, pp. 6523–6527.

[11] Amin Kheradmand and Peyman Milanfar, "A general framework for regularized, similarity-based image restoration," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5136–5151, 2014.

[12] Weiyu Huang, Thomas AW Bolton, John D Medaglia, Danielle S Bassett, Alejandro Ribeiro, and Dimitri Van De Ville, "A graph signal processing perspective on functional brain imaging," *Proceedings of the IEEE*, 2018.

[13] Weiyu Huang, Leah Goldsberry, Nicholas F Wymbs, Scott T Grafton, Danielle S Bassett, and Alejandro Ribeiro, "Graph frequency analysis of brain signals," *IEEE journal of selected topics in signal processing*, vol. 10, no. 7, pp. 1189–1203, 2016.

[14] Jeffrey A Kleim, Scott Barbay, Natalie R Cooper, Theresa M Hogg, Chelsea N Reidel, Michael S Remple, and Randolph J Nudo, "Motor learning-dependent synaptogenesis is localized to functionally reorganized motor cortex," *Neurobiology of learning and memory*, vol. 77, no. 1, pp. 63–77, 2002.

[15] Ana S Rufino Ferreira, Justin Hsia, Murat Arcak, Michel Maharbiz, and Adam Arkin, "Pattern formation with a compartmental lateral inhibition system," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 5413–5418.

[16] Stefania Sardellitti, Sergio Barbarossa, and Paolo Di Lorenzo, "Graph topology inference based on sparsifying transform learning," *IEEE Transactions on Signal Processing*, vol. 67, no. 7, pp. 1712–1727, 2019.

[17] Pierre Humbert, Batiste Le Bars, Laurent Oudre, Argyris Kalogeratos, and Nicolas Vayatis, "Learning laplacian matrix from graph signals with sparse spectral representation," .

[18] Aliaksei Sandryhaila and Jose MF Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, 2014.

[19] John C Gower, Garmt B Dijksterhuis, et al., *Procrustes problems*, vol. 30, Oxford University Press on Demand, 2004.

[20] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze, "Introduction to information retrieval," *Natural Language Engineering*, vol. 16, no. 1, pp. 100–103, 2010.