NODAL DOMAINS OF THE DIRICHLET TO NEUMANN MAP ON GRAPHS

RAM BAND

We consider graphs with finite sets of vertices $\mathcal{V}$ and edges $\mathcal{E}$. Each edge $e \in \mathcal{E}$ connects some two vertices, $x, y \in \mathcal{V}$, and we will also use the notation $e = (x, y)$. See an example of a graph and indexing of its vertices in Figure 1.

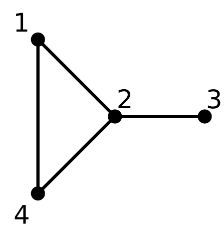

FIGURE 1. A simple graph with four vertices and four edges.

On each edge, we put some positive weight, and indicate this by the function $w : \mathcal{E} \to \mathbb{R}^{+}$, where we will use the notation w_{xy} for the value which w takes on the edge (x, y).

The weighted Laplacian on the graph is defined as

$$\begin{aligned} \mathbf{L} \, : \, \mathbb{R}^{|\mathcal{V}|} \; & \to \; \mathbb{R}^{|\mathcal{V}|} \\ (\mathbf{L}f)(x) \; & = \; \sum_{y \sim x} w_{xy} \left(f(x) - f(y) \right), \end{aligned}$$

where $y \sim x$ indicates that vertices x and y are connected by an edge.

We may now solve the eigenvalue problem, $\mathbf{L}f = \lambda f$, searching for the eigenvalues, the λ's, and the eigenvectors, the f's.

Once an eigenvector is given (see Figure 2-Left) we can consider its signs at the vertices (see Figure 2-Middle) and delete edges which connect a positive vertex to a negative one (see Figure 2-Right). This was, we obtain connected components of the graph on which the eigenvector has constant sign (either $+$ or $-$). Those connected components are called *nodal domains* (in the example of the figure, the eigenvector has only two nodal domains - one positive and one negative).

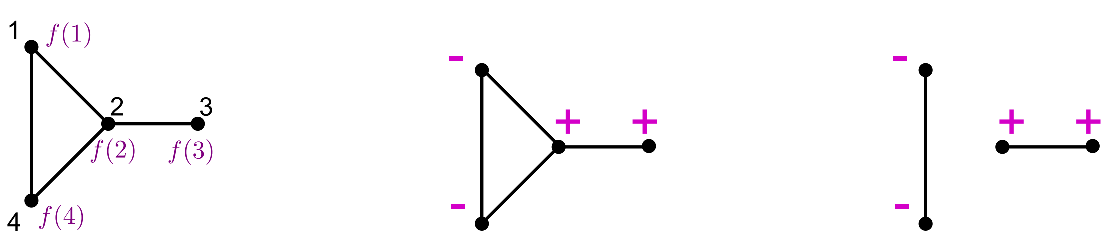

FIGURE 2. A simple graph with an eigenvector. In the example above the signs in the middle figure are dictated from the following $f(1) = -1,\ f(2) = 4,\ f(3) = 2,\ f(4) = -1$.

We are interested in counting how many nodal domains there are for every eigenvector.

Order the eigenvalues of $\mathbf{L}_w$ increasingly, $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{|\mathcal{V}|}$. For each eigenvalue λ_n we denote the corresponding eigenvector by f_n.

We then have the following theorem: the number of nodal domains of f_n is not greater than n.
This theorem is less than 20 years old (which is considered rather new in mathematics - compare that to the dates of other theorems you have learned so far in your courses :) and it was proven by Davies, Gladwell, Leydold and Stadler.

In our project we will try to prove a similar theorem for a different operator (matrix), which is defined on the graph. Instead of the weighted Laplacian, we will take as the operator the so-called Dirichlet to Neumann operator. This is a fascinating operator which is related to electric currents and voltages, to tomography in medicine and much more. We will not elaborate more here (you need to choose the project in order to find out more :)
A word of caution - keep in mind that we will not really be doing any electrical or medical applications. As pure mathematicians, we will restrict ourselves just to proving theorems on nodal domains (which can be fascinating enough on its own!)

What we are actually going to do?

- We will start by understanding the problem and its solution for the weighted Laplacian. For that we will need to read the paper of Davies, Gladwell, Leydold and Stadler. (which proves the theorem mentioned above).
- We will learn about the Dirichlet to Neumann operator on a graph and understand its properties.
- We will try to apply methods from the paper mentioned above for the study of the Dirichlet to Neumann operator.
- We will try to do numerics (in python or Matlab for example) and study different examples of graphs in order to understand the problem better.
- Maybe we will prove a theorem :)

Prerequisite.

- An excellent knowledge of Linear algebra (Algebra A and B) is a must! We will need to understand and explore various quadratic forms.
- Some knowledge of expressing eigenvalues in terms of variations over quadratic forms can help. This means for example the Rayleigh quotient (also called Rayleigh-Ritz quotient) and is sometimes studied in courses on functional analysis, variation methods, and some physics courses.
  This kind of knowledge is an advantage, but it is not essential that you have learned it before and we will be able to learn that together.
- Some knowledge in numerical programming might help. This means for example some acquaintance with python or Matlab.