

# Notes on Graph Spectral Theory

## ACM Reference Format:

. 2023. Notes on Graph Spectral Theory. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 THE MAIN CHALLENGES OF SIGNAL PROCESSING ON GRAPHS

Graphs are widely used thanks to their generic data representation forms which are useful for describing the geometric structures of data domains in very different application fields, including social, energy, transportation, and brain modelisation. However, classical tools and techniques from signal processing can not be used seamlessly/candidly on graphs unlike audio signal and images [4].

- Graphs have no inherent ordering of the vertices, unlike images where pixel is uniquely identified by its position within the image. We therefore need algorithms that are node-order equivariant: they should not depend on the ordering of the nodes [1].
- Many graphs are irregular structures that lack a shift-invariant notion of translation, which is a key property of the Fourier transform.
- The analogous spectrum in the graph setting is discrete and irregularly spaced, and it is therefore non-trivial to define an operator that correspond to translation in the graph spectral domain.
- We may need to downsample a graph, and therefore we need a method to generate a coarser version of the graph that somehow captures the structural properties embedded in the original graph.
- We need localized transforms that compute information about the data at each vertex by using data from a small neighbourhood of vertices close to it in the graph

Graphs can also be very large but, in our study, we consider sparse population graphs, i.e. individuals or nodes are connected to a limited number of nodes<sup>1</sup>. Spectral graph theory has enabled constructing, analyzing, and manipulating graphs.

In this section, we present some basic definitions from spectral graph theory that will be needed to apply neural networks on graphs. As stated previously, we consider an undirected, connected, weighted graph  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$ .

<sup>1</sup>We also say that the number of edges is linear in the number of nodes.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

Conference'17, July 2017, Washington, DC, USA

© 2023 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 2 THE NON-NORMALIZED AND NORMALIZED GRAPH LAPLACIAN

The **non-normalized graph Laplacian**, also called the combinatorial graph Laplacian, is defined as  $\mathbf{L} = \mathbf{D} - \mathbf{W}$ , where the degree matrix  $\mathbf{D}$  is a diagonal matrix whose  $i$ th diagonal element,  $d_i$ , is equal to the sum of the weights of all edges incident to vertex  $i$  (e.g. the sum over the rows of  $\mathbf{W}$ ).

The graph Laplacian  $\mathbf{L}$  is a real symmetric matrix, it has therefore real, non-negative eigenvalues  $\{\lambda_l\}_{l=0,\dots,N-1}$ . We denote their associated orthonormal eigenvectors by  $\{u_l\}_{l=0,\dots,N-1}$ . In the following,  $\Lambda$  and  $\mathbf{U}$  denote the diagonal matrix of eigenvalues and the matrix of eigenvectors, respectively.

Since we consider connected graphs, the eigenvalue  $\lambda_0 = 0$  has multiplicity 1 [4]. (There are as many null eigen values as there are connected components in the graph.)

A popular practice is to normalize each weight  $W_{i,j}$  by a factor of  $\frac{1}{\sqrt{d_i d_j}}$ . Doing so leads to the **normalized graph Laplacian**, which

is defined as  $\tilde{\mathbf{L}} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} = \mathbf{I}_N - \mathbf{D}^{-1/2} \mathbf{W} \mathbf{D}^{-1/2}$

Note that the rows of  $\tilde{\mathbf{L}}$  do not sum to zero, unlike the rows of  $\mathbf{L}$ .

## 3 A GRAPH FOURIER TRANSFORM AND NOTION OF FREQUENCY

The Fourier transform for analogous function  $f$

$$\hat{f}(\xi) = \left\langle f \middle| e^{2\pi i \xi t} \right\rangle = \int_{\mathbb{R}} f(t) e^{-2\pi i \xi t} dt$$

is the expansion of a function  $f$  in terms of the complex exponentials, which are the eigenfunctions of the one-dimensional Laplace operator  $\Delta$  :

$$-\Delta(e^{2\pi i \xi t}) = -\frac{\partial^2}{\partial t^2} e^{2\pi i \xi t} = (2\pi i \xi)^2 e^{2\pi i \xi t}$$

Similarly, we can define the *Graph Fourier Transform*  $\hat{f}$  of any function on the vertices of  $\mathcal{G}$  as the expansion of  $f$  in terms of the eigenvectors of the graph Laplacian  $\hat{f}(\lambda_l) = \langle f | u_l \rangle = \sum_{i=1}^N f(i) u_l^*(i)$  : where  $u_l^*(i)$  is the conjugate of  $u_l(i)$  :

$$\hat{f} = \mathbf{U}^T f \quad (1)$$

The *inverse graph Fourier transform* is then given by  $f(i) = \sum_{l=1}^N \hat{f}(\lambda_l) u_l(i)$  :

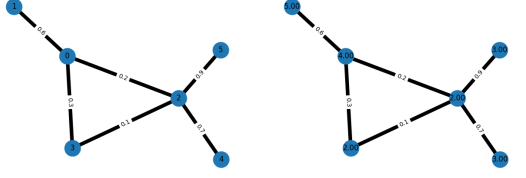
$$f = \mathbf{U} \hat{f} \quad (2)$$

Note that, in our case, the signal is  $f : \mathcal{V} \rightarrow \mathbb{R}^N$  that associate a feature vector to each node of the graph.

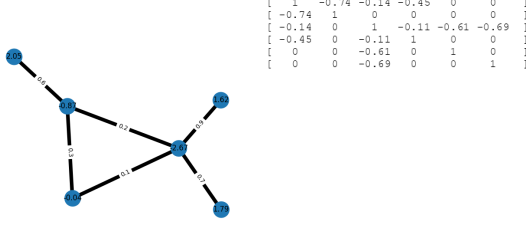
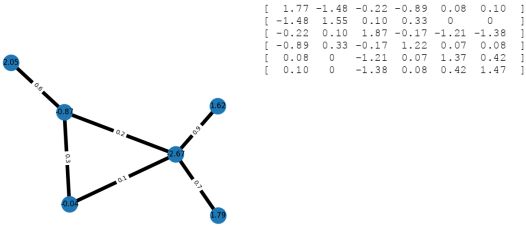
## 4 SPECTRAL GRAPH CONVOLUTIONS

Spectral convolution of the signal  $f$  with a filter  $g_\theta$  is defined as  $g_\theta * f = g_\theta(\tilde{\mathbf{L}})f = \mathbf{U} g_\theta(\Lambda) \mathbf{U}^T f = \mathbf{U} g_\theta(\Lambda) \hat{f}$ . Indeed, we recover the property that the convolution in the vertex domain is equivalent to a multiplication in the graph spectral domain.

Polynomial filters, defined as  $g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k$ , are commonly used in localized convolutions. Those filters are localized thanks to the lemma 5.2 in [2]. We can show that for some integer



(a) Toy graph

← indexes of the nodes | initial features values ( $K = 1$ ) →(b) ← After convolution with  $K = 2$  |  $\tilde{L} \rightarrow$ (c) Convolution with  $K = 3$  |  $\tilde{L}^2 \rightarrow$ **Figure 1: Convolution of a toy graph and the normalized laplacian matrix  $\tilde{L}^K$  for  $K = 1, 2, 3$ .**

$s$  and for any vertices  $m$  and  $n$  in the graph  $\mathcal{G}$ , if  $d_{\mathcal{G}}(m, n) > s$  then  $(L^s)_{m,n} = 0$ .  $d_{\mathcal{G}}(m, n)$  denotes the number of edges in the shortest path connecting the nodes  $m$  and  $n$ . It follows that a  $K$ -order

polynomial filter is strictly  $K$ -localized, i.e. it only depends on the  $K$ -hop neighbourhood of each vertex. In the studied paper [3], the authors chose  $K = 3$ .

Futhermore, according to the lemma 1, such filters can be well approximated by a truncated Chebyshev expansion of the form  $g_{\theta}(\tilde{L}) = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L})$  where  $T_k$  is the  $k$ th Chebyshev polynomial of the first kind.

LEMMA 1. *In the appropriate Sobolev space, the set of Chebyshev polynomials form an orthonormal basis, so that a function in the same space can, on  $-1 \leq x \leq 1$ , be expressed via the expansion :*

$$g(x) = \sum_{n=0}^{\infty} a_n T_n(x)$$

This decomposition significantly reduces the computational complexity of the convolution operator [3].

## 5 A SIMPLE ILLUSTRATION

We have implemented a simple application of the spectral graph convolution on a toy graph. On the illustration 1, we notice that the greater the order of the convolution, the more the signal is smoothed, and the less the graph laplacian have null coefficients.

For the example displayed in figure 1, we initialized the weight of the convolution to one.

## REFERENCES

- [1] Ameya Daigavane, Balaraman Ravindran, and Gaurav Aggarwal. [n. d.]. Understanding Convolutions on Graphs. 6, 9 ([n. d.]), e32. <https://doi.org/10.23915/distill.00032>
- [2] David K. Hammond, Pierre Vandergheynst, and R  mi Gribonval. [n. d.]. Wavelets on graphs via spectral graph theory. 30, 2 ([n. d.]), 129–150. <https://doi.org/10.1016/j.acha.2010.04.005>
- [3] Sarah Parisot, Sofia Ira Ktena, Enzo Ferrante, Matthew Lee, Ricardo Guerrero Moreno, Ben Glocker, and Daniel Rueckert. 2017. "Spectral Graph Convolutions for Population-Based Disease Prediction".
- [4] David I. Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. [n. d.]. The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains. 30, 3 ([n. d.]), 83–98. <https://doi.org/10.1109/MSP.2012.2235192> arXiv:1211.0053 [cs]