Learning spatial filters from EEG signals with Graph Signal Processing methods

Pierre Humbert[1,2], Laurent Oudre[1,2] and Clément Dubost[1,2,3]

Abstract— **In this paper, we propose to learn a spatial filter directly from Electroencephalography (EEG) signals using graph signal processing tools. We combine a graph learning algorithm with a high-pass graph filter to remove spatially large signals from the raw data. This approach increases topographical localization, and attenuates volume-conducted features. We empirically show that our method gives similar results that the surface Laplacian in the noiseless case while being more robust to noise or defective electrodes.**

Clinical relevance— **The proposed method is an alternative to the surface Laplacian filter that is commonly used for processing EEG signals. It could be used in cases where this standard approach does not provide satisfying results (low signal-to-noise ratios due to a low number of epochs, defective electrodes). This could be particularly interesting in case of an electrode defect, as it can happen in clinical practice.**

I. Introduction

Electroencephalography (EEG) is commonly used to track the brain activity by measuring the scalp electrical potentials generated by cortical postsynaptic currents. However, the analysis of EEG data remains a major challenge. Indeed, because the spatial–temporal information is mixed at the source level, the signal recorded from each electrode reflects a complex combination of electrophysiological dynamics from multiple brain regions [1]. To extract meaningful and physiologically interpretable patterns of activity from EEG signals, one standard approach consists of using filters (temporal and/or spatial) that will allow to highlight the phenomena of interest. Although there are several different types of temporal filters, most of the filters that are applied in cognitive electrophysiology (e.g., Morlet wavelet convolution, filter-Hilbert, and short-time FFT-based procedures such as Welch's method or multitapers) produce nearly identical results [1]. This can be contrasted with spatial filters, the topic of this paper.

In EEG analysis, a spatial filter is a mathematical method that produces new values by taking weighted combinations of signals from other electrodes. As in temporal filtering, the purpose is to highlight features present in the raw data but difficult to observe without applying any transformations. One popular example of such filter is the surface Laplacian. The most common motivations for using the surface Laplacian are to increase topographical localization, to facilitate electrode-level connectivity analyses, and to attenuate volume-conducted features of the data. However, because the surface Laplacian is inherently computed from signal differences, it is very sensitive to noise [2] e.g. if one electrode is defective, it would affect all the others surrounding in the filter's computation. Another concern is linked to the difficulty of computing the surface Laplacian at the edge of the EEG montage. Indeed, while an approximation, (usually referred to as Hjorth's approximation) [3] is used for central electrodes, we cannot compute it for estimates along the border of the electrode grid and in this case, a less accurate approximation has to be used.

To overcome these issues, one idea is to construct adaptive filters that will be able to model the irregular domain the signals are residing on. Thus, rather than considering that the neighbouring electrodes record similar signals, we quantify the similarity between electrodes based on their corresponding signals. Two electrodes that are not necessarily close in space could now be close in the resulting irregular domain. To define this domain, Graph Signal Processing (GSP) [4] has emerged as a powerful alternative to standard approaches. In this formalism, a graph defines a support (the irregular domain), and signals, now called graph signals, are defined on this support. This allows to capture the structure on which a signal evolves, thus providing more information than considering the signal alone. By generalizing concepts and tools of signal processing to signals recorded over graphs, GSP has proven its success in many tasks such as reconstruction [5], sampling [6], and filtering [7].

In this paper, we show that it is possible to automatically learn a spatial filter from the raw signals. More specifically, using techniques from the GSP field, we first learn a graph on which the raw EEG signals are smooth and well-represented, a property known as bandlimitedness. Then, we removed the graph low-frequencies (i.e. the spatially large signals) present in the raw signals with a graph filter constructed from the learned graph. Results suggest that our method is more robust to defective electrodes than the surface Laplacian. Furthermore, the modularity of our framework allows to define multiple types of spatial filters which may pave the way to others possibilities of analysis.

II. Background

a) Surface Laplacian: The proposed method is an alternative to the surface Laplacian, also commonly referred to as Current Source Density (CSD) or Scalp Current Density (SCD) [8], [1]. This is a spatial filter that is useful for attenuating volume conduction effects for better connectivity analyses. It is computed using the methods from [9]. The

[1] Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli, F-91190 Gif-sur-Yvette, France
[2] Université de Paris, CNRS, Centre Borelli, F-75005 Paris, France
[3] Service d'anesthésie-réanimation, Service de Santé des Armées, Hôpital d'Instruction des Armées Bégin, F-94160 Saint Mandé, France

three parameters for the Laplacian are smoothing (lambda), Lagrange order (number of iterations when computing the Legendre polynomial), and spherical spline order (m).

b) Graph notations: Throughout the paper, let consider a weighted and undirected graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of N nodes and $\mathcal{E}$ is a set of edges. The combinatorial Laplacian matrix of this graph is a N by N matrix defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}$ is the degree matrix and $\mathbf{W}$ the weight matrix. Since G is an undirected graph, $\mathbf{L}$ is a symmetric and positive semi-definite matrix verifying $\mathbf{L} = \mathbf{X}\mathbf{\Lambda}\mathbf{X}^\mathsf{T}$ with $\mathbf{\Lambda}$ the diagonal matrix of non-negative eigenvalues of $\mathbf{L}$ and $\mathbf{X}$ the matrix of the corresponding eigenvectors as columns. Assuming that G has one connected component, $\mathbf{L}$ has $\lambda_1 = 0$ for first eigenvalue associated with the eigenvectors $\mathbf{X}_{:,1} = \mathbf{1}_N/\sqrt{N}$ with $\mathbf{1}_N$ the unitary vector of size N. **Care should be taken not to confuse the Laplacian matrix and the surface Laplacian in the following.**

c) Graph Signal Processing: A graph signal is a function $\mathbf{y} : \mathcal{V} \to \mathbb{R}^N$ assigning a scalar value to each node of a graph G. This function can be represented as a vector $\mathbf{y} \in \mathbb{R}^N$, where $\mathbf{y}_i$ is the function value at the i-th node. The eigenvectors of the Laplacian of G provide a Fourier-like basis for graph signals, allowing to decompose any signal into its spectral components. From this formalism, the Graph Fourier Transform (GFT) of $\mathbf{y}$ is defined by $\mathbf{h} = \mathbf{X}^\mathsf{T}\ \mathbf{y}$. A K-bandlimited graph signal is thus a signal for which $\mathbf{h}_i \neq 0$ in K entries [4]. Finally, a smooth graph signal is a signal for which the quantity $\mathbf{y}^\mathsf{T}\mathbf{L}\mathbf{y}$ is small.

III. Proposed method

In our context, the input corresponds to multichannel EEG signals which can be represented by a matrix $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_T]$ in $\mathbb{R}^{P\times T}$, with P the number of channels and T the number of time points. Throughout this paper our goal is to learn the underlying structure of the channels in order to construct an adapted spatial filter. To do so, we consider a particular factor analysis model first introduced in [10]: $\forall i$, $\mathbf{y}_i = \mathbf{z}_i + \varepsilon_i$ where $\mathbf{y}_i$ follows a $\mathcal{N}(0, \mathbf{L}^\dagger + \sigma\mathbf{I}_P)$ with noise parameter $\sigma \in \mathbb{R}$ and pseudo-inverse $\dagger$; Here, $\mathbf{L}$ is the Laplacian of the spatial graph we want to learn. These assumptions implies that $\mathbf{Y}$ is smooth and (mutually) K-bandlimited under the spatial graph characterized by $\mathbf{L}$. In matrix notation, this means that (i) $\mathrm{tr}(\mathbf{Y}^\mathsf{T}\mathbf{L}\mathbf{Y})$ is small and (ii) there exist a set of indices $\mathcal{K}$ of size K such that $\mathbf{H}_{\mathcal{K},:} = 0$ where $\mathbf{H}$ has the same size as $\mathbf{Y}$ and corresponds to the spectral representation of the graph signals through the GFT. Note that, in our context, smoothness makes sense because of the volume conductivity which implies that electrodes tend to record very similar signals.

A. Denoising with graph learning

To learn $\mathbf{L}$, we choose to minimize the following objective function:

$$\min_{\mathbf{H},\mathbf{X},\mathbf{\Lambda}} \|\mathbf{Y} - \mathbf{X}\mathbf{H}\|_F^2 + \alpha\|\mathbf{\Lambda}^{1/2}\mathbf{H}\|_F^2 + \beta\|\mathbf{H}\|_{2,1} \ , \quad (1)$$

$$\text{s.t.} \quad \begin{cases} \mathbf{X}^\mathsf{T}\mathbf{X} = \mathbf{I}_P,\ \mathbf{x}_1 = \frac{1}{\sqrt{N}}\mathbf{1}_P \ , & \text{(a)} \\ (\mathbf{X}\mathbf{\Lambda}\mathbf{X}^\mathsf{T})_{k,\ell} \leq 0 \quad k \neq \ell \ , & \text{(b)} \\ \mathbf{\Lambda} = \mathrm{diag}(0, \lambda_2, \ldots, \lambda_P) \succeq 0 \ , & \text{(c)} \\ \mathrm{tr}(\mathbf{\Lambda}) = N \in \mathbb{R}_*^+ \ , & \text{(d)} \end{cases}$$

where $\mathbf{I}_P$ is the identity matrix of size P, $\mathrm{tr}(\cdot)$ denotes the trace of an input matrix, and $\mathbf{\Lambda} \succeq 0$ indicates that the matrix $\mathbf{\Lambda}$ is semi-definite positive.

This problem conjointly learns the Laplacian $\mathbf{L}$ (i.e. $(\mathbf{X}, \mathbf{\Lambda})$) and a bandlimited smooth approximation $\mathbf{X}\mathbf{H}$ of the true EEG signals $\mathbf{Y} - \mathcal{E}$. Finally, notice that $\widehat{\mathbf{X}}\widehat{\mathbf{H}}$ is a filtered version of $\mathbf{Y}$ obtained with a non-linear low-pass filter adapted to the learned graph $\widehat{\mathbf{L}}$. Hence, this approximation should have less noise than $\mathbf{Y}$.

The solution of (1) can be computed using one of the two algorithms proposed in [11], [10]. The first one solved the problem by combining barrier methods, alternating minimization, and manifold optimization. The second one is a relaxed algorithm which is more scalable with respect to the graph dimensions. In the following, we will use the relaxed algorithm.

B. High-pass graph filtering

From the learned Laplacian $\widehat{\mathbf{L}}$, we can construct a spatial graph filter that will adapt to the structure of the recorded signals. Because of the volume conductivity problem, we need to remove parts of the signals which are common in all channels i.e. spatially broad features which are characterized by low frequencies on the learned graph. In the following, we will therefore design a high-pass graph filter.

Formally, a graph filter is a linear operator acting on a graph signal by amplifying or attenuating parts of its spectrum [12]. Denoting this operator by F, a filtered graph signal $\mathbf{y}_F$ is then obtained through $\mathbf{y}_F = F\mathbf{y}$. Probably one of the most used low-pass graph filter is the one obtained via Tikhonov regularization. Given a scalar $\gamma \geq 0$ and a graph signal $\mathbf{y}$, the filtered signal $\mathbf{y}_F$ is the solution of

$$\mathbf{y}_F = \arg\min_{\mathbf{s}} \|\mathbf{y} - \mathbf{s}\|_2^2 + \gamma\mathbf{s}^\mathsf{T}\widehat{\mathbf{L}}\mathbf{s} \ . \quad (2)$$

The left term asks for a signal to be close to the input $\mathbf{y}$, while the second term enforces a smoothness prior. The solution has an explicit formulation given by $\mathbf{y}_F = \left(\mathbf{I} + \gamma\widehat{\mathbf{L}}\right)^{-1}\mathbf{y} = F\mathbf{y}$ with F the low-pass graph filter associated to (2). As this filter extracts the low frequencies of the input signal that we want to remove, we will use the residual $\hat{\mathbf{y}} = (\mathbf{y} - \mathbf{y}_F)$ as the high-pass version of our original signal. Note that, we filter the reconstruction $\widehat{\mathbf{X}}\widehat{\mathbf{H}}$ instead of $\mathbf{Y}$ as it is consider closer to the true signals. The full process of our method is provided in Algorithm 1.

IV. Results

Our approach is tested both on synthetic and real EEG data. All experiments are performed in `Python` using MNE [13]. Code is available online.

Algorithm 1 Learned spatial graph filtering

Input: $\mathbf{Y} \in \mathbb{R}^{P \times T}, \alpha, \beta, \gamma$

▷ `Learn the Laplacian`

$\widehat{\mathbf{H}}, \widehat{\mathbf{L}} \longleftarrow$ Solve problem (1)

▷ `Filtering of the graph signals`

$\mathbf{Y}_F \longleftarrow \left(\mathbf{I} + \gamma\widehat{\mathbf{L}}\right)^{-1} \widehat{\mathbf{X}}\widehat{\mathbf{H}}$

▷ `Remove low frequencies`

$\widehat{\mathbf{Y}} \longleftarrow (\mathbf{Y}_F - \widehat{\mathbf{X}}\widehat{\mathbf{H}})$

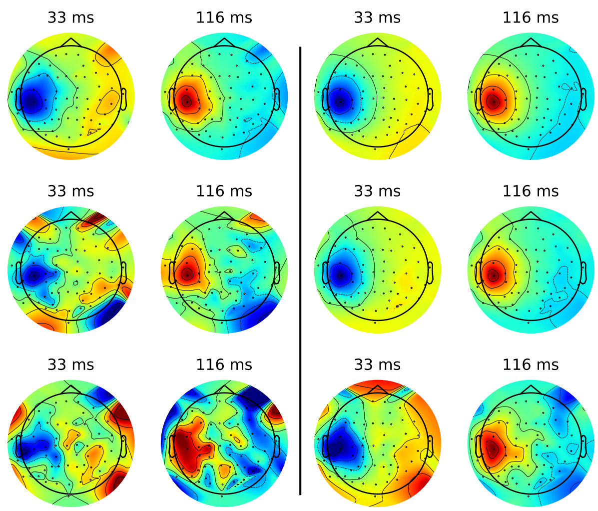

Fig. 1. On the left, the raw ERP at two given times. On the right, the approximations returned by Problem (1).

A. Experimental setup

a) Synthetic data: We simulate 60 EEG signals with sampling frequency of 600 Hz for 5 phase-locked epochs of 150 points (250 ms) as follows. First, we define a dipole time series consisting of a sine wave at 18 Hz with a peak amplitude of 10 nAm (signal at the cortical level). This dipole is activated in a region located at the bottom left of the brain (i.e. cortical areas around superior temporal sulcus). To obtain simulated sensor data we project the time series onto 60 virtual EEG electrodes arranged according to the 10-20 system (forward problem). Reference is set to average. Finally, we add a multivariate Gaussian noise. We repeat these operations 10 times and for 10 different amount of noise in order to have a Signal to Noise Ratio (SNR) between 10 and 60. We therefore obtain $10 \cdot 10 = 100$ datasets of size $(5 \times 60 \times 150)$. We report the Mean Square Error (MSE) between the true Event-Related Potential (ERP) i.e. the average of the true epochs – without noise – and the ERP approximation returned by Problem (1) (i.e. average of the refold version of $\widehat{\mathbf{X}}\widehat{\mathbf{H}}$ in $\mathbb{R}^{5\times60\times150}$ along the first dimension) as a function of the SNR.

b) Real data: We consider an EEG dataset from [14]. It consists in 64 EEG signals with sampling frequency of 256 Hz for 99 epochs of 640 points. The reference is set to the average. During the graph learning phase, we concatenate the epochs and obtain a matrix $\mathbf{Y}$ of size (64×40960). The graph is learned from this matrix. Parameters in (1) are set to $\beta = 0.001$ and $\alpha = 0.0001$. Once done, we solve (2) with $\gamma = 10$ and compute $\widehat{\mathbf{Y}} = \widehat{\mathbf{X}}\widehat{\mathbf{H}} - \mathbf{Y}_F$. Finally, we refold $\widehat{\mathbf{Y}}$ into an array of size $(99 \times 64 \times 640)$ and

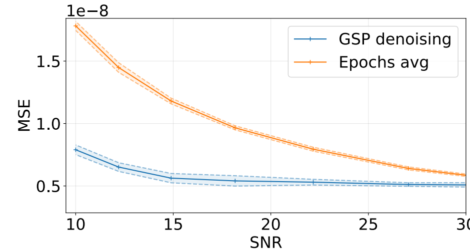

Fig. 2. Evolution of the MSE (avg $\pm$ std) between the true ERP (without noise) and: (blue) the ERP approximation returned by Problem (1) (denoising); (orange) the raw ERP, as a function of the SNR.

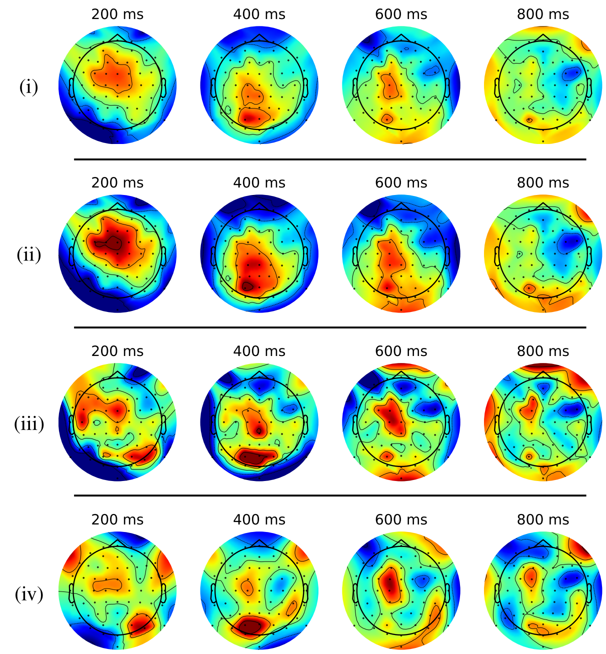

Fig. 3. Topographies at four given times from the experiment on real EEG data. From top to bottom: Topographies of (i) raw ERP, (ii) data returned by Problem (1), (iii) data returned by Problem (1) follow by (2), and (iv) data after applying the surface Laplacian.

average the 99 epochs to obtain the approximated ERP. The proposed method is compared to the surface Laplacian with smoothing parameter, Lagrange order, and spherical spline order respectively set to 10^5, 10, and 4. The smoothing and spline order values are typical parameters for 64-channel EEG [14].

B. Results on synthetic data

In this section, we highlight the capacity of Problem (1) to remove the noise from the raw EEG data. Note that because of the low number of epochs, only averaging them is not enough to properly remove the noise (Figure 1, left panel). The surface Laplacian is therefore not effective in this situation. In contrast, our method returns well-localized topographies even for a large amount of noise (Figure 1, right panel). This observation is corroborated by Figure 2 where our method allows a better denoising of the initial signal than an averaging of the epochs.

C. Results on real data

a) Sharper topographies: The raw ERP and its filtered version (with our GSP filter and the standard surface Laplacian filter) are displayed in Figure 3. This figure illustrates

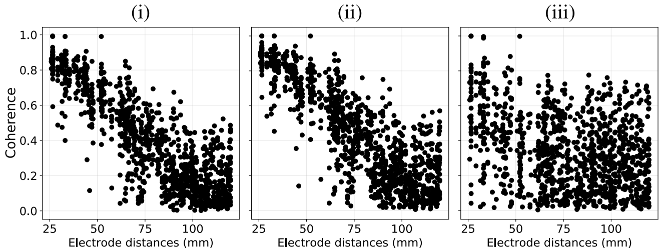

Fig. 4. Evolution of the Coherence at 8 Hz as a function of the electrode distances. From left to right: coherence on (i) raw ERP, (ii) after Problem (1) (denoising), and (iii) after Problem (2)

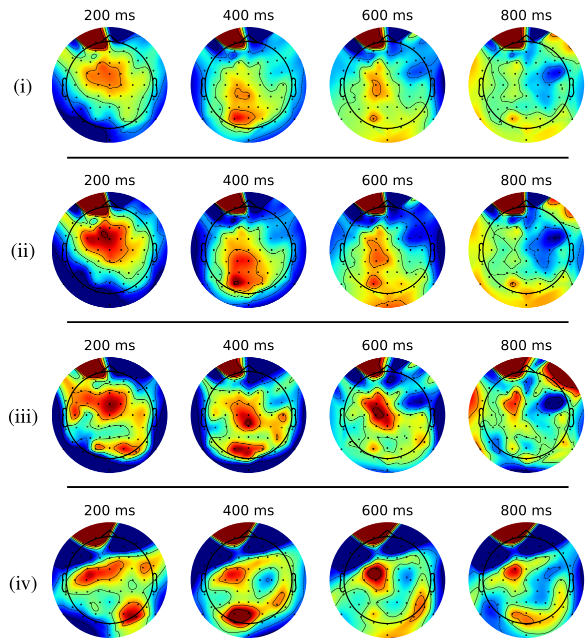

Fig. 5. Topographies at four given times from the experiment on real EEG data with a defective Fp1 electrode. From top to bottom: Topographies of (i) raw ERP, (ii) data returned by Problem (1), (iii) data returned by Problem (1) follow by (2), and (iv) data after applying the surface Laplacian.

the capacity of our filter to separate individual components based on differences in topography. Furthermore, we see that, while providing similar results as the surface Laplacian, the proposed method could tend to sharper topographies. To highlight the capacity of our method to attenuate the volume-conductivity, we display on Figure 4 the spectral coherence between channels [14] as a function of the distance between electrodes. We see that in the raw ERP (left panel of Figure 4) the coherence is driven almost entirely by interelectrode distance. In contrast, after our method or the surface Laplacian (middle and right panels of Figure 4), connectivity is no longer correlated with distance.

b) Robustness to malfunction: In this second experiment, we simulate the malfunction of an electrode at the edge of the EEG montage. To do so, the signal recorded at Fp1 (Top left electrode) is replaced by $\max(\mathbf{Y})$. The obtained results are displayed on Figure 5. Despite electrode Fp1 dysfunction, the GSP filter is globally not affected and returns comparable results as the ones of Figure 3 (when all electrodes are good). From the learned graph, we actually see that the node corresponding to Fp1 is almost disconnected from the other ones. Hence, it does not contribute in the graph filtering. In the other hand, because the surface Laplacian is computed from spatially close signal differences, it fails to recover the true topography near Fp1. For instance, at 800 ms, we clearly see that we loose information at the top right while the GSP filter works.

V. Conclusion

In this article, we introduced an GSP-based spatial filter that adapts to the data. We empirically shown that thanks to an adequate graph learning procedure, it automatically provides similar if not better results than the well-known surface Laplacian, especially in presence of defective electrodes and low signal-to-noise ratio. This preliminary work paves the way for further investigations on the links between EEG study and GSP, especially regarding the design of efficient GSP filters. Possible future investigation could be to consider other filters [12], time varying graph or graph products [15], and to incorporate physical prior in the graph learning process.

References

[1] M. X. Cohen, "Comparison of different spatial transformations applied to eeg data: a case study of error processing," *International Journal of Psychophysiology*, 2015.

[2] J. Kayser and C. E. Tenke, "Issues and considerations for using the scalp surface laplacian in EEG/ERP research: a tutorial review," *International Journal of Psychophysiology*, 2015.

[3] C. Carvalhaes and J. A. de Barros, "The surface laplacian technique in EEG: theory and methods," *International Journal of Psychophysiology*, 2015.

[4] 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," *Signal Processing Magazine*, 2013.

[5] S. Chen, A. Sandryhaila, J. M. Moura, and J. Kovačević, "Signal recovery on graphs: variation minimization," *IEEE Transactions on Signal Processing*, 2015.

[6] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Sampling of graph signals with successive local aggregations," *IEEE Transactions on Signal Processing*, 2016.

[7] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs," *IEEE Transactions on Signal Processing*, 2013.

[8] F. Perrin, O. Bertrand, and J. Pernier, "Scalp current density mapping: value and estimation from potential data," *IEEE Transactions on Biomedical Engineering*, no. 4, pp. 283–288, 1987.

[9] F. Perrin, J. Pernier, O. Bertrand, and J. Echallier, "Spherical splines for scalp potential and current density mapping," *Electroencephalography and Clinical Neurophysiology*, 1989.

[10] P. Humbert, B. Le Bars, L. Oudre, A. Kalogeratos, and N. Vayatis, "Learning Laplacian matrix from graph signals with sparse spectral representation," https://www.researchgate.net/publication/339553098_Learning_Laplacian_Matrix_from_Graph_Signals_with_Sparse_Spectral_Representation, 2019.

[11] B. Le Bars, P. Humbert, L. Oudre, and A. Kalogeratos, "Learning Laplacian matrix from bandlimited graph signals," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

[12] N. Tremblay, P. Gonçalves, and P. Borgnat, "Design of graph filters and filterbanks," in *Cooperative and Graph Signal Processing*. Elsevier, 2018.

[13] A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen *et al.*, "MEG and EEG data analysis with MNE-Python," *Frontiers in Neuroscience*, 2013.

[14] M. X. Cohen, *Analyzing neural time series data: theory and practice*. Cambridge: MIT press, 2014.

[15] T. Gnassounou, P. Humbert, and L. Oudre, "Adaptive subsampling of multidomain signals with product graphs," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021.