GRAPH NODAL DOMAINS AND DATA

Wesley A. Hamilton

A dissertation submitted to the faculty at the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Mathematics in the College of Arts and Sciences.

Chapel Hill
2021

Approved by:

Jeremy L. Marzuola

Jingfang Huang

Shahar Kovalsky

J.S. Marron

Nicolas Fraiman

ABSTRACT

Wesley A. Hamilton: Graph Nodal Domains and Data
(Under the direction of Jeremy L. Marzuola)

This thesis proposes and develops a graph spectral flow for computing nodal counts and nodal deficiencies of graph Laplacian eigenvectors. Background on Laplace eigenfunctions and their nodal domains, as well as the corresponding results in the spectral graph theory literature, is given. We also review some effective tools that adapt spectral methods for the analysis of data, in particular the use of ratio cuts for partitioning and diffusion maps for dimensionality reduction. We then define two versions of the graph spectral flow and develop properties of each, after which examples of the spectral flow on a number of graphs are provided. Finally we mention ongoing lines of research related to both theoretical and applied aspects of graphs' nodal counts.

For my grandparents.

ACKNOWLEDGEMENTS

First and foremost I'd like to acknowledge my academic mentors. Jeremy Marzuola took me under his wing for four years, believing in me despite the (1) many non-thesis math things I would get involved in, (2) the many non-thesis non-math things I would get involved in, and (3) the many (non-thesis) math problems I'd pitch and play around with when I could've focused on the problems this thesis focuses on. His patience, support, and belief are what made this thesis and my academic career possible. Thank you. One important non-thesis math thing I started early on was a project with Steve Marron, who I also want to acknowledge and thank. I knocked on his office door one spring day and said I wanted to work on a TDA project, and as luck would have it he had one at the ready. Despite the fact that I had minimal programming and statistical experience, he was patience and continued to collaborate with me, all the while imparting some of his statistical wisdom. I'm truly grateful for the projects and opportunities I've been able to work on with Jeremy and Steve, as well as all of the future opportunities their collaborations continue to lead to.

I want to thank my thesis committee for their extensive comments on earlier drafts of this thesis.

It was a happy coincidence that my parents and sister ended up in the Triangle area while I was working on my PhD. Of course their support goes way beyond these four years: thank you Ralph and Nancy for unconditional love and support throughout my entire schooling, and thank you Nina for sharing an appreciation for puzzles. Kudos to Tatenda. Special acknowledgment goes to Taylor Enrico for her love and support,helping me to push through and finish this thesis.

Important support was afforded to me by the many math and non-math friends made during my graduate career (in no particular order): Ben Vadala-Roth, Brian Adam, Yiyan Shou, Nuch Aminian, Blake Keeler, Alyssa Vollaro, Taylor Rhoads, Kate Daftari, and Leo Brody. Many friends from before my time at UNC also provided much needed support: Becca Robbins, Sam Blandford, Manali Banerjee, Alex Miller, Grace Koclar, Shawn McCloy, Emily Meigs, Sofia Tsoni, Cody Ruffalo, Jake Keverline, Coley Clayton, and Amanda Emery. Thank you all!

Mathematical outreach has also been a major part of my academic career, so I want to thank some of my colleagues that have helped me put together and run some rather stellar programs: Linda Green, for letting me help out with the Chapel Hill Math Circle (and a whole bunch of other teaching mentorship), David Yavenditti, for helping me run Math Circle and MathGems; Paul Kruse, for Math Circle (and in general); and Kate Daftari, for really helping shape our Highschool Math Competition in Modeling workshops and helping ensure Math Circle has (applied) leadership going forward. I'd also like to thank Will Davis and Luke Conners for helping run MathGems my last semester at UNC to ensure a smooth transition, as well as numerous undergraduates who formed a core collection of great speakers: Yicheng Wang, Cooper Faile, Simon Bertron, Daniel Cantwell, Cooper Schoone, Nick Tapp-Hughes, and Scott Hallyburton. Finally I want to thank my co-organizers for the Workshop for Inclusive Teaching Practices: Katri Morgan, Paul Kruse, Suzanne Crifo, and Katherine Harris; may the WITP be a thing for years to come. Similarly, thanks to Hans Christianson and Katie Newhall for sharing an interest in improving DEI in our department and classrooms. Finally, I want to thank Joey Hart for introducing me to the SIAM Education Committee, and Katie Kavanagh and the other committee members for the support and opportunities that serving on a SIAM committee has given me.

The last acknowledgments I want to make are for two Triangle area institutions: UNC, for funding opportunities (including the i4 boundary spanner project and the writing center) and various professional development opportunities (particularly Brian Rybarczyk, the CIRTL liaison), and SAMSI, for a number of interesting workshops, funding me as an RA, and general mathematical support.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

Introduction

This thesis is concerned with a problem from spectral graph theory: given a network, and given a "resonant frequency" for the network, is it possible to count the number of regions where the "resonant frequency" is non-zero? In a sense the answer is yes, and this introductory chapter is meant to give a layperson explanation of the main results followed by an overview of the thesis.

1.1 Main Results

By spectral, we mean certain "resonant" properties of functions defined on a network of interest. A definition by analogy is the following: if you pour sand on a metal plate and vibrate the plate (by a speaker or violin bow), resonant standing waves will push the sand to a family of lines where the standing waves do not move; see Figure 1.1. These waves are called *eigenfunctions* (of the Laplacian), and the regions avoided by the sand are called *nodal domains*. Although this motivating example is defined for continuous domains, related constructions also exist for networks and have been effectively used for data analysis and machine-learning tasks. These constructions for networks are the focus of this thesis.

A network consists of nodes, edges that connect nodes, and edge-weights that characterize the similarity between connected nodes: if the edge-weight is large, the nodes are highly similar; if the edge-weight is small, the nodes are dissimilar but still similar enough to be connected; if the edge-weight is zero, no edge exists between the nodes. The network analogue for vibrating a membrane is through the network's *graph Laplacian*. In many ways the graph Laplacian describes how the network should vibrate if such a procedure was possible, since it gives a discretization of the same differential equation that describes vibrations in something like a metal plate.

While the patterns that appear on the plate are interesting in their own right, we can ask a slightly simpler question: how many regions are demarcated by the sand? Another way to ask this question is to take a snapshot of the vibrating plate, and look at the height of the plate within each nodal domain. Regions of the plate in which the height is positive (relative to the non-vibrating plate) are called positive nodal domains, and regions where the height is negative are called the negative nodal domains. Some nice results about counting

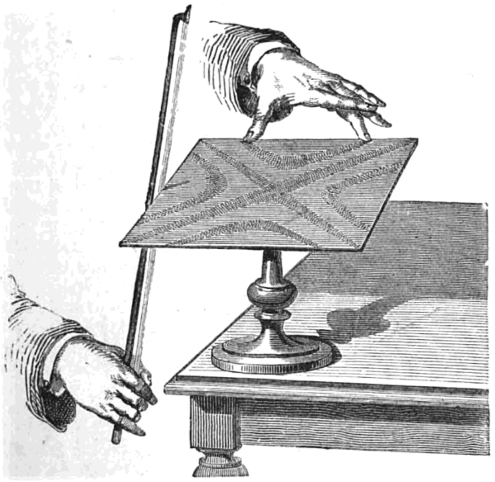

Figure 1.1: Vibrating a metal plate covered with sand will produce standing waves and illuminate nodal patterns in the sand. From [1].

these nodal domains exist, which we describe in the next chapter, and the same kind of question can be asked for the network: how many "nodal regions" are there, where a "nodal region" will correspond to collections of nodes that are connected to each other and all have the same "height". Though the network doesn't necessarily have any interpretable geometry, so that the notion of the "height" of a node isn't well-defined, we can use the resonant states of the network's graph Laplacian to give us something close.

In particular, the main result of this thesis is that:

Theorem 1.1.1. *The number of nodal domains of a network's resonant states can be counted in an explicitly computable way.*

See Theorem 3.2.6 and Theorem 3.3.11 for the mathematically precise versions of this theorem.

We're also interested in what these nodal counts can tell us from a data science perspective. We build intuition for this tool by seeing what the nodal counts are for a number of networks, both synthetic and real-world.

1.2 Overview of the Thesis

Chapter 2 introduces the continuum (/metal plate) and graph (/network) frameworks and intuition that underline the main results. Chapter 3 makes up the core of this thesis, in which Theorem 3.2.6 and Theorem 3.3.11 are set up and proven, and in which some open questions are posed about the behaviour of these resonant states and their relations to nodal counts. Chapter 4 provides a number of numerical results to build intuition for what these nodal counts look like in the context of data. Finally, Chapter 5 mentions some avenues of ongoing work, both on the theoretical side and the data-analytic side.

CHAPTER 2

Spectral Methods in Data Analysis

In this chapter we review the literature on nodal domains, in the continuum and graph setting, before reviewing recent applications of spectral methods to the analysis of point clouds and data. Sections 2.1 and 2.2 are adapted from [3], while Sections 2.3 and 2.4 contain preliminary results from ongoing research.

2.1 Motivation from the Analysis of PDEs

The starting point of our study is the spectral theory of the Laplacian. Recall that for a domain $\Omega \subset \mathbb{R}^n$, the Laplacian is the operator Δ defined by $\Delta f = -\sum_{i=1}^{n} \partial_{x_i}^2 f$, where f is taken from an appropriate function space. The Laplacian appears naturally in a number of physical models, including the diffusion of heat and the propagation of waves. We are primarily interested in the spectral theory of Δ: what are admissible eigenvalue/eigenfunction pairs (λ, ψ) such that $\Delta\psi = \lambda\psi$ wherever ψ and $\Delta\psi$ are defined. These eigenvalue/eigenfunction pairs are fundamentally important in a number of fields, but are interesting to study in their own right. In particular, the nodal domains and nodal sets of eigenfunctions ψ have been an active area of research since the 1800s. In this section we review the history of Laplace-eigenfunction nodal domains, and end with a spectral flow procedure that is the basis of this thesis.

2.1.1 Basic notions and Courant's Theorem

Suppose $\Omega \subset \mathbb{R}^n$ is a connected, bounded domain, and Δ is the usual Laplacian on $\mathbb{R}^n$ with either Dirichlet or Neumann boundary conditions on Ω. In the case of Dirichlet boundary conditions, the spectrum of Δ consists of eigenvalues $0 < \lambda_0 < \lambda_1 \leq \cdots$; in the case of Neumann boundary conditions, Δ has spectrum $0 = \lambda_0 < \lambda_1 \leq \cdots$ [4, 5]. Let $\phi_0, \phi_1, ...$ be the corresponding eigenfunctions. The **nodal sets** of ϕ_k are the connected components of $\{x \colon \phi_k(x) = 0\} =: \Gamma$, the **nodal domains** of ϕ_k are the connected components of $\Omega \setminus \Gamma$, and the **nodal count** $\nu(\phi_k)$ is the number of nodal domains. Of interest as well is the **nodal deficiency** $\delta(\phi_k)$, which can be thought of as quantifying the lack of oscillation of ϕ_k: when λ_k is a simple eigenvalue, $\delta(\phi_k) := k - \nu(\phi_k)$; when λ_k has multiplicity, set $k_* = \min\{s \colon \lambda_s = \lambda_k\}$ and define $\delta(\phi_k) := k_* - \nu(\phi_k)$. The main result about nodal counts is that $\delta(\phi_k) \geq 0$ for all k, or equivalently, $\nu(\phi_k) \leq k$.

The first result of interest to us is called the Sturm-Liouville theorem, which states that the nodal deficiency for Laplace eigenfunctions ϕ_k on a bounded interval always satisfy $\nu(\phi_k) = 0$ for all k.

Theorem 2.1.1. *Let* $\Omega = [0, 1]$ *and consider the Dirichlet eigenvalue problem*

$$\partial_{xx}u(x) = \lambda u(x) \textit{ for } x \in (0, 1) \textit{ and } u(0) = u(1) = 0.$$

Sort the eigenvalues $0 < \lambda_0 \leq \lambda_1 \leq \cdots$*, and call the corresponding eigenfunctions* $\phi_0, \phi_1, ...$ *Then* ϕ_k *has exactly* k *zeros in* $(0, 1)$.

One approach to proving Theorem 2.1.1 works as well for higher-dimensional domains, which we discuss next. See [6] for a relatively elementary discussion of this (and related) results, and [7, Chapter XIII] for a modern discussion.

In one dimension the nodal deficiency is identically zero, which may or may not be interesting depending on your needs. In higher dimensions, however, the nodal deficiency may be non-zero. This is the content of Courant's nodal domain theorem:

Theorem 2.1.2. *Let* $\Omega \subset \mathbb{R}^n, n \geq 2$*, be a bounded, connected domain with Laplacian* Δ*, and let* $0 < \lambda_1 \leq \lambda_2 \leq \cdots$ *be the ordered eigenvalues for the Dirichlet eigenvalue problem*

$$\begin{cases} \Delta u = \lambda u & \textit{in } \Omega, \\ u = 0 & \textit{on } \partial\Omega. \end{cases}$$

If $\phi_1, \phi_2, ...$ *are the associated eigenfunctions, then* $k \geq \nu(\phi_k)$.

The techniques we adopt in the graph setting mimic one set of approaches to proving Courant's theorem, so we provide a short sketch of the proof here. This proof is replicated from [4], and interested readers are directed there for details.

Sketch of proof. Let $\Omega_1, \Omega_2, ..., \Omega_k, \Omega_{k+1}, ...$ be the nodal domains of ϕ_k, and define new functions

$$\psi_j := \begin{cases} \phi_k|_{\Omega_j}, & \text{on } \Omega_j, \\ 0 & \text{on } \Omega \setminus \Omega_j. \end{cases}$$

One can then choose coefficients α_i such that the function $f = \sum_{j=1}^{k} \alpha_j \psi_j$ is orthogonal to each eigenfunction $\phi_1, ..., \phi_{k-1}$. By construction f is zero on Ω_{k+1}. But then f can be shown to be an eigenfunction for λ_k, and since f is zero on a domain with a limit point the maximum principle implies f must be identically zero. Thus ϕ_k can have no more than k nodal domains. □

The key thing here was that by restricting the eigenfunction to each nodal domain, we got a family of eigenfunctions for the same eigenvalue as ϕ_k. In the graph setting we use this same approach, though the discrete nature of graphs requires some care.

A corollary of this result is that the first eigenfunction for each nodal domain is signed (i.e. does no change sign), the first eigenvalue is simple, and higher eigenfunctions must be signed. A graph analogue of this result is proved in Proposition 3.1.4.

Proposition 2.1.3 ([4, §1.5 Cor. 2])**.** *With the same terminology as in Theorem 2.1.2,*

- ϕ_1 *has constant sign;*
- λ_1 *has multiplicity* 1*;*
- λ_1 *is characterized as being the only eigenvalue with eigenfunction of constant sign.*

The upper bound in Courant's theorem theorem 2.1.2 can only be attained finitely many times, as shown by Pleijel in [8]. The argument is nice: for a domain $\Omega \subset \mathbb{R}^2$, an eigenvalue/eigenfunction pair (λ_k, ψ), and nodal domains $\Omega_1, ..., \Omega_N$, one has the inequality $\frac{\text{Area}(\Omega_i)}{j_0^2 \pi} \geq \frac{1}{\lambda_n}$, where j_0 is the smallest positive zero of the Bessel function J_0. This result is a consequence of well known shape/eigenvalue optimization results, for which [9] is a modern reference. Adding these inequalities over all nodal domains gives $\frac{\text{Area}(\Omega)}{\pi j_0^2} \geq \frac{k}{\lambda_k}$. Now assuming that $\nu(\phi_n) = n$ for infinitely many n, we can use Weyl's law to get that $\frac{\text{Area}(\Omega)}{\pi j_0^2} \geq \frac{\text{Area}(\Omega)}{4\pi}$. Recall that Weyl's law in two dimensions states

$$\lim_{\lambda \to \infty} \frac{\#\{\lambda_k \leq \lambda\}}{\lambda} = \frac{\text{Area}(\Omega)}{4\pi},$$

where the eigenvalues in question are for the Laplacian with Dirichlet boundary conditions on Ω [10]. We have that $j_0 \cong 2.404...$ so $j_0^2 > 4$, which gives us a contradiction.

On the other hand, there exist eigenfunctions with arbitrarily large index that have few nodal domains. One procedure to construct such eigenfunctions is by perturbing the Laplacian by an $L^2(\Omega)$ potential function V, so that we work with eigenfunctions of $\Delta + V$ on Ω with appropriate boundary conditions. Another procedure that does not require changing the operator or domain is the following: let $\Omega = [0, \pi] \times [0, \pi]$

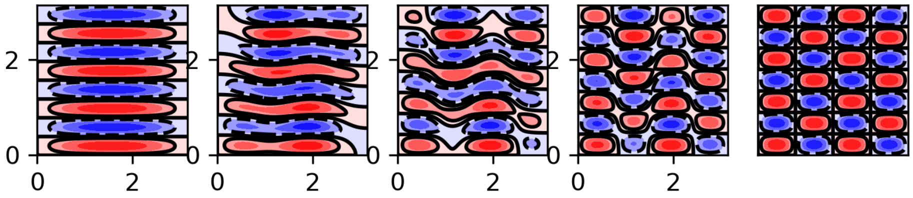

Figure 2.1: The nodal domains of $\phi_t(x, y) = (1 - t)\sin(k_1 x)\sin(l_1 y) + t\sin(k_2 x)\sin(l_2 y)$ for (left to right) $t = 0, 0.25, 0.5, 0.75, 1.0$. The nodal counts are, respectively, $\nu(\phi_t) = 8, 8, 8, 10, 28$.

and let $\psi_1 = \sin(k_1 x)\sin(l_1 y), \psi_2 = \sin(k_2 x)\sin(l_2 y)$ be two eigenfunctions of Δ with Dirichlet boundary conditions, such that $k_1^2 + l_1^2 = k_2^2 + l_2^2$. One such choice is $k_1 = l_2 \neq k_2 = l_1$; another family of choices is integers expressible as the sum of two squares [11]. Consider the 1-parameter family of eigenfunctions $\psi_t = (1 - t)\psi_1 + t\psi_2$ for $t \in [0, 1]$. As t varies from 0 to 1, the nodal domains of ψ_1 will merge and transform until they align with the nodal domains of ψ_2 when $t = 1$. Depending on the choice of k_i, l_i the number of nodal domains of ψ_t for $t \in (0, 1)$ may get as low as 2, but in general will be significantly fewer than the number of nodal domains for ψ_1 or ψ_2 for k_i, l_i large. As an explicit example, take $k_1 = 1, l_1 = 8, k_2 = 4, l_2 = 7$: $\nu(\psi_1) = 8$ but $\nu(\psi_2) = 28$, so ψ_t for varying t will have nodal counts between 8 and 28; see Figure 2.1.

Despite these observations, methods to compute nodal counts exist. We survey those next.

2.1.2 Continuum Spectral Flows

The main result followed in this thesis towards counting nodal domains, and nodal deficiencies, comes from [12]. Before discussing said result, we discuss aspects of the Dirichlet-to-Neumann operators for domains which play a fundamental role.

The Dirichlet-to-Neumann takes as input a(n appropriate) function f defined on the boundary of Ω, finds the harmonic extension u of f (so that $\Delta u = 0$ in Ω and $u|_{\partial\Omega} = f$), and then computes the outgoing normal derivative of u along $\partial\Omega$ (denoted $\partial_\nu u := \nabla u \cdot n$ with n the outgoing normal to $\partial\Omega$). Roughly speaking, the Dirichlet-to-Neumann operator associates a charge placed on the boundary of a domain (f) to the induced current across the boundary ($\partial_\nu u$). The spectral theory of Dirichlet-to-Neumann operators is an active area of research, and [13] is a recommended reference.

The Dirichlet-to-Neumann operator lets us associate boundary value information on the nodal sets to the behaviour of eigenfunctions for a perturbed Laplacian acting on the domain. The spectrum of these perturbed Laplacians turns out to count the nodal deficiency as the strength of the perturbation increases. One version of this result is as follows:

Theorem 2.1.4. *The nodal deficiency of ϕ_k is precisely the number of eigenvalues of the bilinear form*

$$B_\sigma(u, v) := \int_\Omega \nabla u \cdot \nabla v d\mu + \sigma \int_\Gamma uvdS$$

that cross $\lambda_k + \epsilon$ for $\epsilon > 0$ sufficiently small, as $0 \leq \sigma \to \infty$. Here $\Gamma = \{x \colon \phi_k(x) = 0\} \cap \Omega$, the nodal set of ϕ_k in the interior of the domain. Equivalently, the number of nodal domains of ϕ_k is exactly the multiplicity of the first Dirichlet eigenvalue on $\Omega \setminus \Gamma$, which are precisely the eigenvalues of the limiting bilinear form B_∞.

The motivation is an observation from [14] that the nodal deficiency is the Morse index (number of negative eigenvalues) of a certain operator acting on certain optimal partitions of the domain. This observation was extended in [15] to incorporate the spectrum of Dirichlet-to-Neumann operators in counting the nodal deficiency. The connection between the Dirichlet-to-Neumann operator and B_σ is by considering the nodal set as the "boundary": B_σ induces operators $L_\sigma = \Delta$ with Dirichlet boundary conditions on $\partial\Omega$ and the condition

$$\frac{\partial u}{\partial \nu_+} + \frac{\partial u}{\partial \nu_-} + \sigma u = 0$$

across the nodal set Γ of the eigenfunction in question, where $\nu_\pm$ is the normal vector to Γ leaving the positive/negative nodal region. This last condition is precisely that of a function being an eigenfunction of the Dirichlet-to-Neumann operator though, namely u is an eigenfunction of B_σ if and only if u is a Dirichlet-to-Neumann eigenfunction over Γ.

The proof of Theorem 2.1.4 is straightforward once the right framework is established, and hinges on two lemmas involving the eigenvalues of B_σ. Let $\gamma_k(\sigma)$ denote the eigenvalue branch of the kth eigenvalue of Δ, and $u_k(\sigma)$ the corresponding eigenfunction branch (which exist by general perturbation theory [16]). Let ϵ be a small parameter. Then

1. $\gamma_k(\sigma) = \lambda_k + \epsilon$ (the eigenvalue branch crosses λ_k) if and only if $-\sigma$ is an eigenvalue of the Dirichlet-to-Neumann operator, and
2. $\gamma_k'(\sigma) = \int_\Gamma u_k(\sigma)^2 dS \geq 0$; if $\gamma_k(0)$ is in the spectrum of B_∞ then $\gamma_k(\sigma)$ is constant, otherwise $\gamma_k(\sigma)$ is increasing.

Together these lemmas give that the eigenvalue branches are increasing and can only cross λ_k with positive slope. The eigenvalue branches that converge to λ_k are in one-to-one correspondence with the nodal domains of the original eigenfunction, establishing the result. In this thesis, we replicate this proof strategy for the corresponding bilinear forms on graphs. In particular, compare item (2) above to Lemma 3.2.3 and Lemma 3.3.5.

Generalizations of the spectral flow mentioned here exist, in which the spectrum of families of bounded, self-adjoint, Fredholm operators are studied. These constructions trace back to Atiyah-Patodi-Singer [17], and have seen extensive use in global analysis, mathematical physics, symplectic geometry, and more. See [18] for an introduction and overview of spectral flows in a functional analytic context, and particularly §5.2 for an overview of the spectral flow literature.

2.2 Motivation from Spectral Graph Theory

Spectral graph theory is an approach to analyzing graphs using eigenvalues and eigenvectors of certain matrices. Classically the graph's adjacency matrix and graph Laplacian have been used, but recent generalizations study objects called generalized Laplacians (Definition 3.1.1). While the underlying ideas are the same as for the continuum, graphs are fundamentally combinatorial objects and thus pose unique challenges and opportunities with respect to their spectral properties. One such challenge is defining a notion of nodal domain and nodal set. In this section we recast the nodal domain theorems from Section 2.1 into the graph setting, and highlight a variety of ideas and proof methods from recent years.

There are plenty of good introductions to spectral graph theory, each with a different flavour. For general algebraic graph theory with some spectral tools, see [19]. For an introductory network science treatment (with more spectral tools), see [20]. Three monographs focused on spectral graph theory, with a heavily analytic flavour, are [21, 22, 23].

2.2.1 Weak and Strong Nodal Domains

Here we consider, for simplicity, an unweighted graph and the standard graph Laplacian. The graph G consists of vertices V and edges E, with adjacency matrix A having a 1 in the (i, j) and (j, i) entries whenever $(i, j) \in E$, zeros elsewhere. The degree d_i of vertex i is the number of edges adjacent to i, and we collect these degrees in a diagonal matrix D. The graph Laplacian of G is defined to be $L = D - A$. Since A and D are symmetric, L is symmetric as well, and so its spectrum consists of real eigenvalues. Moreover, L is positive semi-definite, since $\langle u, Lu\rangle = \sum_{(i,j)\in E}(u_i - u_j)^2$, and so its spectrum consists of non-negative eigenvalues. The constant vector is in the kernel of L, so 0 is indeed in the spectrum of L.

In the continuum we were primarily interested in eigenfunctions of the Laplacian, and that interest carries over to the graph setting. Let (λ_k, ψ) be the kth eigenvalue/eigenvector pairs of L, with $\lambda_0 = 0$. The kinds of results we are interested in are analogues of Theorem 2.1.2, though one difficulty quickly becomes apparent: the notion of a "zero" for an eigenvector is less well-defined. Clearly if an eigenvector is zero on a vertex, we consider that to be a true zero. If the eigenvector changes sign across an edge, however, we do not have a single point in the graph that represents where the zero "should be" across said edge. If we collect all zero vertices and sign-change edges and call that our nodal set, we're left working with a collection of objects that may contain a mix of vertices and edges.

In the graph setting it turns out to be more fruitful to work directly with the nodal domains of the eigenvector, rather than the nodal sets. Moreover we focus on two kinds of nodal domains: **weak nodal domains** are the maximally connected subgraphs H of G such $\psi_i\psi_j \geq 0$ for all vertices $i, j \in H$, and **strong nodal domains** are the connected components of the subgraph H of G for which $\psi_i\psi_j > 0$ for all vertices $i, j \in H$ [24]. Intuitively, all vertices in a weak nodal domain must have the same sign or be zero. An equivalent definition for strong nodal domains is that they are the connected components of the subgraph for which sign-change edges ((i, j) such that $\psi_i\psi_j < 0$), zero edges ((i, j) such that $\psi_i\psi_j = 0$), and zero vertices (i such that $\psi_i = 0$) are removed. For both weak and nodal domains we have analogues of Courant's nodal domain theorem:

Theorem 2.2.1 ([22, Theorem 3.1],[25, Theorem 2])**.** *For any connected graph G, the kth eigenfunction ϕ_k of the graph Laplacian L has at most k weak nodal domains and at most $k + r - 1$ strong nodal domains, where r is the multiplicity of λ_k.*

Proofs of this theorem are generally fairly direct utilizing matrix-theoretic formulations. We highlight the approach in [22], in which the inequality for strong nodal domains is attained by showing that if ψ has m nodal domains, then $\lambda_m < \lambda_{k+r}$, so

$$\nu(\psi) = m \leq k + r - 1 < k + r.$$

The approach is heavily inspired by the proof of 2.1.2.

Another approach utilizes objects called magnetic Laplacians, for which the framework is the following: for each edge (i, j) we associate an edge w_{ij}, and for each *oriented* edge we associate a complex number B_{ij} so that $B_{ji} = \overline{B_{ij}}$. The collection of all oriented edges is denoted $\mathcal{E}$. The magnetic Laplacian, for the choice of magnetic field B, is the matrix L_B defined by the quadratic form

$$q_B(u) := \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} w_{ij} |u_i - e^{B_{ij}\sqrt{-1}} u_j|^2 - \sum_{i\in V} V_i |u_i|^2,$$

where $V_i = \sum_{(i,j)\in E} w_{ij}$ gives a notion of degree for each vertex. Note that L_B is Hermitian, and so has real spectrum

$$\lambda_1(L_B) \leq \lambda_2(L_B) \leq \cdots \leq \lambda_{|V|}(L_B).$$

See [26] for more on this construction.

The main result is another version of the nodal domain counts for Laplace eigenvectors:

Theorem 2.2.2 ([27, Theorem 1.1]). *Let β_1 be the minimal number of edges of G that need to be removed to turn G into a tree. If λ_k is simple and ψ is never zero, then the number of edges Z_ψ over which ψ changes sign satisfies $k - 1 \leq Z_e \leq k - 1 + \beta_1$.*

Moreover, the nodal defect $\delta(\psi) = Z_\psi - \nu(\psi)$ is the Morse index (number of negative eigenvalues) of the operator $\Lambda_k \colon B \to \lambda_k(L_B)$, and Λ_k is smooth at its critical point $B \sim 1$.

A direct corollary is that $k - \beta_1 \leq \nu(\psi) \leq k$, where $\nu(\psi)$ is the number of strong nodal domains of ψ. For proofs and further discussion, see [28, 27, 26].

2.3 Spectral Methods in Data Analysis: Spectral Partitions

In this section, we touch on one family of recent approaches for using spectral information in the analysis of data: using eigenfunctions of certain Laplacians to split a domain into two pieces, i.e. a clustering procedure. We start our discussion with a recent application of these ideas: gerrymandering and computational redistricting. This motivates the introduction of ratio cuts of graphs, after which we discuss the continuum analogue of these kinds of cuts.

2.3.1 Congressional redistricting and analogous partitions

Gerrymandering is process by which voter districts are drawn in a way to provide unfair political advantages to one party or another. While gerrymandering has been around since the 19th century, recent advances in computational methods and social awareness have resulted in mathematicians taking an active role in aiding legislatures in the quantification and analysis of gerrymandered regions. For a survey on computational approaches and their uses in gerrymandering cases in North Carolina see [29].

Recent approaches to computational redistricting treat this as a graph partitioning problem: given counties/tracts/blocks in a state, construct a graph that encodes geographic relationships between the counties/tracts/blocks and then split said graph into the desired number of districts. While states have a number of rules for what valid districts can look like, trying to enumerate all possible partitions is still unwieldy. For example, Minnesota is estimated to have over 6×10^{18} possible Senate redistricting plans just by merging House districts; the entire space of plans is estimated to have over 10^{100} possibilities [30]. This shows that direct enumeration is not a viable approach, and smarter methods for exploring the space of partitions are necessary. One of the challenges in computational redistricting is quantifying the various rules about what may or may not constitute valid districts; see [31, §3.2] for a discussion of some common rules. We briefly mention that other approaches to detecting gerrymandering based on the geometry of districts have been proposed and analyzed [32].

One recent approach to exploring the space of redistricting plans is the recombination algorithm, or ReCom for short [31]. Let G be the dual graph for counties/tracts/blocks in a state, where vertices correspond to counties/tracts/blocks, and suppose G is partitioned into k disjoint sets $G_1, G_2, ..., G_k$, which we want to interpret as voter districts. The simplest version of ReCom chooses two sets G_i, G_j, takes the subgraph of G induced by the vertices $G_i \cup G_j$, and splits the induced subgraph into two new sets $\tilde{G}_i, \tilde{G}_j$. One possibility for cutting is via spanning trees: choose a spanning tree T of the induced subgraph and cut T into two disjoint trees $\tilde{G}_i, \tilde{G}_j$ in such a way that the population of each half is approximately equal. In practice if the cut results in two halves with one half's population significantly larger than 50%, then the proposed tree is rejected and a new tree is sampled. In this discussion we will assume that no trees are rejected, while still aiming for as close to a $50 - 50$ population split as possible.

In the rest of this section we focus on more theoretical aspects of this kind of partitioning. Let T be some spanning tree over precincts V, and suppose each vertex $v \in V$ has an associated weight $p(v)$. ReCom seeks two disjoint subtrees T_1, T_2 such that the quantity $\frac{1}{|T_1||T_2|}$ is minimized, where $|T_i| := \sum_{v \in T_i} p(v)$. This ratio ensures that the population of each T_1 and T_2 cannot be too small, forcing each to be roughly the same size. If instead of two subtrees one seeks k disjoint sets to partition T, the objective functional $\sum_{i=1}^{k} \frac{1}{|T_i|}$. A direct computation shows that for $k = 2$ minimizing this last functional over a connected component V_1 is equivalent to minimizing $\frac{1}{|T_1||T_2|}$. We note that this construction is specific to the case of a tree with unweighted edges.

Another generalization of the balanced-cut discussed above is the **ratio cut** of a graph. Let $G = (V, E, w, p)$ be a graph with edge-weight function w and vertex-weight function p. The ratio cut of G is the quantity

$$\min_{E_c \subset E} \frac{\sum_{e \in E_c} w(e)}{|V_1||V_2|} \tag{2.1}$$

where V_1 and V_2 are the two connected components of the graph with the edges E_c removed. For a tree with unweighted edges, E_c only consists of a single edge and so $\sum_{e \in E_c} w(e) = 1$ as analyzed above. The minimizing set E_c is called the cut set. If instead one wants a partition into k subsets, the relevant minimization problem is

$$\min_{V_1, V_2, \ldots, V_k} \sum_{i=1}^{k} \frac{\sum_{e \in E(V_i, V_i^c)} w(e)}{|V_i|},$$

where $E(V_i, V_i^c)$ consists of all edges with one vertex in V_i and one vertex in V_i^c, and $V_1 \cup \cdots \cup V_k = V$ [33]. For $k = 2$, this problem boils down to the ratio cut of a graph.

To get a sense of how ReCom and the ratio cut work in practice, we run the following numerical experiment:

1. sample 1000 points from the rectangle $[0, 1] \times [0, 2]$,

2. construct a graph G on the sampled points,

3. compute the ratio cut of G.

For ReCom we construct a minimal spanning tree (MST) on the sampled points, where the edge-weights are Euclidean distances. MSTs are defined to be spanning trees on the underlying graph for which the sum of edge-weights is minimal, and computationally efficient approaches exist to compute the MST for a given graph [34, 35]. For the ratio cut, we use a self-tuning heat kernel approach to build a similarity matrix using

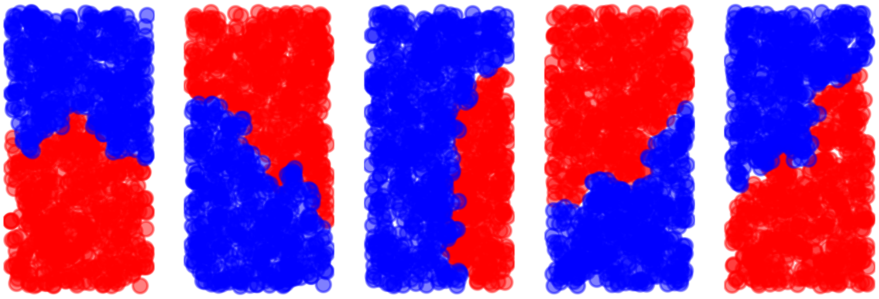

Figure 2.2: The ReCom cuts for 1000 points sampled uniformly from the rectangle $[0, 1] \times [0, 2]$. One of the cuts exhibits an interface that cuts the domain into two nearly 1×1 squares, three of the cuts have interfaces that cut the domain diagonally but still roughly in half, and one of the cuts seems to cut the domain into a quarter and three-quarters.

points' nearest neighbors [36, 37]; more details on this self-tuning construction can be found in Chapter 4.

Figure 2.2 shows the results of ReCom using the MST for five different samples of 1000 points. The leftmost plot gives what we might call a "reasonable" partition, especially when trying to keep the redistricting motivation in mind; the cut results in two domains that are each close to being a 1×1 square. The middle-left and right two plots show more generic behaviour of these ReCom cuts, namely a slanted, bumpy interface between the two domains. The middle plot shows behaviour we want to argue is non-generic, namely that the interface does not intersect the two long axes. Analyzing the behaviour of ReCom with MSTs, particularly in the setting of points sampled from rectangles, is the subject of ongoing work.

If instead of a MST we use a ratio cut to partition these sampled points, we get remarkably regular cuts that all seem to cut the rectangle into two 1×1 squares; see Figure 2.3. The motivation here is that ratio cuts incorporate local-neighborhood information among vertices when computing the cut edges, whereas a MST

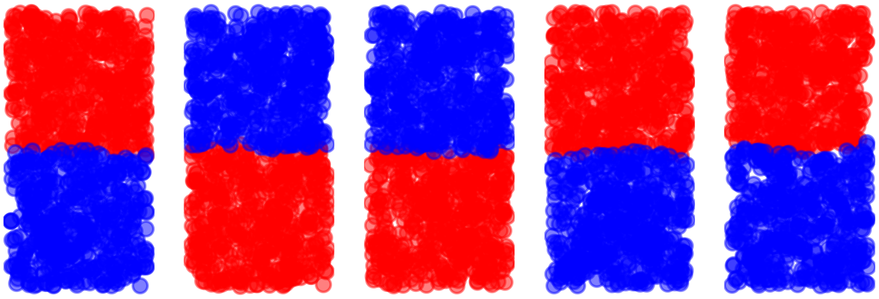

Figure 2.3: The ratio cuts for the same sampled points as in Figure 2.2, in which the graphs are constructed with edges connecting a vertex to its 11 nearest-neighbors. Unlike the ReCom approach, the ratio cut exhibits more regular interfaces that each cut the domain into two 1×1 squares.

does not account for local connections and just relies on a covering property for the graph. The underlying network, as mentioned, has edges between a point and its 11 nearest-neighbors. This construction is useful for numerical reasons: since each row of the network's adjacency matrix has on average 11 non-zero entries, solving the relevant systems of equations to find the ratio cut is computationally more feasible. Moreover, enough edges originating from each vertex are present to preserve notions of local connectivity; i.e. we do not run the risk of having multiple connected components in the sparsified network.

The results of Figure 2.3 suggest that as more points get sampled from $[0, 1] \times [0, 2]$ and the graph "fills in" the domain, we should expect to recover "filled in" squares in the limiting partition. We discuss this limiting behaviour, and its consistency, next.

2.3.2 1-Laplacian partitions

To discuss the limiting behaviour of these graph ratio cuts, we need to know what the right kind of limiting object should look like. Recall that the k-partition ratio cut of a graph G asks for k disjoint subgraphs

$G_1, ..., G_k$ so as to give a solution to $\min_{V_1,...,V_k} \sum_{i=1}^k \frac{\sum_{e \in E(V_i, V_i^c)} w(e)}{|V_i|}$. We can interpret the numerators as lengths of the boundaries between V_i and V_i^c, and the denominators as volumes of the regions V_i. This suggests the following continuum formulation: let $\Omega \subset \mathbb{R}^n$ be an open, bounded domain, and for a subset $X \subset \Omega$ set $\partial_\Omega X = \partial X \cap \Omega$, namely $\partial_\Omega X$ is the boundary of X that sits in the interior of Ω. Then the k-partition ratio cut for the continuum can be read as the solution to

$$\min_{\Omega_1,...,\Omega_k} \sum_{i=1}^{k} \frac{\text{Area}(\partial_\Omega X_i)}{\text{Vol}(X_i)},$$

with the condition that $\Omega = \cup_{i=1}^k \overline{\Omega_i}$; see [38] for a discussion that incorporates non-uniform sampling/measures. Note that when $k = 2$, this problem reduces to

$$\min_{\Omega_1} \frac{\text{Vol}(\Omega)\text{Area}(\partial_\Omega \Omega_1)}{\text{Vol}(\Omega_1)\text{Vol}(\Omega \setminus \Omega_1)}. \tag{2.2}$$

With the target minimization problem identified, we can state an informal version of the main result about consistency of the ratio cut.

Theorem 2.3.1 ([38])**.** *Suppose we have a collection of points $X_n = \{x_1, ..., x_n\}$ sampled uniformly from Ω. Pick a family of scales ϵ_n that tend to 0 with some prescribed rate, a non-negative, non-increasing, $L^2([0,\infty))$ kernel function k_0, and construct a graph G_n on X_n with edge-weights $w_{ij} := k_0\left(\frac{\|x_i - x_j\|}{\epsilon}\right)$. Then as $n \to \infty$, the solutions V_1, V_2 for the problem Equation* (2.1) *on G converge to the solutions $\Omega_1, \Omega \setminus \Omega_1$ for the problem Equation* (2.2).

The ratio cut can be realized in the continuum as the nodal set of the first 1-Laplacian eigenfunction on Ω. The 1-Laplacian is the operator $\Delta^{(1)}$ defined by the variational formulation $\langle f, \Delta^{(1)} f \rangle := \int_\Omega |\nabla f| dx$, and its first eigenvalue is

$$\lambda_1 = \inf_{\int_\Omega f dx = 0} \frac{\int_\Omega |\nabla f| dx}{\int_\Omega |f| dx}.$$

Call the minimizing function f. Then since $\int_\Omega f dx = 0$, f must change sign in Ω and hence have a nodal set Γ. Let $S = \{x \in \Omega \colon f(x) > 0\}$. An alternative characterization of λ_1 shows that the Γ must minimize the ratio cut

$$\frac{\text{Len}(\Gamma)\text{Area}(\Omega)}{\text{Area}(S)\text{Area}(\Omega \setminus S)}$$

([2, §3.1]).

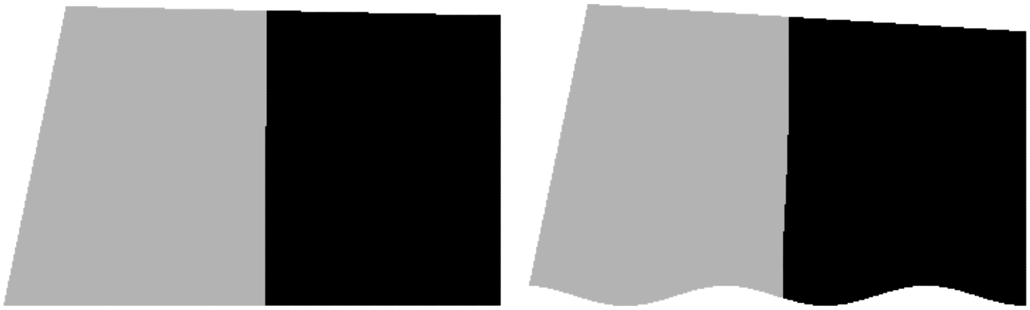

Figure 2.4: Two nearly rectangular domains with optimal cuts indicated. On the left, the cut occurs in a neighborhood of the axis of (near) symmetry of the domain. On the right, even though the boundary of the domain is perturbed in a sinusoidal way the cut occurs in nearly the same spot. Reproduced with permission from [2].

Theorem 2.3.1 established that graph ratio cuts converge to continuum ratio cuts for certain graph constructions, and the last paragraph gave a characterization of the ratio cut as the nodal set of a (1-)Laplace eigenfunction. The last aspect we address in this subsection is the stability of ratio cuts, which can manifest in two ways:

1. if the boundary of the domain Ω is perturbed by a small amount, the geometry of the cut should not significantly change, and

2. if we iteratively cut a domain using ratio cuts, the shapes limit to rectangles away from the boundary.

The first manifestation is illustrated in Figure 2.4. On the left is a domain Ω, and the grey/black regions correspond to the positive/negative nodal domains of a 1-Laplacian eigenfunction. The interface between the two regions is referred to as the cut. On the right is another domain whose boundary is a sinusoidal perturbation of the left domain. Even though the boundary has changed, the nodal domains and the cut seem to remain largely unaffected.

Figure 2.5 illustrates the second manifestation, wherein iterative cuts seem to result in rectangles, at least away from the boundary. Of course in the situation that the domain is a rectangle, the cut will occur in a neighborhood of the axis of symmetry.

The situation in Figure 2.4 can be made precise, in that the stability of the cut can be quantified. This is the content of [2], where they show that domains whose boundaries are close to the boundary of a rectangle, the cut is close to a straight line cut along the rectangle's axis of symmetry. Some specific families of near-rectangles are studied in depth as well, and the dependence of the cut geometry on the parameters for the near-rectangles is made explicit.

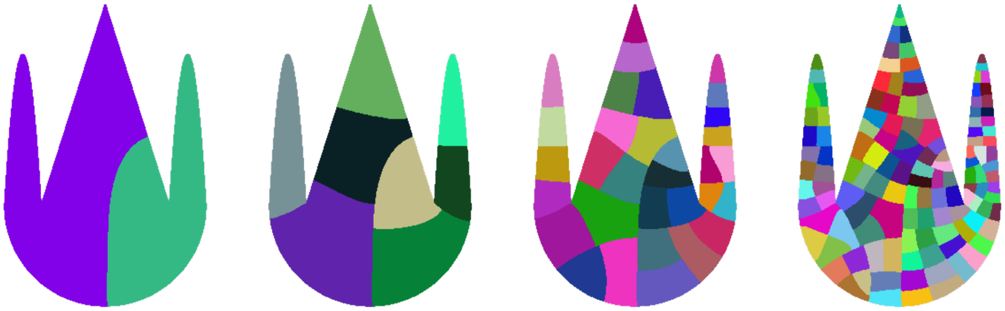

Figure 2.5: On the left, a domain with interesting boundary geometry. From left to right, successive ratio cuts are performed on each of the subdomains. As more ratio cuts are performed, the regions seem to converge to rectangles. Reproduced with permission from [2].

2.4 Spectral Coordinates

Another family of spectral tools for analyzing data are through manifold learning and dimensionality reduction techniques. Here, one wishes to find low-dimensional representations for high-dimensional and messy data that preserve any intrinsic geometric structure. We first provide a case study of community detection using census data, after which we survey the family of techniques known as diffusion maps.

2.4.1 Manifold Cities

The United States census collects data every 10 years through a nationwide questionnaire, which is used to redraw voting districts, review funding allocations, etc. The census also runs the American Community Survey (ACS) program, which collects data from approximately 300,000 addresses monthly and provides a coarse picture of socioeconomic profiles across America in between census years. Other groups and organizations may also collect data about who lives where in cities and other communities. The question we consider here is whether communities of interest can be detected using just socioeconomic data. For example, can one detect where universities are located within a city by just examining socioeconomic data? In some sense, the answer is yes.

In [39], the authors propose a method for transforming census data into a network, and using intrinsic properties of the network to detect notable communities (like student housing regions). Their method is as follows:

1. Collect census data vectors X_i for each socioeconomic unit, like census tract or block, for the area of interest,

2. Normalize each X_i by subtracting its mean, and dividing by its standard deviation,

3. Construct a k-nearest neighbor network where the census units are vertices, and edges connect units if the corresponding vectors X_i are close enough (in the paper, a k-nearest neighbor network construction was used),

4. assign edge-weights such that the edge-weight for edge (i, j) is $k_0(\|X_i - X_j\|)$, where $k_0 \colon [0, \infty) \to [0, \infty)$ is non-increasing and satisfies $k_0(0) = 0$ (in the paper, $k_0(r) = \frac{1}{r}$ was used),

5. From the network's adjacency matrix, construct the normalized graph Laplacian $\mathcal{L} = I - D^{-1/2} A D^{-1/2}$, where A is the weighted adjacency matrix and D is the diagonal matrix with $d_i = \sum_{(i,j)\in E} w_{ij}$,

6. compute the eigenvectors of $\mathcal{L}$ and use these as Euclidean coordinates for the census units.

Once the data has been embedded into (low-dimensional) Euclidean space, standard clustering techniques can be applied to find communities of interest. In [39], the values of each eigenvector were plotted over census blocks for Bristol and student residential areas were identified by inspection. While communities of interest are detectable, there isn't necessarily any systematic way of picking out communities. Here we present first steps towards a more systematic framework of community detection using these techniques.

Baltimore is the 30th most populous city in the United States, home to nearly 600,000 people. In addition Baltimore is home to a number of colleges and universities. Figure 2.6 shows the 199 census tracts that cover Baltimore, and which make up the census units in our replication of [39]. In this study the 5-year ACS estimates of census tracts were used [40], which contain a number of estimated variables including: "Total population, Female" , "Total households, Married-couple family", and "Households with one or more people 65 years and over". For this study we focused on the 467 variables in the ACS5-profile collection.

We convert Baltimore census tract data into a network, with one variation. Instead of choosing $k_0(r) = \frac{1}{r}$, we chose to use a self-tuned heat kernel approach [36], which almost amounts to choosing $k_0(r) = \exp(-r^2/\sigma^2)$ with σ a real parameter. In particular, the edge-weight between tract i and tract j is set as $\exp(\|X_i - X_j\|/(\sigma_i \sigma_j))$, where σ_i is the distance from vector X_i to its kth nearest neighbor. Using such exponentials in constructing edge-weights is popular due to the connection to continuum Laplacians and their heat kernels, as well as their desirable smoothness properties. Given the weighted adjacency matrix we then construct the normalized graph Laplacian and compute the first few eigenvectors.

Figure 2.7 shows the first few eigenvector values for each tract, with red tracts having positive value and blue tracts having negative value. The first eigenvector seems to pick out the central and southern portions

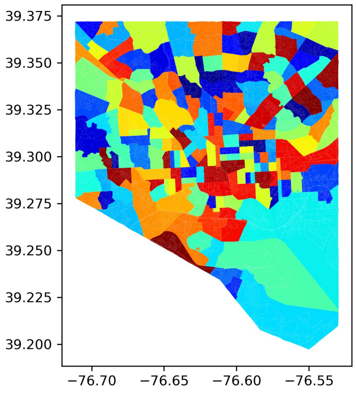

Figure 2.6: The 199 US census tracts of Baltimore city county.

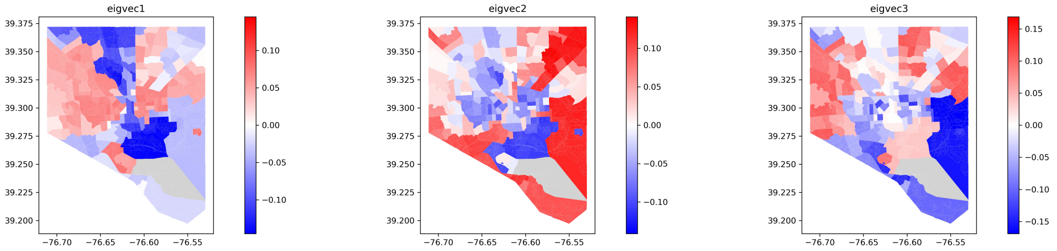

Figure 2.7: The first three eigenvectors of the graph Laplacian for the Baltimore census data. The first eigenvector seems to pick out more affluent regions of Baltimore.

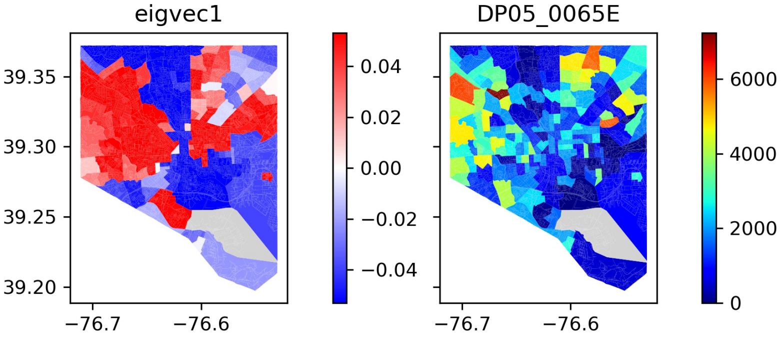

Figure 2.8: The first eigenvector of the census graph Laplacian on Baltimore (left), and the social variable it most strongly correlates with (right). The variable in question is “Total population, Black or African American.”

of Baltimore as one resonant community, and east/west Baltimore as a separate community; the second eigenvector seems to behave similarly, though not as strongly. These observations correspond loosely to local observations: central Baltimore tends to be more affluent as compared to east/west Baltimore. Note that geography was not used to construct the census tract network, just socioeconomic data.

One natural question is what the latent variables are that drive these eigenvectors. Naively we can look for the census variable that most strongly correlates with the first eigenvector, which in this case is DP05_0065E “Total population, Black or African American”; Figure 2.8 This approach is insufficient for two reasons (among others): it ignores the fact that the census variables are not independent, and spurious correlations can arise. In particular, the second eigenvector most strongly correlates with “Occupied units paying rent, $2,000 to $2,499”.

A more robust approach to detecting communities in Baltimore might be to incorporate more eigenvectors and use these values as Euclidean coordinates, on which we can perform standard clustering techniques. In our analysis we used ten eigenvectors, resulting in an embedding of Baltimore’s census tracts into $\mathbb{R}^{10}$. We then performed hierarchical clustering using Ward’s linkage [41] to pick out two, three, and four clusters; the results are displayed in Figure 2.10. Note that in the 2-cluster plot, the grey center of Baltimore is more clearly

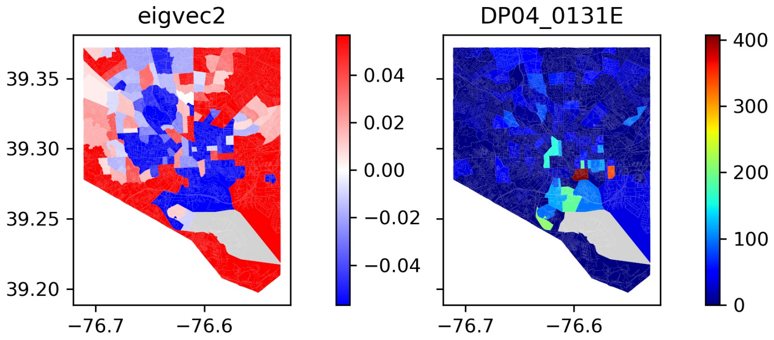

Figure 2.9: The second eigenvector of the census graph Laplacian on Baltimore (left), and the social variable it most strongly correlates with (right). The variable in question is “Occupied units paying rent, $2,000 to $2,499”

distinguished from the green east/west Baltimore. Two regions of interest are downtown Baltimore and the harbour area (center, bottom) and Johns Hopkins University (center). Of note are other communities coloured grey that are disconnected from the grey central region: Morgan State University is the grey region more top-right, the University of Maryland Baltimore County is near the grey region in the lower-left, and Glen neighborhood is included in the grey region in the upper-left. Glen is notable for having a large community of coexisting African-Americans and Orthodox Jews [42].

This small study suggests that using graph Laplacian eigenvectors can be effective for community, and other structure, detection in complex data. We give a more complete treatment of this technique, with

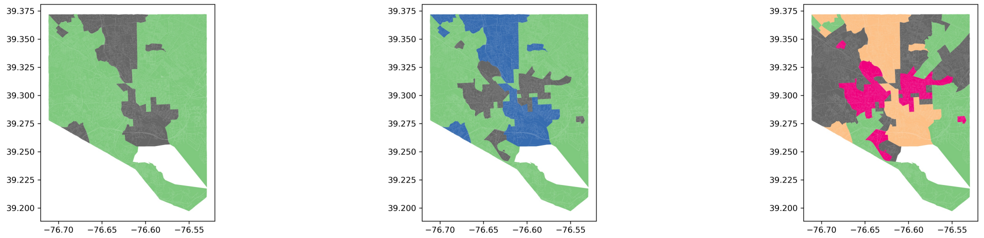

Figure 2.10: The results of using hierarchical clustering with Ward’s linkage to cluster Baltimore’s census tracts.

references, next.

2.4.2 Diffusion maps

Diffusion maps [43, 44] are a collection of techniques that incorporate "spectral information" to find good low-dimensional representations of a data set. While other techniques exist that incorporate information about intrinsic manifold structure [45], we focus on the more spectral techniques in this thesis. These methods work by studying random walks on a network associated to the data, and use related probabilities to construct a distance between data points. Since the introduction of this tool a number of adaptations and generalizations have been proposed for a variety of problems [46, 47, 48, 49].

As mentioned, diffusion maps provide a theoretically robust family of techniques for manifold learning based on a network's graph Laplacian. The idea is that if the data comes from some underlying manifold structure, then a version of the graph Laplacian can be used to approximate a (family of) diffusion process on the manifold. This diffusion process provides a multi-scale geometric view of the data, wherein points of the network are "close" or "highly similar" if there is a high probability of a random walker jumping from one point to another. This then leads to an approximation of the manifold's Laplace-Beltrami operator, from which the eigenvalues and eigenvectors/eigenfunctions can be used to embed data in low-dimensional spaces.

The pipeline for using diffusion maps is as follows: after choosing a parameters $\alpha \in [0, 1]$ and $t \in [0, \infty)$,

1. Construct the graph Laplacian $L = D - A$ for the network,
2. Set $L^{(\alpha)} = D^{-\alpha} L D^{-\alpha}$,
3. Normalize by setting $\mathcal{L} = D^{-\alpha} L^{(\alpha)}$,
4. Compute the eigenvalues/eigenvectors (λ_j, u_j) of $\mathcal{L}$,
5. Set the diffusion coordinates for the ith data object as $\Psi_i = (\lambda_1^t u_{1,i}, \lambda_2^t u_{2,i}, \cdots, \lambda_k^t u_{k,i})$, where k is the dimension of the desired embedding dimension.

By construction the eigenvalues of $\mathcal{L}$ satisfy $1 = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \geq 0$ for a network of N vertices, so the λ_i^t for small i are the largest such quantities. Even though $\mathcal{L}$ is not symmetric, $\mathcal{L}$ is conjugate to a symmetric matrix via $D^{-\alpha/2} \mathcal{L} D^{\alpha/2}$, and so the spectrum is real. Important to note here is that we're not working with the graph Laplacian directly, but rather a random-walk Laplacian associated to the network. This random-walk Laplacian makes it possible to define a metric on the network by saying two points are close if a random walk is more likely to jump from one to another; this notion is made precise next.

Suppose we have a random walk process on the network, where the probability of ending up at vertex y from vertex x in time t is denoted $p_t(y|x)$. As $t \to \infty$, we can expect the random walk to converge to a stationary distribution $\phi_0(y)$, i.e. $p_t(y|x) \to \phi_0(y)$ as $t \to \infty$. We can incorporate this stationary distribution into a metric on the network by setting

$$D_t^2(x_1, x_2) = \|p_t(y|x_1) - p_t(y|x_2)\|_{\phi_0}^2 := \sum_y (p_t(y|x_1) - p_t(y|x_2))^2 \frac{1}{\psi_0(y)}.$$

The weight $\frac{1}{\psi_0}$ encodes local densities of points through the stationary distribution, and the resulting distance is small when the random walk is highly likely to jump between x_1 and x_2 in time t; see [44] for more details.

In practice, let ψ_k be the kth eigenvector of $L^{(\alpha)}$ and define $\Psi_t(x) = (\lambda_1^t \psi_1(x), ..., \lambda_k^t \psi_k(x))$, where x is a vertex of the network and k is the desired dimension of the embedding space. Ψ_t are the diffusion embeddings at scale t, and their utility is in the fact that the Euclidean distance between diffusion embeddings is exactly the diffusion distance between vertices:

Theorem 2.4.1. $D_t^2(x_1, x_2) = \|\Psi_t(x_1) - \Psi_t(x_2)\|^2 = \sum_{j \geq 1} \lambda_j^{2t} (\psi_j(x_1) - \psi_j(x_2))^2$.

The proof is a straightforward computation after rewriting the eigenvectors ψ_j in terms of the random-walk Laplacian $\mathcal{L}$; see [43]. While the parameter α did not explicitly show up in the theorem, α plays an important role in determining what the underlying continuum operator should be. If the network is constructed on a point cloud sampled from a manifold M, then as more points are sampled: when $\alpha = 0$ we recover the Laplace-Beltrami operator for M with a potential term, when $\alpha = \frac{1}{2}$ we recover a Fokker-Planck diffusion process on M, and when $\alpha = 1$ we recover the Laplace-Beltrami operator without a potential. In our work, we primarily work with either $\alpha = \frac{1}{2}$ or $\alpha = 1$.

In practice, not all eigenvectors $\psi_1, ..., \psi_{|V|}$ need to be used since $\lambda_j \to 0$. The user has freedom to use the first k dominant eigenvectors if the embedding dimension is provided, or ask that the diffusion embeddings preserve the diffusion distance up to scale δ. Explicitly, set $s(\delta, t) = \max\{l \in \mathbb{N} \colon |\lambda_l|^t > \delta |\lambda_1|^t\}$. Then taking the first $s(\delta, t)$ eigenvectors ensures the diffusion embedding distances and diffusion distances are within δ of each other [44].

A related manifold learning technique is t-Distributed Stochastic Neighbor (t-SNE), which seeks embedding coordinates for vertices of a network based on an asymmetric random-walk between vertices [50]. The first step of t-SNE essentially constructs (directed) edge-weights by a similar construction to $\mathcal{L}$ from the diffusion map pipeline, and these directed edge-weights are symmetrized to result in an undirected network with edge-weights p_{ij} between vertices i and j such that $p_{ii} = 0$ and $\sum_j p_{ij} = 1$ for all i. Next, Euclidean vectors y_i for each vertex i are determined to minimize the Kullback-Liebler divergence

$$KL(P||Q) := \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}},$$

where $q_{ij} := \frac{\eta(y_i, y_j)}{\sum_k \sum_{l \neq k} \eta(y_k, y_l)^2}$ for an appropriate kernel η. t-SNE is particularly used for data visualization when the vectors y_i are chosen from $\mathbb{R}^2$ or $\mathbb{R}^3$, but it can be incorporated into a statistical pipeline with other tools. Since its introduction, t-SNE has been shown to be effective at a number of clustering and data visualization tasks [51, 52] with some theoretical guarantees [53].

CHAPTER 3

The Graph Spectral Flow

In this chapter we prove two versions of the main theorem, Theorem 3.2.6 and Theorem 3.3.11, which state that the kth eigenvector of a graph's generalized Laplacian $\mathcal{L}$ can have at most k (strong) nodal domains. While the two proofs are similar, there are some nuances arising from different underlying graph structures that we highlight. Section 3.1 introduces the family of generalized graph Laplacians we use and establishes some general properties of their spectra. Section 3.2 proves the graph nodal count theorem using the edge-based construction, while Section 3.3 proves the same theorem using the vertex-based construction; these are all slight generalizations of the results in [3]. Examples and applications are provided in the next chapter.

3.1 Definitions

Suppose $G = (V, E, w)$ is a weighted graph without multiple edges. Vertices will generally be denoted by natural numbers, edges will be 2-tuples of vertices and will be denoted as either (i, j), e_{ij}, or just e, and edge weights will be written $w(e), w((i, j))$, or w_{ij}; $w(e) = 0$ means the edge e is not present in the graph. We only consider graphs with non-negative edge weights. The adjacency matrix of G is the $|V| \times |V|$ matrix $W = (w_{ij})_{(i,j)\in E}$, and the degree matrix is the diagonal $|V| \times |V|$ matrix $D = (\sum_j w_{ij})_{i\in V}$. For more on graph Laplacians and their spectra, see [21] or [54].

The results that appeared in [3] were stated just for the unnormalized graph Laplacian of a graph. In this chapter we work instead with **generalized Laplacians**, which we define next. While the results are the same, working in this framework allows us to better incorporate spectral information into a statistical framework when studying actual data.

Definition 3.1.1. *A **generalized Laplacian** $\mathcal{L}$ on a graph $G = (V, E, w)$ is the $|V| \times |V|$ matrix with $\mathcal{L}_{ij} = -w_{ij}$ whenever $(i, j) \in E$, $\mathcal{L}_{ij} = 0$ whenever $(i, j) \notin E$, and the diagonal terms $\mathcal{L}_{ii}$ are free to take any value.*

*A generalized Laplacian is **diagonally dominant** if the corresponding bilinear form*

$$u^t \mathcal{L} u = \sum_{(i,j)\in E} w_{ij}(d_i u_i - d_j u_j)^2 + \sum_{i\in V} Q_i u_i^2$$

has $Q_i \geq 0$ and $d_i > 0$ for all $i \in V$ for $u \in \mathbb{R}^{|V|}$.

This definition of a generalized Laplacian includes a number of familiar graph Laplacians, including the ordinary graph Laplacian, the Dirichlet graph Laplacian, and the normalized graph Laplacian. The graph Laplacian is easily recovered by setting each $Q_i = 0$ and $d_i = 1$.

Definition 3.1.2. *Given a subset $S \subset V$, the* ***Dirichlet Laplacian*** *corresponding to S, denoted $L^{(S)}$, is the ordinary graph Laplacian L of G with rows and columns corresponding to $V \setminus S$ removed.*

Let E_S be the edges of G with both vertices in S. The Dirichlet Laplacian is diagonally dominant, since

$$u^t L^{(S)} u = \sum_{(i,j)\in E_S} w_{ij}(u_i - u_j)^2 + \sum_{i\in S} Q_i u_i^2$$

with $Q_i = \sum_{j\in V\setminus S} w_{ij}$. Since $Q_i \geq 0$, $L^{(S)}$ is diagonally dominant.

Definition 3.1.3. *The* ***normalized graph Laplacian*** *$L^{(n)}$ is the matrix $L^{(n)} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}AD^{-1/2}$.*

The normalized graph Laplacian is also diagonally dominant, since

$$u^t L^{(S)} u = \sum_{(i,j)\in E} w_{ij} \left(\frac{u_i}{\sqrt{w_i}} - \frac{u_j}{\sqrt{w_j}} \right)^2,$$

where $w_i = \sum_{(i,j)\in E} w_{ij}$.

The coefficients d_i in the expansion $\sum_{(i,j)\in E} w_{ij}(d_i u_i - d_j u_j)^2 + \sum_{i\in V} Q_i u_i^2$ generally appear when considering a generalized Laplacian of the form $T^t L T$, with T invertible. In this case T is interpreted as a change of basis, and we see $u^t T^t L T u = (Tu)^t L (Tu)$, so that the effective bilinear form is still L.

Note that given a diagonally dominant generalized Laplacian, adding a diagonal matrix with positive entries preserves the diagonally dominant property. Sometimes such matrices are referred to as **Schrodinger operators** on graphs; see for example [28].

The first result of this section is a graph analogue for eigenvalues/eigenfunctions of the Laplacian acting on a domain with Dirichlet boundary conditions; compare with Proposition 2.1.3.

Proposition 3.1.4. *Suppose G is connected and $\mathcal{L}$ is a diagonally dominant generalized Laplacian on G. Then the first eigenvector ϕ_0 of $\mathcal{L}$ is signed (ϕ_0 is always everywhere positive or everywhere negative), is nowhere zero, and λ_0 is a simple eigenvalue.*

Proof. The variational characterization for eigenvalues [16, Chapter 1, §10] tells us that ϕ_0 satisfies

$$\phi_0^t \mathcal{L} \phi_0 = \min_{u \in \mathbb{R}^{|V|}, u \neq 0} u^t \mathcal{L} u = \min_{u \in \mathbb{R}^{|V|}, u \neq 0} \sum_{(i,j) \in E} w_{ij}(d_i u_i - d_j u_j)^2 + \sum_{i \in V} Q_i u_i^2.$$

We start by assuming ϕ_0 is nowhere zero. Suppose ϕ_0 changes sign across an edge $e = (i, j)$, and without loss of generality suppose $\phi_{0,i} > 0$ and $\phi_{0,j} < 0$. Then $(d_i \phi_{0,i} - d_j \phi_{0,j})^2 > (d_i \phi_{0,i} - d_j(-\phi_{0,j}))^2$, so by flipping the sign of $\phi_{0,j}$ we further minimize $\phi_0^t \mathcal{L} \phi_0$. Since G is connected, this process can be performed across all sign-change edges to ensure ϕ_0 has the same sign on every vertex.

Now suppose ϕ_0 has a zero, say $\phi_{0,i} = 0$. Then perturbing ϕ_0 at vertex i by a small ϵ will decrease $\phi_0^t \mathcal{L} u$, and so such a minimizer can have no zeros. A straightforward calculation shows that perturbing by $\epsilon = \frac{\sum_{(i,j) \in E_i} d_i d_j u_j}{Q_i + \sum_{(i,j) \in E_i} d_i^2}$ results in such a minimum when considering just perturbations at vertex i.

If λ_0 was not simple, i.e. $\lambda_0 = \lambda_1$, then again by the variational characterization we would have ϕ_1 and ϕ_0 be two orthogonal eigenvectors that are signed on G. This is impossible, and so λ_0 must be simple. □

Similar proofs of this result specific to Dirichlet Laplacians can be found in the literature, e.g. [55] and [22, Lemma 6.1].

We note a straightforward consequence of this last theorem, due to the fact that eigenvectors must be orthogonal:

Corollary 3.1.5. *For G and $\mathcal{L}$ as above, the eigenvectors $\phi_1, \phi_2, ...$ must be signed.*

The rest of this chapter focuses on the nodal domains of a fixed eigenvalue/eigenvector pair (λ_k, ψ), for which we make two assumptions:

Assumption 3.1.6. *In case λ_k has multiplicity greater than 1, k will be the first index for which λ_k appears in the spectrum, i.e. $k = \min\{l \colon L\psi = \lambda_l \psi\}$.*

Assumption 3.1.7. *The coefficients d_i in $u^t \mathcal{L} u = \sum_{(i,j) \in E} w_{ij}(d_i u_i - d_j u_j)^2 + \sum_{i \in V} Q_i u_i^2$ are identically 1.*

Assumption 3.1.8. *The eigenvector ψ is non-zero on each vertex of G. In Section 3.2.3 we discuss what can happen if ψ is zero on some vertices, though this situation turns out to be non-generic; see the introduction of [27] for an extended discussion.*

This first assumption ensures that our bounds are not affected by eigenvalue multiplicities. The second assumption simplifies notation, and ensures we can write $(\mathcal{L}u)_i = \sum_{(i,j) \in E} w_{ij}(u_i - u_j) + Q_i u_i$. The third

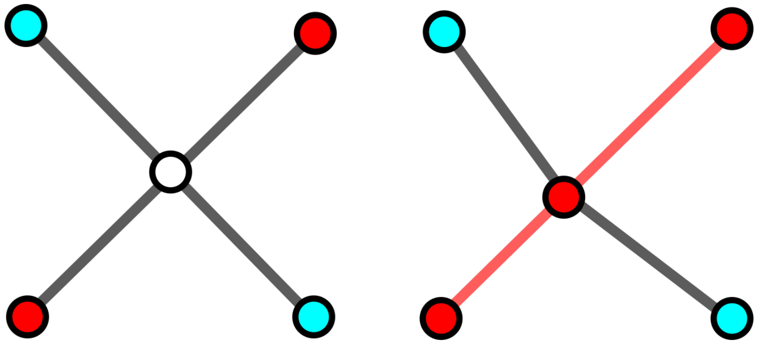

Figure 3.1: On the left, four (strong) nodal domains are present (blue vertices have negative sign, red vertices have positive sign, and white vertices are zero). No matter how the graph is perturbed so that the center vertex acquires a sign, two of the nodal domains will merge to result in three nodal domains in total (as in the right).

assumption greatly simplifies notation and the ensuing arguments, and can always be enforced by (1) perturbing the graph Laplacian by a diagonal matrix Q, or (2) perturbing the edge weights to "shift" a zero off of a vertex. Note that this perturbation may significantly change the number of weak and strong nodal domains: suppose four strong nodal domains/two weak nodal domains meet in an "X" with the center of the "X" a zero vertex. Performing the aforementioned perturbation will cause two of the strong nodal domains to merge, while splitting one of the weak nodal domains into two separate components; see Figure 3.1 for an illustration of this example.

In light of this example, many times we may not want to perturb the graph at all. In these situations we can modify the constructions to allow for the possibility of ψ having zeros, as described in section 3.2.3. We also mention that, by Assumption 3.1.8, the nodal domains we consider are what are called strong nodal domains in the literature [22].

3.2 The Edge-based Flow

In this subsection we replicate the edge-based spectral flow constructed in [3], with the necessary modifications for the diagonally dominant generalized Laplacian framework. Section 3.2.1 contains the main construction, and Section 3.2.2 proves some basic properties of the flow as well as the main theorem Theorem 3.3.11. Section 3.2.3 discusses modifications to account for zero-vertices.

3.2.1 The Construction

The idea for the edge-based flow is to perturb $\mathcal{L}$ along the edges for which an eigenvector ψ changes sign, in such a way that imposes a "zero boundary condition" across said edges. As the strength of the perturbation increases the sign-change edges effectively disappear, and resulting bilinear form feels a graph whose connected components are the nodal domains of ψ.

Definition 3.2.1. *Given an eigenvalue/eigenvector pair (λ_k, ψ) of the generalized Laplacian $\mathcal{L}$, define the sign change edges $E_\pm = \{(i, j) \colon \psi_i \psi_j < 0\}$. For each $(i, j) \in E_\pm$, define the rank-1, $|V| \times |V|$ matrices P_{ij}, in which the 2×2, (i, j) block is $w_{ij} \begin{pmatrix} q_{ji} & 1 \\ 1 & q_{ij} \end{pmatrix}$ with $q_{ij} := -\frac{\psi_i}{\psi_j}$ and all other terms are zero. We define the **edge-based spectral flow** as the collection of eigenvalues associated to the family of bilinear forms*

$$\begin{aligned} B_\sigma(u, v) :=& \langle u, \mathcal{L} v \rangle + \sigma \langle u, \sum_{(i,j) \in E_\pm} P_{ij} v \rangle \\ =& \langle u, \left(\mathcal{L} + \sigma \sum_{(i,j) \in E_\pm} P_{ij} \right) v \rangle. \end{aligned}$$

We set $P = \sum_{(i,j) \in E_\pm} P_{ij}$ and $\mathcal{L}_\sigma = \mathcal{L} + \sigma P$, so that $B_\sigma(u, v) = \langle u, \mathcal{L}_\sigma v \rangle$. The edge-based spectral flow is the curve $(\lambda_1(\sigma), \lambda_2(\sigma), ..., \lambda_{|V|}(\sigma))$ with $\sigma \in [0, 1]$ where each λ_i is an eigenvalue branch of B_σ.

Note that setting $\sigma = 1$ gives $\mathcal{L}_{1,ij} = 0$ for $(i, j) \in E_\pm$, reflecting the notion that the sign-change edges have disappeared from the graph.

3.2.2 Properties of the Flow

We next present a series of lemmas that together constitute a proof of Theorem 3.2.6. As mentioned earlier, these proofs are replicated from [3] and adapted to our generalized Laplacian framework. Recall that (λ_k, ψ) is an eigenvalue/eigenvector pair of $\mathcal{L}$.

Lemma 3.2.2. *The eigenvalue λ_k is in the spectrum of $\mathcal{L}_\sigma$ for $0 \leq \sigma \leq 1$, and $L_\sigma \psi = \lambda_k \psi$. In particular the λ_k eigenvalue branch is constant.*

Proof. Note that ψ is in the kernel of each P_{ij}, since

$$P_{ij} \psi = w_{ij} \begin{pmatrix} q_{ji} \psi_i + \psi_j \\ \psi_i + q_{ij} \psi_j \end{pmatrix} = w_{ij} \begin{pmatrix} -\psi_j + \psi_j \\ \psi_i - \psi_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Thus,

$$\mathcal{L}_\sigma \psi = \mathcal{L} \psi + \sigma P \psi = L \psi = \lambda_k \psi.$$

□

Lemma 3.2.3. *The eigenvalues of B_σ, which are the eigenvalues of the matrix $\mathcal{L}_\sigma$, are non-decreasing eigenvalue branches in σ for $0 < \sigma < 1$.*

Proof. Suppose $(\lambda, u) = (\lambda_\sigma, u_\sigma)$ is an eigenvalue/eigenvector pair of B_σ with $\langle u, u\rangle = 1$, so that

$$B_\sigma(u, v) = \lambda\langle u, v\rangle \quad \forall v \in \mathbb{R}^{|V|}.$$

Each λ_σ is an analytic curve in σ, branching from the eigenvalue/eigenvector pairs of $\mathcal{L}_0 = \mathcal{L}$; this follows from standard perturbation theory [16]. Differentiating with respect to σ gives

$$B_\sigma(u, v)' = B'_\sigma(u, v) + B_\sigma(u', v), \text{ and } (\lambda\langle u, v\rangle)' = \lambda'\langle u, v\rangle + \lambda\langle u', v\rangle.$$

By the variational formulation for eigenvalues we must have $B_\sigma(u, u') = \lambda\langle u, u'\rangle$, and so

$$\lambda'\langle u, u\rangle + \lambda\langle u', u\rangle = B'_\sigma(u, u) + B_\sigma(u', u),$$

which in turn gives

$$\lambda' = B'_\sigma(u, u) = \langle u, \mathcal{L}'_\sigma u\rangle = \langle u, Pu\rangle\,.$$

Now $\langle u, P_{ij}u\rangle = (\sqrt{q_{ji}}u_i + \sqrt{q_{ij}}u_j)^2 \geq 0$, so $\langle u, Pu\rangle \geq 0$ and $\lambda' \geq 0$ as desired. □

Lemma 3.2.4. *Since $\mathcal{L}_1$ corresponds to the graph G with sign-change edges deleted, $\mathcal{L}_1$ consists of $\nu(\psi)$ nodal domains $G_1, G_2, ..., G_{\nu(\psi)}$. Denote this graph by G_ψ, and let $\mathcal{L}_\psi$ be the graph Laplacian of G_ψ with entries $(i, j) \in E_\pm$ set to zero, and diagonal entries unaffected. Note that $\mathcal{L}_\psi = \mathcal{L}_1$. The spectrum $0 < \lambda_1^\psi \leq \lambda_2^\psi \leq \cdots \leq \lambda_{|V|}^\psi$ of $\mathcal{L}_\psi$ consists of:*

1. *$\lambda_1^\psi = \cdots = \lambda_{\nu(\psi)}^\psi = \lambda_k$, and*
2. *$\lambda_{\nu(\psi)+1}^\psi > \lambda_k$.*

Restricting ψ to each G_i gives a signed eigenvector of λ_1^ψ, so the eigenspace of $\mathcal{L}_1$ for λ_1^ψ is the span of $\psi|_{G_1}, ..., \psi|_{G_{\nu(\psi)}}$. Moreover, eigenvectors of higher eigenvalues must be signed on each connected component of G_ψ.

Proof. To see that $\lambda_1^\psi > 0$, suppose $\lambda_1^\psi = 0$ with u an eigenvector. Then $u^t \mathcal{L}_1 u = \lambda_1^\psi u^t u$, which written out is

$$\sum_{(i,j)\in E} w_{ij}(u_i - u_j)^2 + \sum_{i\in V} Q_i u_i^2 + \sum_{(i,j)\in E_\pm} w_{ij}(\sqrt{q_{ji}}u_i + \sqrt{q_{ij}}u_j)^2 = 0.$$

Every term on the left-hand side is non-negative, so equality holds only for the trivial solution $u \equiv 0$. Thus, $\lambda_1^\psi > 0$.

We show $\lambda_1^\psi = \cdots = \lambda_{\nu(\psi)}^\psi = \lambda_k$ by producing $\nu(\psi)$ eigenvectors of $\mathcal{L}_1$, each with eigenvalue λ_k. Proposition 3.1.4 implies that each eigenvector is signed, so any eigenvalue beyond $\lambda_{\nu(\psi)}^\psi$ must be strictly greater than λ_k.

The $\nu(\psi)$ eigenvectors in question are precisely the restrictions of ψ to each nodal domain G_i:

$$\begin{aligned}
(\mathcal{L}_\psi \psi|_{G_i})_j &= \sum_{(j,m)\in E} w_{\psi,jm}\psi_j - \sum_{(j,m)\in E\setminus E_\pm} w_{\psi,jm}\psi_m + Q_j\psi_j \\
&= \sum_{(j,m)\in E_\pm} w_{\psi,jm}\psi_j + \sum_{(j,m)\in E\setminus E_\pm} w_{\psi,jm}(\psi_j - \psi_m) + Q_j\psi_j \\
&= \sum_{(j,m)\in E_\pm} w_{jm}(1 + q_{mj})\psi_j + \sum_{(j,m)\in E\setminus E_\pm} w_{jm}(\psi_j - \psi_m) + Q_j\psi_j \\
&= \sum_{(j,m)\in E_\pm} w_{jm}(\psi_j - \psi_m) + \sum_{(j,m)\in E\setminus E_\pm} w_{jm}(\psi_j - \psi_m) + Q_j\psi_j \\
&= \sum_{(j,m)\in E} w_{jm}(\psi_j - \psi_m) + Q_j\psi_j \\
&= \lambda_k\psi_j,
\end{aligned}$$

In the right-hand side of the first line, the first term is the multiplication of ψ_j with the diagonal entry of $\mathcal{L}_\psi$, the second term is the off diagonal terms collected, and the third term is the potential piece. □

The next limit classifies the behaviour of eigenvalue branches when they cross λ_k. In particular, crossings occur only with positive slope, since otherwise the eigenvalue branch turns out to be in the spectrum of both $\mathcal{L}$ and $\mathcal{L}_1$ and hence constant.

Lemma 3.2.5. *Let (λ_σ, u) be an eigenvalue/eigenvector pair of $\mathcal{L}_\sigma$ for $0 \le \sigma \le 1$, where u depends on σ. If $\lambda'_{\sigma^*} = 0$ for some σ^* then λ_σ is constant and in the spectrum of $\mathcal{L}_1$. Moreover if $\lambda_\sigma = \lambda_k$ then we also have that u is a multiple of ψ.*

Proof. Recall that

$$\lambda' = \langle u, Pu\rangle = \sum_{(i,j)\in E_\pm} w_{ij}(\sqrt{q_{ji}}u_i + \sqrt{q_{ij}}u_j)^2.$$

If $\lambda' = 0$ then $u_i = \frac{\psi_i}{\psi_j}u_j$ for each $(i, j) \in E_\pm$, and $Pu = 0$. But then $\mathcal{L}_{\sigma^*}u = \mathcal{L}u = \lambda_{\sigma^*}u$, so λ_{σ^*} is in the spectrum of $\mathcal{L}$ with u a corresponding eigenvector. Thus λ_σ is constant on the interval $[0, \sigma^*]$, and since these eigenvalue branches are analytic, λ_σ is constant and in the spectrum of $\mathcal{L}_1$.

If moreover $\lambda_{\sigma^*} = \lambda_k$, by Lemma 3.2.4 u is a linear combination of the restrictions $\psi|_{G_k}$ with G_k a nodal domain of ψ. For a fixed G_k, we can find a constant α such that $u|_{G_k} = \alpha\psi|_{G_k}$. The condition $u_i = \frac{\psi_i}{\psi_j}u_j$ for each $(i, j) \in E_\pm$ shows $u|_{G_l} = \alpha\psi|_{G_l}$ whenever G_l and G_k are connected by a sign-change edge. Since G is connected, $u = \alpha\psi$ on all of G. □

Recall that the main result we are interested in is a graph analogue of the continuum spectral flow. Here we restate the main theorem for the edge-based flow and prove it with the string of lemmas from above.

Theorem 3.2.6. *Suppose (λ_k, ψ) is the kth eigenvalue/eigenvector pair of $\mathcal{L}$, λ_k is simple, and that ψ is non-zero at each vertex. Define*

$$B_\sigma(u, v) = \langle u, Lv\rangle + \sigma\langle u, \sum_{(i,j)\in E_\pm} P_{ij}v\rangle$$

where $E_\pm = \{(i, j)\colon \psi_i\psi_j < 0\}$, $P_{ij} = w_{ij}\begin{pmatrix} q_{ji} & 1 \\ 1 & q_{ij}\end{pmatrix}$, and $q_{ij} = -\frac{\psi_i}{\psi_j}$. Then as $\sigma \to 1$,

1. *there are $k - \nu(\psi)$ eigenvalues of B_σ which cross λ_k, and*

2. *the number of eigenvalues of B_σ that converge to λ_k is exactly the number of nodal domains $\nu(\psi)$ of ψ.*

Proof. By Lemma 3.2.3 the eigenvalue branches of L_σ are non-decreasing, and so are either constant or strictly increasing by Lemma 3.2.5. Lemma 3.2.4 tells us that precisely $\nu(\psi)$ eigenvalue branches of L_σ converge to λ_k, so $\delta(\psi) = k - \nu(\psi)$ of the eigenvalues below λ_k will cross λ_k with positive slope and hence converge to eigenvalues strictly greater then λ_k. □

3.2.3 Inclusion of Zero-Vertices

In case the eigenvector ψ does have zeros on G, we can modify the spectral flow construction to give us the correct nodal count. The key idea is to replicate the Dirichlet Laplacian construction from the start, though we show by example that a construction similar to the vertex-based flow of Section 3.3 does not work. The correct variant is described shortly after.

Given (λ_k, ψ), we define the sign change edges $E_\pm = \{(i,j)\colon \psi_i\psi_j < 0\}$ as before but now keep track of the zero edges $E_0 = \{(i,j)\colon \psi_i\psi_j \leq 0\}$. As before we define the rank-1, $|V| \times |V|$ matrices $P_{ij} = w_{ij}\begin{pmatrix} q_{ji} & 1 \\ 1 & q_{ij} \end{pmatrix}$ with $q_{ij} := -\frac{\psi_i}{\psi_j}$ and $(i,j) \in E_\pm$. For each $(i,j) \in E_0$, set $P'_{ij} = w_{ij}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Vertices for which ψ is zero are called zero vertices, and this collection we denote V_0. One candidate for a zero-vertex modified edge-based spectral flow utilizes the bilinear form

$$\begin{aligned} B_\sigma(u,v) :&= \langle u, \mathcal{L}v\rangle + \sigma\langle u, \sum_{(i,j)\in E_\pm} P_{ij}v\rangle + \sigma\langle u, \sum_{(i,j)\in E_0} P'_{ij}v\rangle \\ &= \langle u, \left(\mathcal{L} + \sigma \sum_{(i,j)\in E_\pm} P_{ij} + \sigma \sum_{(i,j)\in E_0} P'_{ij}\right) v\rangle. \end{aligned}$$

We set $P = \sum_{(i,j)\in E_\pm} P_{ij} + \sum_{(i,j)\in E_0} P'_{ij}$ and $\mathcal{L}_\sigma = \mathcal{L} + \sigma P$, so that $B_\sigma(u,v) = \langle u, \mathcal{L}_\sigma v\rangle$. The edge-based spectral flow is the curve $(\lambda_1(\sigma), \lambda_2(\sigma), ..., \lambda_{|V|}(\sigma))$ with $\sigma \in [0,1]$ where each λ_i is an eigenvalue branch of B_σ; compare to Definition 3.3.4.

As in Lemma 3.2.2, this flow satisfies $\langle \psi, \mathcal{L}_\sigma \psi\rangle = \lambda_k$ for all σ. Unlike Lemma 3.2.3, however, this flow is not non-negative. Intuitively, the form B_1 imposes Dirichlet boundary conditions by re-weighting appropriate edges in a way that effectively deletes the sign-change edges from the graph. When zero edges are incorporated, the form B_1 deletes these edges too but does not delete the zero vertex.

As an example of this phenomena, consider the complete graph on three vertices K_3 with the eigenvector $\psi = (1, -1, 0)$. We compute the ordinary graph Laplacian

$$L = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}, P_{12} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, P'_{13} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, P'_{23} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

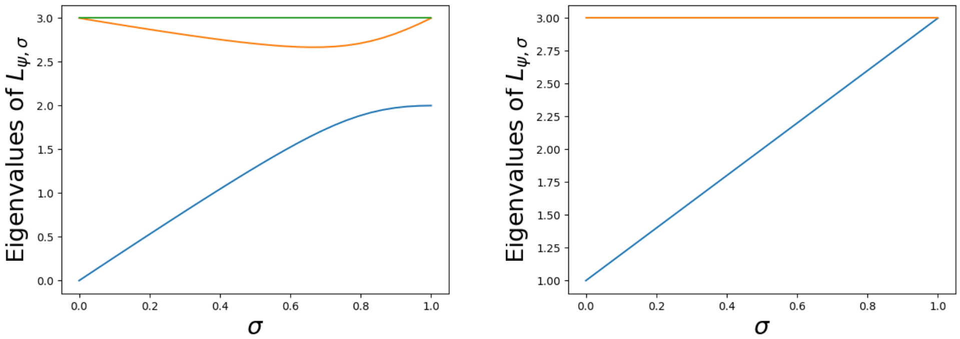

Figure 3.2: The edge-based spectral flows for K_3 and $\psi = (1, -1, 0)$; the flow on the left incorporates the term $\sum_{(i,j)\in E_0} P'_{ij}$, while the flow on the right has the rows and columns corresponding to zeros of ψ deleted from L_σ. Not all branches in the flow on the left are non-decreasing, and hence Lemma 3.2.3 does not hold for this construction.

Putting these pieces together gives

$$L_\sigma = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} + \sigma \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

Figure 3.2 shows the spectral flow for L_σ when the zero vertex is kept in L_σ (left) and when the zero vertex is deleted (right). The spectral flow with a zero vertex is clearly not non-decreasing, and this behaviour can be attributed to the presence of an extraneous connected component of G_ψ corresponding to the zero. Deleting the zero vertex from L_σ results in the correct flow studied in Section 3.2.2.

Though the spectral flow is not non-negative, this new B_1 can still be used to compute nodal domains and nodal deficiencies. In practice, if ψ is zero on vertices of G, we construct $\mathcal{L}_\sigma = \mathcal{L} + \sigma \sum_{(i,j)\in E_\pm} P_{ij}$ and then delete from $\mathcal{L}_\sigma$ the rows and columns corresponding to zero vertices. The multiplicity of λ_k in the spectrum of $\mathcal{L}_1$ is the desired nodal count, as before.

3.3 The Vertex-based Flow

While the edge-based flow can compute nodal deficiencies of Laplacian eigenvectors, it cannot directly incorporate zero vertices from the graph. In this section we provide a second construction of a graph spectral flow called the vertex-based flow which adds "ghost vertices" corresponding to zeros of the eigenvector, and results in the same nodal counts supplied by Theorem 3.2.6. One key use of this framework is in establishing consistency of the graph spectral flow, wherein the addition of ghost vertices provides a clearer bridge to the continuum nodal sets; this is the subject of ongoing work.

3.3.1 The Construction

In this subsection we give the basic construction of a ψ-subdivision graph, in which ghost points are added where the zeros of ψ "should" appear. We also give a means for extending vectors on the original graph, interpreted as functions on the graph, to the ψ-subdivision, and show that ψ is still a λ_k eigenvector of the ψ-subdivision's graph Laplacian.

Definition 3.3.1. *Given an eigenvector ψ of the diagonally dominant generalized Laplacian $\mathcal{L}$ we define*

- *the* ***sign-change edges*** $E_\pm \subset E$ *as those edges* (i, j) *such that* $\psi_i\psi_j < 0$*;*
- *the* ***ghost vertices*** V_{gh}*, defined by adding a vertex* 0_{ij} *to* G *for each* $(i, j) \in E_\pm$*:*

$$V_{gh} = \{0_{ij} \colon (i, j) \in E_\pm\}.$$

The ***ψ-subdivision graph*** $G_{\psi,\sigma}$ *of* G *is the new graph*

$$G_{\psi,\sigma} = (V_\psi, E_\psi, w_{\psi,\sigma}),$$

depending on a parameter $\sigma \in [0, \infty)$*, with*

- $V_\psi := V \cup V_{gh}$*,*
- $E_\psi := E \cup \{(i, 0_{ij}), (0_{ij}, j)\}_{(i,j)\in E_\pm}$*, and*
- $w_{\psi,\sigma}(e) = \begin{cases} w(e), & e \in E \setminus E_\pm, \\ \frac{1}{1+\sigma} w(e), & e \in E_\pm, \\ \frac{\sigma}{1+\sigma} w(\tilde{e})(1 + q_{ji}), & e = (i, 0_{ij}), \tilde{e} = (i, j),\ q_{ji} := \frac{-\psi_j}{\psi_i} > 0. \end{cases}$

Finally, we write $\mathcal{L}_{\psi,\sigma}$ *for the corresponding diagonally dominant generalized Laplacian of* $G_{\psi,\sigma}$*.*

As mentioned, the idea behind this construction is to explicitly incorporate the zeros that should appear across a sign-change edge into the graph itself. The parameter σ determines the strength of the sign-change edges versus the strength of edges that are adjacent to the added zeros, which we refer to as ghost vertices.

We would like to say that ψ is an eigenvector of $\mathcal{L}_{\psi,\sigma}$ as well, though ψ is not defined on $G_{\psi,\sigma}$. The next definition provides a means of extending vectors/functions on G to vectors/functions on $G_{\psi,\sigma}$, which is a "correct" way if we want ψ to extend to an eigenvector; this is the content of Lemma 3.3.3.

Definition 3.3.2. *A vector $f \in \mathbb{R}^{|V|}$, interpreted as a function on G, can be extended to $\tilde{f} \in \mathbb{R}^{|V|+|V_{gh}|}$, interpreted as a function on $G_{\psi,\sigma}$, by setting $\tilde{f}_i = f_i$ for $i \in V$, and $\tilde{f}_{0_{ij}} = a_{ij}f_i + a_{ji}f_j$ for $0_{ij} \in V_{gh}$ with $a_{ij} = \frac{1}{1+q_{ij}}$.*

Lemma 3.3.3. *Suppose (λ_k, ψ) is an eigenvalue/eigenvector pair for the graph G, i.e. $\mathcal{L}\psi = \lambda_k\psi$. Then $\mathcal{L}_{\psi,\sigma}\tilde{\psi} = \lambda_k\tilde{\psi}$ for all σ.*

Proof. This is a straightforward computation. Because ψ is an eigenvector with eigenvalue λ_k, we have

$$(\mathcal{L}\psi)_i = \sum_{(i,j)\in E} w_{ij}(\psi_i - \psi_j) + \sum_{i\in V} Q_i\psi_i = \lambda_k\psi_i.$$

If the vertex i is not in V_{gh}, then

$$\begin{aligned}
(\mathcal{L}_{\psi,\sigma}\tilde{\psi})_i &= \sum_{(i,j)\in E_\psi} w_{ij,\sigma}(\psi_i - \psi_j) + Q_i\psi_i \\
&= \sum_{(i,j)\in E} w_{ij,\sigma}(\psi_i - \psi_j) + \sum_{(i,0_{ij})\in E_\psi\setminus E} \frac{\sigma}{1+\sigma} w_{ij}(1+q_{ji})(\psi_i - \psi_{0_{ij}}) + Q_i\psi_i \\
&= \sum_{(i,j)\in E\setminus E_\pm} w_{ij}(\psi_i - \psi_j) \\
&\quad + \sum_{(i,j)\in E_\pm} w_{ij}\left[\frac{1}{1+\sigma}(\psi_i - \psi_j) + \frac{\sigma}{1+\sigma}(1+q_{ji})\psi_i\right] + Q_i\psi_i \\
&= \sum_{(i,j)\in E\setminus E_\pm} w_{ij}(\psi_i - \psi_j) \\
&\quad + \sum_{(i,j)\in E_\pm} w_{ij}\left[\frac{1}{1+\sigma}(\psi_i - \psi_j) + \frac{\sigma}{1+\sigma}(\psi_i - \psi_j)\right] + Q_i\psi_i \\
&= \sum_{(i,j)\in E} w_{ij}(\psi_i - \psi_j) + Q_i\psi_i = \lambda_k\psi_i = \lambda_k\tilde{\psi}_i.
\end{aligned}$$

Otherwise

$$
\begin{aligned}
(\mathcal{L}_{\psi,\sigma}\tilde{\psi})_{0_{ij}} &= \frac{\sigma}{1+\sigma}w_{ij}(1+q_{ji})(\psi_{0_{ij}}-\psi_i) + \frac{\sigma}{1+\sigma}w_{ij}(1+q_{ij})(\psi_{0_{ij}}-\psi_j) \\
&= \frac{-\sigma w_{ij}}{1+\sigma}((1+q_{ji})\psi_i + (1+q_{ij})\psi_j) \\
&= 0 = \lambda_k\tilde{\psi}_{0_{ij}},
\end{aligned}
$$

and so $\mathcal{L}_{\psi,\sigma}\tilde{\psi} = \lambda_k\tilde{\psi}$. □

The definition of the vertex-based spectral flow is similar to the edge-based flow, with the key differences being that (1) zero/ghost vertices are explicitly incorporated, and (2) $\sigma \to \infty$. The rank-1 perturbations from the edge-based construction are directly incorporated into the ψ-subdivision graph, though one can tease out the edge-based flow from this construction; see Section 3.3.3.

Definition 3.3.4. *Define the family of bilinear forms B_σ on G_ψ by*

$$
B_\sigma(u,v) = \langle u, \mathcal{L}_{\psi,\sigma}v\rangle + \sigma\langle u,v\rangle_{V_{gh}}.
$$

Here, $\langle u,v\rangle_{V_{gh}}$ is the inner product for G_ψ restricted to V_{gh}.

Written out in full,

$$
\begin{aligned}
B_\sigma(u,v) = &\sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i-u_j)(v_i-v_j) + \sum_{i\in V} Q_i u_i v_i \\
&+ \sum_{(i,j)\in E_\pm} w_{ij}\frac{1}{1+\sigma}(u_i-u_j)(v_i-v_j) \\
&+ \sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}\Big[(1+q_{ji})(u_i-u_{0_{ij}})(v_i-v_{0_{ij}}) \\
&\qquad\qquad + (1+q_{ij})(u_j-u_{0_{ij}})(v_j-v_{0_{ij}})\Big] \\
&+ \sigma\sum_{i\in V_{gh}} u_i v_i.
\end{aligned}
$$

3.3.2 Properties of the Flow

The properties of the vertex-based flow are all analogous to the edge-based case. Since the vertex-based flow has a different underlying graph, more can be said and understood about the topology of the nodal sets. As such, the portions of this subsection that are distinct all relate to the inclusion of ghost vertices. Theorem 3.3.11 is the vertex-based analogue of Theorem 3.2.6, and the rest of this subsection builds to its proof.

We start with the vertex-based analogue of Lemma 3.2.3.

Lemma 3.3.5. *The eigenvalues of B_σ are non-decreasing eigenvalue branches of the eigenvalues of $\mathcal{L}_{\psi,0}$, for $0 < \sigma < \infty$.*

Proof. The proof is the same as in the edge-based flow case from Lemma 3.2.3: we have

$$\lambda' = B'_\sigma(u,u) = \langle u, \mathcal{L}'_{\psi,\sigma}u\rangle + \langle u,u\rangle_{V_{gh}},$$

and since $\langle u,u\rangle \geq 0$ we just need $\langle u, \mathcal{L}'_{\psi,\sigma}u\rangle \geq 0$. This is a straightforward computation though:

$$\begin{aligned}
\langle u, \mathcal{L}'_{\psi,\sigma}u\rangle &= \sum_{(i,j)\in E_\psi} w'_{ij,\sigma}(u_i - u_j)^2 \\
&= \sum_{(i,j)\in E} \left(\frac{1}{1+\sigma}\right)' w_{ij}(u_i - u_j)^2 \\
&\quad + \sum_{(i,0_{ij})\in E_\psi\setminus E} \left(\frac{\sigma}{1+\sigma}\right)' w_{ij}(1+q_{ji})(u_i - u_{0_{ij}})^2 \\
&= \sum_{(i,j)\in E_\pm} \left(\frac{1}{1+\sigma}\right)' w_{ij}(u_i - u_j)^2 + \left(\frac{\sigma}{1+\sigma}\right)' w_{ij}(1+q_{ji})(u_i - u_{0_{ij}})^2 \\
&\quad + \left(\frac{\sigma}{1+\sigma}\right)' w_{ij}(1+q_{ij})(u_j - u_{0_{ij}})^2 \\
&= \sum_{(i,j)\in E_\pm} \frac{w_{ij}}{(1+\sigma)^2}\Big(-(u_i - u_j)^2 + (1+q_{ji})(u_i - u_{0_{ij}})^2 \\
&\qquad\qquad + (1+q_{ij})(u_j - u_{0_{ij}})^2\Big) \\
&= \sum_{(i,j)\in E_\pm} \frac{w_{ij}}{(1+\sigma)^2} q_{ij}(u_{0_{ij}} + q_{ji}u_{0_{ij}} - q_{ji}u_i - u_j)^2.
\end{aligned}$$

Since w_{ij}, q_{ij} are both non-negative we conclude $\langle u, \mathcal{L}'_{\psi,\sigma}u\rangle \geq 0$ and so $\lambda' \geq 0$. □

We next introduce the notion of boundary sets for graphs. In [3] this framework is necessary to introduce the Dirichlet Laplacian, which we already have via Proposition 3.1.4. Instead, we use this notion of boundary set to adapt Lemma 3.2.4 into the vertex-based framework. This is necessary since in the edge-based flow, the underlying combinatorial graph for $\mathcal{L}_1$ has $\nu(\psi)$ connected components. In our construction, however, the underlying graph of $\mathcal{L}_{\psi,\infty}$ has a single connected component; this is due to edges connecting the nodal domains to ghost vertices. As such, we need a framework to discuss connected components that intersect at ghost vertices and nowhere else.

Definition 3.3.6. *For a graph $G = (V, E, w)$ and a subset of vertices S, we define:*

- *the **vertex boundary** $\partial_V S$ as the vertices in $V \setminus S$ that are adjacent to some vertex in S, and*
- *the **edge boundary** $\partial_E S$ as the edges in E that connect a vertex in $\partial_V S$ to a vertex in S.*

The space of vectors $u \in \mathbb{R}^{|V|}$ that are zero on $\partial_V S \subset V$ is denoted D^*_S or just D^* when S is clear, i.e.

$$D^* = \{u \in \mathbb{R}^{|V|} : u|_S = 0\}.$$

*Finally, the **Dirichlet subgraph induced by** S, or the D-subgraph induced by S, denoted $S^{(D)}$, is the subgraph of G induced by the vertices in S, together with the vertices of $\partial_V S$ and edges of $\partial_E S$; explicitly, the induced subgraph is $(S \cup \partial_V S, E|_S \cup \partial_E S, w|_{E|_S \cup \partial_E S})$.*

For more on vertex and edge boundaries of graphs, see [21, Chapter 8]

Definition 3.3.7. *Given a graph $G = (V, E)$ and a subset of vertices S, we call the induced D-subgraph of S **Dirichlet disconnected** if there are subgraphs S_1, S_2 of G such that $S^{(D)} = S_1^{(D)} \cup S_2^{(D)}$ and $S_1 \cap S_2 \subset \partial_V S$. Otherwise, S is **Dirichlet connected** if S is not Dirichlet disconnected and both S_1 and S_2 are connected subgraphs of G. We will write this last term as D-connected.*

This next lemma is the vertex-based analogue of the first half of Lemma 3.2.4, and follows quickly from Proposition 3.1.4. Note that this version is stated for a single (D-)connected component, whereas Lemma 3.2.4 was stated for an entire, possibly disconnected, graph.

Lemma 3.3.8. *Suppose that the subgraph $S^{(D)}$ is D-connected with Laplacian $\mathcal{L}_{S^{(D)}}$ induced from the Laplacian $\mathcal{L}$ of G. Then*

1. *the eigenvector ϕ_1 corresponding to $\lambda_1^{(D)}$ is signed,*
2. *$\lambda_1^{(D)}$ is simple, and*

3. *higher index eigenvectors ϕ_i cannot be signed, implying a signed eigenvector must correspond to the first Dirichlet eigenvalue.*

Proof. That $S^{(D)}$ is D-connected tells us that the vertices S are a connected subgraph of G. This lemma then follows directly from Proposition 3.1.4, since

$$u^t \mathcal{L}_{S^{(D)}} u = \sum_{(i,j)\in E|_S} w_{ij}(u_i - u_j)^2 + \sum_{i\in V}\left(\sum_{(i,j)\in\partial_E S} w_{ij}\right)u_i^2 + \sum_{i\in V} Q_i u_i^2$$

is a diagonally dominant generalized Laplacian. □

The following proposition forms the second half of the vertex-based analogue of Lemma 3.2.4, and shows that the λ_k eigenspace of $\mathcal{L}_{\psi_\sigma}$ is spanned by restrictions of ψ to each D-connected component of $G_{\psi,\infty}$.

Proposition 3.3.9. *Given a graph G and a nowhere zero Laplace eigenvector ψ with eigenvalue λ, decompose the nodal domains $S = \{i \colon \psi_i > 0\} \cup \{i \colon \psi_i < 0\}$ of the ψ-subdivision $G_{\psi,\infty}$ into D-connected graphs $S_1, S_2, ..., S_n$. Then the restriction of ψ to each S_l, $\psi|_{S_l}$, is a Dirichlet eigenvector of $S^{(D)}$ with eigenvalue λ. Moreover, $\psi|_{S_l}$ is signed, and so λ is the first Dirichlet eigenvalue for each S_l.*

Proof. Recall that $G_{\psi,\infty}$ contains the original vertices of G together with ghost points 0_{ij} for each $(i, j) \in E_\pm$, and each edge $(i, j) \in E_\pm$ is replaced by two edges $(i, 0_{ij})$ and $(0_{ij}, j)$, with respective edge weights $(1 + q_{ji})w_{ij}$ and $(1 + q_{ij})w_{ij}$.

For a D-connected component S_l, define

$$\psi|_{S_l} = \begin{cases} \psi_i, & i \in S_l, \\ 0, & i \notin S_l, \end{cases}$$

which is the restriction of ψ to S_l, followed by an extension by zero to the rest of the graph. We claim that $\psi|_{S_l}$ is an eigenvector of $L_{\psi,\infty}$ restricted to S_l, which implies that $\psi|_{S_l}$ is also a Dirichlet eigenvector of S_l.

In general, for any vector u that is zero on V_{gh} we have

$$(\mathcal{L}_{\psi,\infty}u)_i = \sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i - u_j) + \sum_{(i,j)\in E_\pm} w_{ij}(1 + q_{ji})(u_i - u_{0_{ij}}) + Q_i u_i$$

For $i \in S_l$,

$$\begin{aligned}
(\mathcal{L}_{\psi,\infty}\psi|_{S_l})_i &= \sum_{(i,j)\in E\setminus E_\pm} w_{ij}((\psi|_{S_l})_i - (\psi|_{S_l})_j) \\
&\quad + \sum_{(i,j)\in E_\pm} w_{ij}(1+q_{ji})((\psi|_{S_l})_i - (\psi|_{S_l})_{0_{ij}}) + Q_i\psi_i \\
&= \sum_{(i,j)\in E\setminus E_\pm} w_{ij}(\psi_i - \psi_j) + \sum_{(i,j)\in E_\pm} w_{ij}(1+q_{ji})\psi_i \; + Q_i\psi_i \\
&= \sum_{(i,j)\in E\setminus E_\pm} w_{ij}(\psi_i - \psi_j) + \sum_{(i,j)\in E_\pm} w_{ij}(\psi_i - \psi_j) + Q_i\psi_i \\
&= \sum_{(i,j)\in E} w_{ij}(\psi_i - \psi_j) + Q_i\psi_i = \lambda\psi_i = \lambda(\psi|_{S_l})_i,
\end{aligned}$$

where the sum over $E_\pm$ can be empty or not depending on if i has neighbors in V_{gh}. This shows each $\psi|_{S_l}$ is a Dirichlet eigenvector of S_l with eigenvalue λ. Moreover, each S_l is a D-connected subgraph of $G_{\psi,\infty}$ corresponding to a nodal domain $\{i\colon \psi_i > 0\}$ or $\{i\colon \psi_i < 0\}$, and so each $\psi|_{S_l}$ is signed.

Thus we have constructed signed Dirichlet eigenvectors for λ on each of the D-connected components of $S^{(D)}$, establishing that λ is the first Dirichlet eigenvalue for each S_l. □

Next up is the vertex-based analogue that eigenvalue branches must cross with positive slope.

Lemma 3.3.10. *If $\lambda'_{\sigma*} = 0$ for some $\sigma* \in (0,\infty)$ then the corresponding eigenvalue branch λ_σ is constant and λ_σ is in the spectrum of $\mathcal{L}_{\psi,\infty}$. Moreover if $\lambda_\sigma = \lambda_k$ then the eigenvector u is a constant multiple of ψ.*

Proof. This proof follows mutatis mutandis as in the proof of Lemma 3.2.5: from $\lambda' = 0$ we have $\langle u, L'_{\psi,\sigma}u\rangle = 0$ and $\langle u, u\rangle_{V_{gh}} = 0$. The latter equality forces $u_{0_{ij}} = 0$ for $(i,j) \in E_\pm$, after with the former imposes $u_i = \frac{\psi_i}{\psi_j}u_j$ across $(i,j) \in E_\pm$:

$$\begin{aligned}
\langle u, L'_{\psi,\sigma}u\rangle &= \sum_{(i,j)\in E_\pm} \frac{w_{ij}}{(1+\sigma)^2} q_{ij}(u_{0_{ij}} + q_{ji}u_{0_{ij}} - q_{ji}u_i - u_j)^2 \\
0 &= \sum_{(i,j)\in E_\pm} \frac{w_{ij}}{(1+\sigma)^2} q_{ij}(-q_{ji}u_i - u_j)^2
\end{aligned}$$

which shows $u_i = \frac{\psi_i}{\psi_j}u_j$, as claimed. □

We are now able to prove the main theorem of this chapter in the vertex-based flow case, i.e. that by adding ghost vertices and appropriate edge weights, we are still able to count the nodal domains of an eigenvector ψ.

Theorem 3.3.11. *As $\sigma \to \infty$, the eigenvalues of B_σ converge to the Dirichlet eigenvalues of the D-subgraph $S^{(D)} = (\{i\colon \psi > 0\} \cup \{i\colon \psi < 0\})^{(D)}$. The number of D-connected components of $S^{(D)}$ is the multiplicity of λ_k for B_∞, and the nodal deficiency of ψ on $G_{\psi,\infty}$ is $\delta(\psi) = k - \nu(\psi)$. Note that, by construction, there will be $k - \nu(\psi) + |V_{gh}|$ eigenvalue branches that cross λ_k as $\sigma \to \infty$.*

Proof. As $\sigma \to \infty$, the eigenvalue branches of B_σ are increasing or constant, and so either cross λ_k with positive slope or are eigenvalues of $\mathcal{L}_{\psi,\infty}$. Since the λ_k eigenspace of $\mathcal{L}_{\psi,\infty}$ has multiplicity $\nu(\psi)$, exactly $\nu(\psi)$ eigenvalue branches of B_σ will converge to λ_k. The remaining $k - \nu(\psi) + |V_{gh}|$ vertices will cross λ_k, and so accounting for the eigenvalue branches originating from ghost vertices we conclude that the nodal deficiency of ψ in G is precisely $\nu(\psi)$. □

As mentioned in the proof, the vertex-based flow will have $|V_{gh}|$ extra eigenvalue branches. If we suppose that $|V_{gh}| > |V|$, then since the branches are non-decreasing and cannot intersect one another we see that all of the eigenvectors in the λ_k eigenspace of $\mathcal{L}_{\psi,\infty}$ stemmed from indicator vectors of ghost vertices, or from the 0-eigenvector of $\mathcal{L}$ (if it exists). This observation suggests that the ghost vertices and their sign-change edges are intimately related with the eigenspace of λ_k, though the details of this connection are still unclear.

Open Problem 3.3.12. *How do the sign-change edges contribute to the nodal domain counts? For each eigenvalue branch converging to λ_*, the corresponding eigenvector will converge to a linear combination of first Dirichlet eigenvectors for each D-connected domain of G_ψ: what do the eigenvectors tell us about the nodal domains, and how does the graph topology determine which sign-change edges give rise to eigenvectors of $L_{\psi,\infty}$?*

In the continuum case, we know which eigenvalue branches will cross λ_k for domains $\Omega = [0, \alpha\pi] \times [0, \pi]$. Indeed, for the Laplacian with separable potential $L = \Delta + q(x) + r(y)$, its eigenvalues $\lambda_{m^*n^*}$ are precisely sums of eigenvalues λ_{m^*} of $\Delta + q(x)$ and λ_{n^*} of $\Delta + r(y)$ acting on $[0, \alpha\pi]$ and $[0, \pi]$ respectively: $\lambda_{m^*n^*} = \lambda_{m^*} + \lambda_{n^*}$. The characterization is as follows:

Theorem 3.3.13 ([12, Theorem 1])**.** *The eigenvalue branch of λ_{mn} crosses $\lambda_{m^*n^*}$ if and only if $\lambda_{mn} \leq \lambda_{m^*n^*}$ and $m > m^*$ or $n > n^*$.*

This theorem has a geometric interpretation as well: if $E_{\lambda_{m^*n^*}} = \{(x, y)\colon x > 0, y > 0, (\frac{x}{\alpha})^2 + y^2 < \lambda_{m^*n^*}\}$ is an open ellipse restricted to the first quadrant, and $R_{\lambda_{m^*n^*}} = \{(x, y) \colon 0 < x \leq m^*, 0 < y \leq n^*\}$ is a half-open rectangle in the same quadrant, then the eigenvalue λ_{mn} crosses $\lambda_{m^*n^*}$ if and only if $\lambda_{m^*n^*} \in E_{\lambda_{m^*n^*}} \setminus R_{\lambda_{m^*n^*}}$.

In general, however, the question of which eigenvalues contribute to crossings, in the continuum and graph settings, remains open.

3.3.3 Relation between the Edge-based and Vertex-based constructions

While the edge-based and vertex-based flows follow different frameworks, they can be related by considering the vertex-based flow for two functions $\tilde{f}, \tilde{g} \colon G_{\psi,\sigma} \to \mathbb{R}$ that are extensions of functions f, g on G. This short subsection makes explicit this relation.

Proposition 3.3.14. *Suppose $\tilde{u}, \tilde{v} \colon G_{\psi,\sigma} \to \mathbb{R}$ are extensions of functions u, v on G. Then*

$$B_\sigma(\tilde{u}, \tilde{v}) = \langle u, \mathcal{L}v \rangle + \sigma \sum_{(i,j)\in E_\pm} \frac{a_{ij}a_{ji}}{w_{ij}} \langle u, P_{ij}v \rangle.$$

Proof. For functions $\tilde{u}$ and $\tilde{v}$ that are extensions of functions u, v on G, we have

$$\tilde{u}_{0_{ij}} = a_{ij}u_i + a_{ji}u_j = \frac{1}{1 + q_{ij}}u_i + \frac{1}{1 + q_{ji}}u_j,$$

so the term $\sigma \sum_{i\in V_{gh}} \tilde{u}_i\tilde{v}_i$ of $B_\sigma(\tilde{u}, \tilde{v})$ becomes

$$\begin{aligned} &\sum_{(i,j)\in E_\pm} a_{ij}^2 u_i v_i + a_{ij}a_{ji}u_i v_j + a_{ij}a_{ji}u_j v_i + a_{ji}^2 u_j v_j \\ = &\sum_{(i,j)\in E_\pm} a_{ij}a_{ji}(\sqrt{q_{ji}}u_i + \sqrt{q_{ji}}u_j)(\sqrt{q_{ji}}v_i + \sqrt{q_{ji}}v_j) \\ = &\sum_{(i,j)\in E_\pm} u^T p_{ij} v. \end{aligned}$$

Here p_{ij} is the matrix with zeros except at the i, j submatrix, taking the form

$$p_{ij} = \begin{pmatrix} a_{ij}^2 & a_{ij}a_{ji} \\ a_{ij}a_{ji} & a_{ji}^2 \end{pmatrix} = a_{ij}a_{ji} \begin{pmatrix} q_{ji} & 1 \\ 1 & q_{ij} \end{pmatrix} = \frac{a_{ij}a_{ji}}{w_{ij}} P_{ij}.$$

We also see that

$$
\begin{aligned}
&\sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}\left[(1+q_{ji})(u_i-u_{0_{ij}})(v_i-v_{0_{ij}})+(1+q_{ij})(u_j-u_{0_{ij}})(v_j-v_{0_{ij}})\right]\\
&=\sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}\Big[(1+q_{ji})((1-a_{ij})u_i-a_{ji}u_j)((1-a_{ij})v_i-a_{ji}v_j)\\
&\quad+(1+q_{ij})((1-a_{ji})u_j-a_{ij}u_i)((1-a_{ji})v_j-a_{ij}v_i)\Big].\\
&=\sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}\left[a_{ji}(u_i-u_j)(v_i-v_j)+a_{ij}(u_j-u_i)(v_j-v_i)\right]\\
&=\sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}(u_i-u_j)(v_i-v_j),
\end{aligned}
$$

since $a_{ij}+a_{ji}=1$ and $\frac{a_{ij}}{1+q_{ji}}=1$. We conclude

$$
\begin{aligned}
B_\sigma(u,v)=&\sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i-u_j)(v_i-v_j)+\sum_{(i,j)\in E_\pm} w_{ij}\frac{1}{1+\sigma}(u_i-u_j)(v_i-v_j)+\sum_{i\in V}Q_iu_iv_i\\
&+\sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}[(1+q_{ji})(u_i-u_{0_{ij}})(v_i-v_{0_{ij}})\\
&\qquad\qquad+(1+q_{ij})(u_j-u_{0_{ij}})(v_j-v_{0_{ij}})]+\sigma\sum_{i\in V_{gh}}u_iv_i\\
=&\sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i-u_j)(v_i-v_j)+\sum_{(i,j)\in E_\pm} w_{ij}\frac{1}{1+\sigma}(u_i-u_j)(v_i-v_j)+\sum_{i\in V}Q_iu_iv_i\\
&+\sum_{(i,j)\in E_\pm} w_{ij}\frac{\sigma}{1+\sigma}(u_i-u_j)(v_i-v_j)+\sigma\sum_{i\in V_{gh}}u_iv_i\\
=&\sum_{(i,j)\in E} w_{ij}(u_i-u_j)(v_i-v_j)+\sum_{i\in V}Q_iu_iv_i\\
&+\sigma\sum_{(i,j)\in E_\pm} a_{ij}a_{ji}(\sqrt{q_{ji}}u_i+\sqrt{q_{ji}}u_j)(\sqrt{q_{ji}}v_i+\sqrt{q_{ji}}v_j)\\
=&\langle u,\mathcal{L}v\rangle+\sigma\sum_{(i,j)\in E_\pm}\frac{a_{ij}a_{ji}}{w_{ij}}\langle u,P_{ij}v\rangle.
\end{aligned}
$$

□

The constant in front of each $\langle u, P_{ij}v\rangle$ determines when effective Dirichlet boundary conditions are imposed across edges $(i,j)\in E_\pm$, so in general we can consider the bilinear form $\langle u, Lv\rangle + \sigma\sum_{(i,j)\in E_\pm} c_{ij}\langle u, P_{ij}v\rangle$. For the choice $c_{ij}=w_{ij}$, we see that when $\sigma=1$ the Laplacian L_1 indicates each edge $(i,j)\in E_\pm$ is no longer present, which is where the Dirichlet boundary conditions come from. If instead we set c_{ij} to be a constant c, then as σ increases the term $\sigma \sum_{(i,j)\in E_\pm} c\langle u, P_{ij}v\rangle$ will impose Dirichlet boundary conditions across each edge in $E_\pm$ independently, before adding the edge back into the graph. Explicitly, when $\sigma = \frac{w_{ij}}{c}$ for $(i, j) \in E_\pm$, we get a Dirichlet boundary condition imposed just on (i, j) and across no other edges. For this reason it's important that the coefficients c_{ij} are adapted to each sign-change edge.

Note as well that if either ψ_i or ψ_j is zero, then the perturbation P_{ij} does not appear in $\mathcal{L}_{\psi,\sigma}$, and so the necessary boundary conditions aren't imposed across the edge (i, j). This shows that incorporating zero-vertices needs to be done with care and via a different construction.

This version of the vertex-based bilinear form requires that $\tilde{u}$ and $\tilde{v}$ are extensions of vectors u and v, which in general may not be the case for eigenvectors of $\mathcal{L}_{\psi,\sigma}$. Nonetheless, as $\sigma \to \infty$ the vertex-based flow forces $u|_{V_{gh}} = v|_{V_{gh}} = 0$. This leads to $0 = u_{0_{ij}} = \frac{1}{1+q_{ij}}u_i + \frac{1}{1+q_{ji}}u_j$, and so $u_i = -\frac{1+q_{ij}}{1+q_{ji}}u_j = \frac{\psi_i}{\psi_j}u_j$ as in Lemma 3.2.4.

CHAPTER 4

Applications to Data

In this section we illustrate the use of nodal domain counts and spectra in analyzing data. The first section contains relatively simple unweighted graphs and their nodal properties, while the second section explores (unweighted) Erdős-Rényi graphs, (unweighted) Stochastic Block Models, and (weighted) random geometric graphs. These examples are meant to build intuition about how intrinsic graph structure affects spectral and nodal counts. The final section sees the nodal count applied to a popular dataset for machine learning tasks, the MNIST handwritten digit data set.

In the first few sections we compute the eigenvalues and eigenvectors using the standard graph Laplacian. Since some authors refer to a graph's spectrum as the spectrum of the graph's adjacency matrix, we fix a definition to be used throughout:

Definition 4.0.1. *The* ***spectrum*** *of a graph G is the spectrum of its standard graph Laplacian $L = D - A$.*

4.1 Simple Graphs

Here we consider a family of "simple" graphs as baseline examples. Of note are complete graphs and interval graphs, which have respectively 2 nodal domains for all eigenvectors, and zero nodal deficiency for all eigenvectors. The interval graphs in particular mimic the Sturm-Liouville theorem (Theorem 2.1.1) from the continuum.

4.1.1 Complete Graphs

Definition 4.1.1. *The* ***complete graph*** *on n vertices, denoted K_n consists of the vertices $\{1, 2, ..., n\}$ and all edges (i, j), $1 \leq i < j \leq n$.*

Proposition 4.1.2. *The spectrum of K_n is $\{0, n, ..., n\}$, with n repeated $n - 1$ times.*

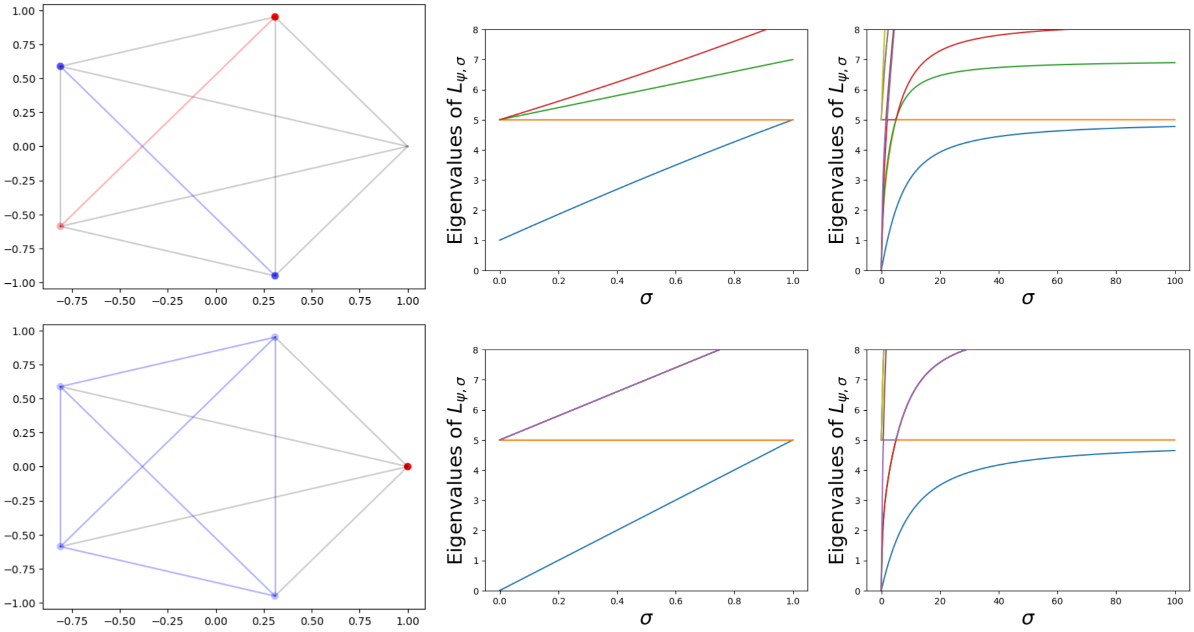

Figure 4.1: The second (top row) and third (bottom row) eigenvector of the graph Laplacian for K_5, along with their edge-based (middle column) and vertex-based (right column) spectral flows. In the edge-based flow for the third eigenvector (middle bottom), only three eigenvalue branches appear; the other two are hidden by the eigenvalue branch above λ_3. Reproduced from [3].

Proof. Note that the graph Laplacian of K_n has structure

$$L = \begin{pmatrix} n-1 & -1 & \cdots & -1 \\ -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & & \vdots \\ -1 & -1 & \cdots & n-1 \end{pmatrix},$$

so 0 is an eigenvalue with eigenvector the constant vector, and a basis for the n-eigenspace is provided by $v_1 = (1, -1, 0, ..., 0)$, $v_2 = (0, 1, -1, 0, ..., 0)$, etc. □

Figure 4.1 shows two eigenvectors (left column) for the complete graph on 5 vertices, K_5, along with the edge-based (center column) and vertex-based (right column) spectral flows. Since each vertex of K_5 is connected to every other vertex, any eigenvector will only have two nodal domains. This is indeed captured by the spectral flows.

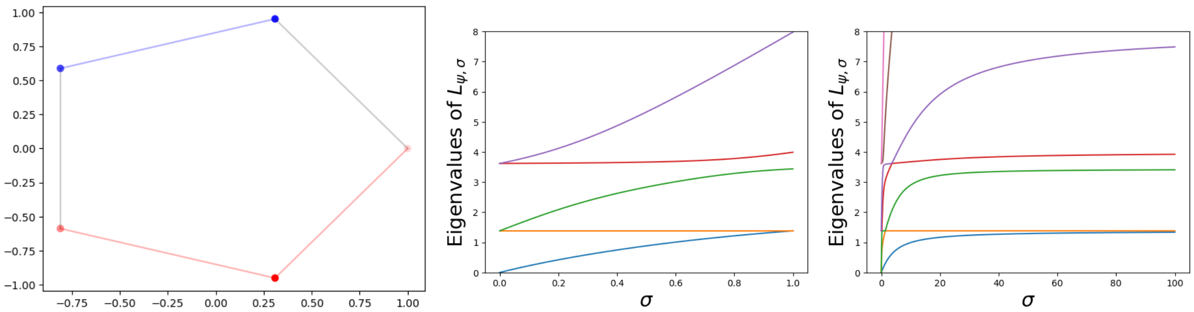

Figure 4.2: The edge-based (middle) and vertex-based (right) spectral flows for the second eigenvector of the graph Laplacian for C_5 (left).

4.1.2 Cyclic Graphs

Definition 4.1.3. *The* ***cyclic graph*** *on n vertices, denoted C_n consists of the vertices $\{1, 2, ..., n\}$ and edges $(i, i + 1)$, $1 \leq i < n$ and $(n, 1)$.*

Proposition 4.1.4. *The spectrum of C_n is $\{2 - 2\cos(\frac{2\pi j}{n})\}_{j=0}^{n-1}$. Accordingly, each eigenvalue has multiplicity 2.*

Proof. The graph Laplacian of C_n takes the form

$$L = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & -1 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{pmatrix}.$$

Let $\zeta = e^{\frac{2\pi}{n}\sqrt{-1}}$ be an nth root of unity. Then $(1, \zeta, \zeta^2, ..., \zeta^{n-1})$ is an eigenvector of L with eigenvalue $2 - \zeta - \zeta^{-1}$, of which we have n such choices of ζ. This procedure explicitly produces n eigenvectors for L [54]. □

Figure 4.2 shows the second eigenvector of the cyclic graph (left) on 5 vertices, C_5, with the edge-based (center) and vertex-based (right) spectral flows. This eigenvector has two nodal domains, with the positive nodal domain having three vertices and the negative nodal domain two, and the spectral flows each have two eigenvalue branches converging to $\lambda_2 = 2 - 2\cos(\frac{6\pi}{5}) \approx 1.3819...$ The edge-based flow does not have any branches that cross λ_2, whereas the vertex-based flow has two crossings. One of these crossings corresponds to a sign-change edge/ghost vertex, whereas the second crossing either corresponds the the other sign-change

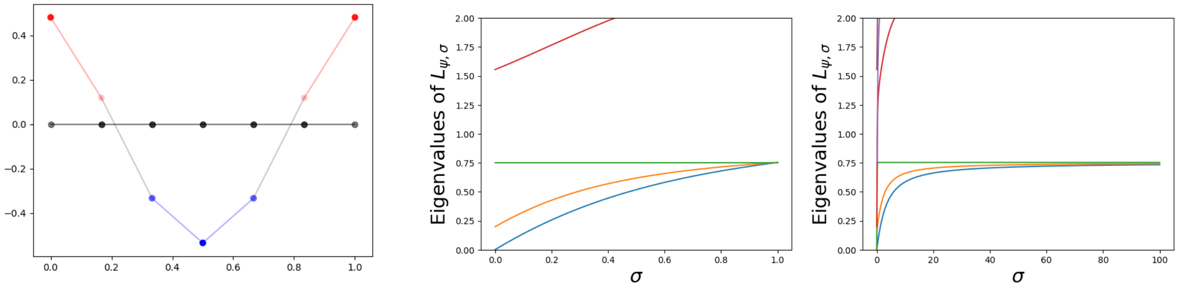

Figure 4.3: The spectral flow for the third eigenvector of the interval graph I_7. We display the eigenvector (left), along with its edge-based (middle) and vertex-based (right) spectral flows. The eigenvector graph shows three nodal domains, and each of the spectral flows have three eigenvalue branches converging to $\lambda_3 = 2(1 - \cos(\frac{2\pi}{7})) = 0.753$.

edge, or the 0-eigenvector of the original graph Laplacian.

4.1.3 Interval Graphs

Definition 4.1.5. *The* ***1D interval graph****, or just* ***interval graph****, on n vertices, denoted I_n consists of the vertices $\{1, 2, ..., n\}$ and edges $(i, i+1)$, $1 \le i < n$.*

Proposition 4.1.6. *The spectrum of I_n is $\{2 - 2\cos(\frac{j\pi}{n})\}_{j=0}^{n-1}$.*

Note that I_n and C_n have the same spectrum. This is not a coincidence, and the proof relies on constructing eigenvectors for I_n through eigenvectors of C_{2n+2}.

Proof. The graph Laplacian of C_n takes the form

$$L = \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

The trick is to consider the spectrum of C_{2n+2}, given in Proposition 4.1.4, and for each eigenvalue $2 - \zeta - \zeta^{-1}$ consider the pair of eigenvector $(1, \zeta, ...) + (1, \zeta^{-1}, ...)$. This eigenvector will have two zeros distance $n + 1$ apart, and so can be restricted to a $2 - \zeta - \zeta^{-1}$ eigenvector on I_n. □

See [54] for a slightly expanded discussion.

In Figure 4.3 we see the third eigenvector (left) of I_7, along with its edge-based (center) and vertex-based (right) spectral flows. Again, the flows each have three eigenvalue branches converging to $\lambda_3 = 0.753...$

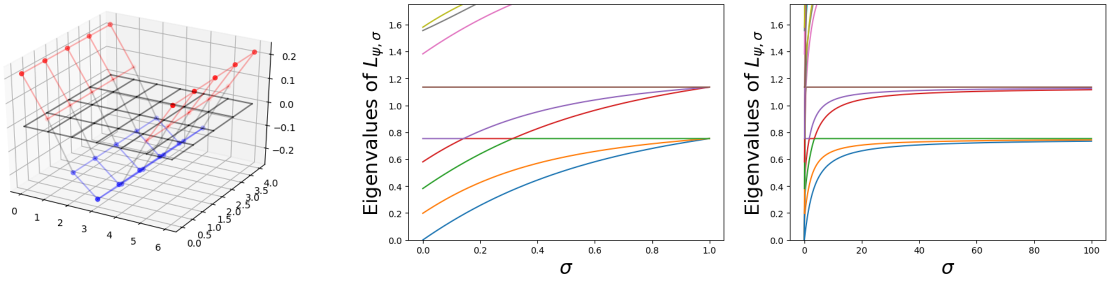

Figure 4.4: The spectral flow for the fifth eigenvector of the interval graph $I_{7,5}$. The eigenvector is displayed (left) along with the edge-based (middle) and vertex-based (right) spectral flows. This eigenvector has three nodal domains, and both the edge- and vertex-based flows have 3 eigenvalues less than or equal to λ_5 in the limit.

Note that in the vertex-based flow, one of the eigenvalue branches that converges to λ_3 corresponded to the 0 eigenvector of L, while the other two branches corresponded to sign-change edges/ghost vertices.

Definition 4.1.7. *The **2D interval graph** on n, m vertices, denoted $I_{n,m}$ consists of the nm vertices*

$$\{v_{1,1}, ..., v_{1,m}, v_{2,1}, ..., v_{n,m}\},$$

and edges of the form $(v_{i,j}, v_{i+1,j})$ and $(v_{i,j}, v_{i,j+1})$ for $1 \leq i < n$ and $1 \leq j < m$.

Proposition 4.1.8. *For the spectrum of $I_{n,m}$, we can take two eigenvectors ϕ_k, ψ_j of I_k, I_j, with corresponding eigenvalues λ_k, λ'_j, and define a Laplace eigenvector $\phi_k \otimes \psi_j$ on $I_{n,m}$ with eigenvalue $\lambda_k + \lambda'_j$.*

Proof. The proof follows by an observation that the graph Laplacian of $I_{n,m}$ can be constructed by taking a Kronecker product of the graph Laplacian for I_n and I_m with appropriate identity matrices. Explicitly, let Id_n be the $n \times n$ identity matrix, and L_n the graph Laplacian for the interval graph I_n. Then the graph Laplacian for the 2D interval graph $I_{n,m}$ is $L_n \otimes Id_m + Id_n \otimes L_m$. Then this Laplacian has eigenvectors $\phi_k \otimes \psi_j$, of which there are nm of them, explicitly producing the required nm orthogonal eigenvectors. The corresponding eigenvalues are $\lambda_i + \lambda'_j$. □

See [54] for more details, where the spectrum is computed for the (slightly) more general Cartesian product of two graphs.

Figure 4.4 shows the fifth eigenvector of $I_{7,5}$ (left), corresponding to the eigenvector of I_5 in Figure 4.3. While there are again three nodal domains for the eigenvector, the distribution of eigenvalues is more complicated and hence both the edge-based (center) and vertex-based (right) spectral flows have eigenvalue branches that cross λ_5. Nonetheless, three eigenvalue branches converge to λ_5 in the limit, matching our observation that the eigenvector has three nodal domains.

4.1.4 Petersen Graphs

Definition 4.1.9. *The **generalized Petersen graph** $GP(n,m)$ for $n \geq 3$ and $1 \leq m \leq \lfloor\frac{n-1}{2}\rfloor$ consists of $2n$ vertices $\{a_0, ..., a_{n-1}, b_0, ..., b_{n-1}\}$, with edges of the form $(a_i, a_{i+1}), (a_i, b_i)$, and (b_i, b_{i+m}) for $0 \leq i \leq n-1$, where the sums are considered modulo n.*

The familiar Petersen graph is $GP(5,2)$.

Proposition 4.1.10 ([56, Theorem 2.4])**.** *The spectrum of the adjacency matrix for $GP(n,m)$ consists of eigenvalues $\delta_{2j}, \delta_{2j+1}$ that are roots of the quadratic equations*

$$\delta^2 - (\alpha_j + \beta_j)\delta + \alpha_j\beta_j - 1 = 0$$

where $\alpha_j = 2\cos(\frac{2\pi j}{n}), \beta_j = 2\cos(\frac{2\pi jm}{n})$. The eigenvalues for the graph Laplacian L of $GP(n,m)$ are thus $3 - \tilde{\lambda}$ for each eigenvalue $\tilde{\lambda}$ of the adjacency matrix.

Proof. For a full proof of the spectrum of the adjacency matrix, see [56]. We comment that the proof utilizes the circulant structure of the adjacency matrix quite explicitly, and seeks eigenvectors on the graph constructed from eigenvectors of the two C_n subgraphs.

That the graph Laplacian spectrum consists of $3 - \tilde{\lambda}$ for each eigenvalue $\tilde{\lambda}$ of the adjacency matrix follows from the 3-regularity of $GP(n,m)$. Namely if A is the adjacency matrix, then $L = 3 * Id - A$. Hence if v is a $\tilde{\lambda}$ eigenvector of A, we get that v is a $3 - \tilde{\lambda}$ eigenvector of L, as desired. □

We can also explicitly write out what the aforementioned roots/eigenvalues are:

Corollary 4.1.11 ([56, Corollary 2.5])**.** *The spectrum of $GP(n,m)$ consists of eigenvalues*

$$3 - \cos\left(\frac{2\pi j}{n}\right) - \cos\left(\frac{2\pi jm}{n}\right) \mp \sqrt{\left(\cos\left(\frac{2\pi j}{n}\right) - \cos\left(\frac{2\pi jm}{n}\right)^2\right) + 1},$$

for $0 \leq j \leq n-1$.

Figure 4.5 shows two eigenvectors for $GP(7,3)$ (left column), along with their edge-based (middle column) and vertex-based (right column) spectral flows. The top row displays results for the 7th eigenvector,

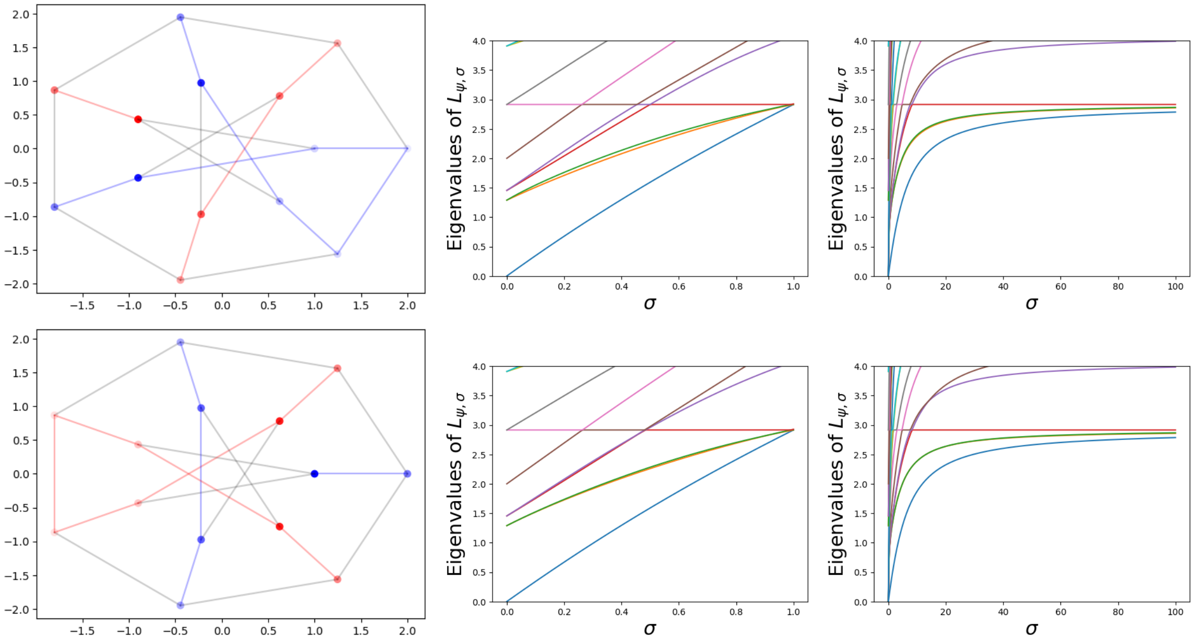

Figure 4.5: The 7th (top left) and 8th (bottom left) eigenvectors for the graph Laplacian of $GP(7, 3)$, with their edge-based (middle column) and vertex-based (right column) flows. Note that, by fig. 4.7 (top right), the corresponding eigenvalues are equal.

and the bottom row displays results for the 8th eigenvector. By inspection both eigenvectors have three nodal domains. Looking at the two spectral flows, however, suggests that in the limit we have four eigenvalue branches converging to the given eigenvalue. In this case, it turns out to that the last crossings occur close to $\sigma = 1$ in the edge-based flow, or much further along in the vertex-based flow. Indeed, Figure 4.6 shows that final crossings occur close to $\sigma = 0.989$ and $\sigma = 600$ in the edge-based and vertex-based flows respectively. This suggests that care must be taken when computing nodal counts, since numerical imprecision and loose tolerances can lead to miscounting the spectrum, and hence miscounting the number of nodal domains.

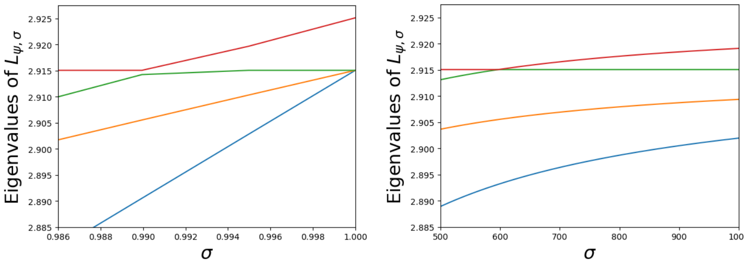

Figure 4.6: The crossing in the edge-based and vertex-based flow of the 7th eigenvector of $GP(7, 3)$. In the edge-based flow the crossing occurs near $\sigma = 0.990$, while in the vertex-based flow the crossing occurs near $\sigma = 600$. The edge-based flow indicates the final nodal count is 3, not 4 as when examined from afar.

4.1.5 Nodal counts

In this short subsection we introduce another statistic to be used extensively when looking at actual data. For each eigenvalue/eigenvalue vector pair (λ_k, ψ), we can compute $\nu(\psi)$ by counting the multiplicity of λ_k in the spectrum of the matrix $L + P$ from Theorem 3.2.6. Collecting each pair $(k, \nu(\psi))$ gives us a summary of the nodal domains across all eigenvectors, though these pairs depend on the choice of eigenbasis for the λ_k eigenspace in the case of repeated eigenvalues. Weighted graphs constructed from point clouds generically do not have repeated eigenvalues, so in practice this dependence on the eigenbasis does not affect our results. In our experiments each eigenvector is normalized, and we have chosen to take the eigenvectors "as-is" directly from the numerical eigenvalue solvers. The NumPy package in Python was used in all of the numerical examples in this thesis, which uses LAPACK [57].

In Figure 4.7 we show the nodal counts for each eigenvector of the cyclic graph C_{31} (top left), $GP(7,3)$ (top right), I_7 (bottom left), and $I_{7,5}$ (bottom right). Each point in the plots corresponds to a pair $(k, \nu(\psi))$, and two points are connected by a solid black line if the corresponding eigenvalues are equal. The red line indicates the upper bound $\nu(\psi) \leq k$ from Theorem 3.2.6 and Theorem 3.3.11. Of particular note are the nodal count for I_7, in which the nodal deficiency is identically 0, and the nodal count for $I_{7,5}$, which consists of sums of the eigenvalues for I_7 and I_5; the 1D interval graphs have nodal counts that conform to the Sturm-Liouville theorem theorem 2.1.1.

4.2 Random Graphs

In this section we go still work with a number of unweighted graphs, which now have some stochastic component to them. The first two families of graphs we explore are the Erdős-Rényi graphs and Stochastic Block Models, after which we explore some geometric graphs constructed on points sampled from $\mathbb{R}^n$.

4.2.1 Erdős-Rényi Graphs

Definition 4.2.1. *An* ***Erdős-Rényi graph*** *$ER(n, p)$ on n vertices with probability p is a graph in which each edge (i, j) exists with probability p.*

A related family of graphs also sometimes referred to as Erdős-Rényi graphs, denoted $ER(n, M)$, consist of n vertices and M total edges. The stochasticity arises in how the graph is chosen: we sample from all possible graphs with n vertices and M edges uniformly. In our numerical examples we utilize the model given in Definition 4.2.1. This family of graphs has been studied since at least the mid-1900s, and much can be said about their asymptotic behaviour (such as properties of the connected components) as $n \to \infty$

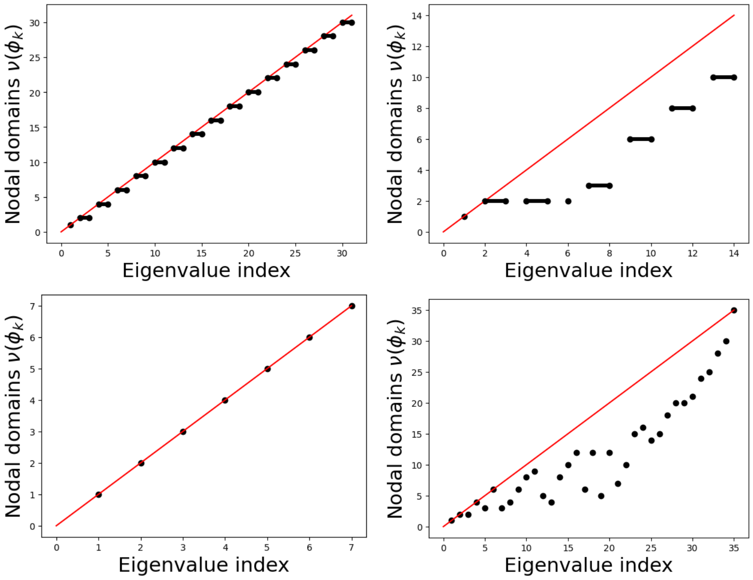

Figure 4.7: For the graphs C_{31} (top left), $GP(7,3)$ (top right), I_7 (bottom left), $I_{7,5}$ (bottom right), plotted points correspond to pairs $(k, \nu(\psi))$ for a λ_k eigenvector ψ of the graph's Laplacian, and black dots are connected by a line if they correspond to the same eigenvalue. The red line is the curve $y = x$; an eigenvector's nodal deficiency is the vertical distance between the corresponding dot and the red line.

[58, 59]. See also [60] for a textbook treatment of these graphs.

Figure 4.8 shows a single $ER(20, p)$ graph for $p = 0.1, 0.3, 0.5, 0.7, 0.9, 0.95$. In case a disconnected ER graph was sampled, we rejected it and sampled again. The nodal counts are displayed directly below each $ER(20, p)$ graph, and suggest that as the probability that an edge is included tends to 1, the ER graph tends towards a complete graph. Likewise, for small p we expect the graph to resemble a tree, and the nodal counts suggest the sampled ER graph is closer to an interval than a complete graph.

4.2.2 Stochastic Block Models

The Erdős-Rényi construction gives one family of graphs that can be informative in reasoning about the "average behaviour" of graphs. One aspect not captured by this family is the possibility of community structure, namely groups of vertices that have many edges within the groups and few edges between vertices of different groups. This model is known as the Stochastic Block Model (SBM) [61], which we discuss next.

Definition 4.2.2. *Given n vertices and a choice of k communities, split the vertices into k clusters of possible variable sizes. For indices $1 \leq i \leq j \leq k$, choose a probability p_{ij} that encodes the likelihood of an edge joining vertices of groups i and j.*

Generally we set each p_{ii} close to 1, and p_{ij} for $i \neq j$ smaller than 1. Note that the SBM agrees with the

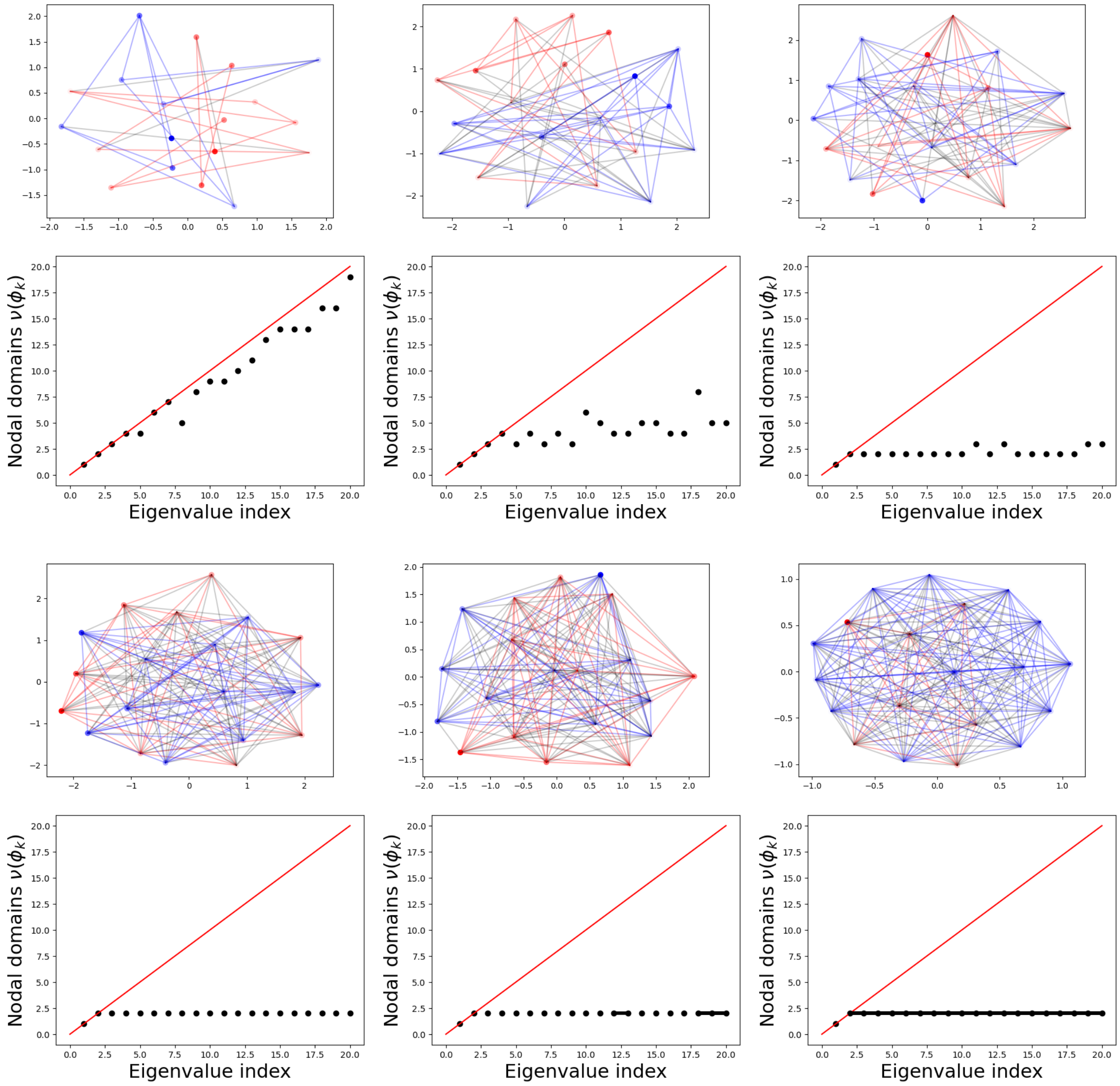

Figure 4.8: Various Erdős-Rényi random graphs on 20 vertices with their nodal domain counts; the graphs correspond to edge probabilities (left to right) $p = 0.1, 0.3, 0.5$ in the top two rows and $p = 0.7, 0.9, 0.95$ in the bottom two rows. As the probability that an edge connects vertices increases, the spectral flow is able to detect that they are closer to being a complete graph than an interval.

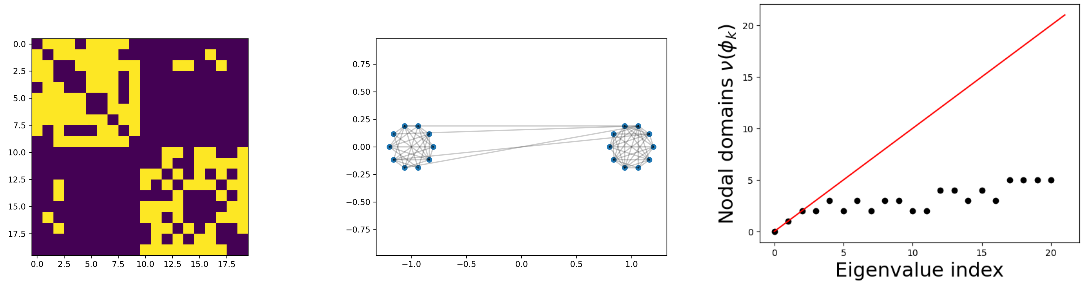

Figure 4.9: An example of a 2-community SBM, with $p_{ii} = 0.7$ and $p_{ij} = 0.1$ for $i \neq j$. The adjacency matrix (left) is displayed with yellow pixels indicating an entry of 1, and purple pixels indicating 0. Two community structures are apparent, corresponding to groups of vertices that have a high probability of being connected. The corresponding graph is also displayed (middle); the two circular groups of vertices correspond to the communities and have more edges among vertices in the same community than to vertices in the other community. The nodal count for each eigenvector is also display (right).

Erdős-Rényi model when $n = k$, or $k = 1$, but for $n > k$ an SBM graph may have a quite different structure.

Two communities We start by analyzing a 2-community SBM, for which the within-community edge probability is $p_{ii} = 0.7$ and the between-community edge probability is $p_{ij} = 0.1$, $i \neq j$. Figure 4.9 shows the adjacency matrix and graph for one sampled SBM graph. In the adjacency matrix plot, a yellow pixel in the (i, j) entry indicates an edge between vertices i and j, whereas a purple pixel indicates the edge (i, j) is not present. The 2-community structure is apparent, corresponding to the two square regions with many yellow pixels. The two squares with primarily purple pixels correspond to the between-community edges. The graph is plotted on the right: the two circles of vertices correspond to the two communities in the SBM, and we can see explicitly the many edges between vertices of the same circles versus the relatively few edges between vertices of different communities.

Three communities We construct a 3-community SBM similar to the 2-community model above: the within-community edge probability is $p_{ii} = 0.7$ and the between-community edge probability is $p_{ij} = 0.1$, $i \neq j$. Figure 4.10 shows the adjacency matrix and graph corresponding to a single sample of a 3-community SBM. The three communities are visible in the adjacency matrix plot as the three square regions of primarily yellow pixels.

Five communities Finally, we construct two 5-community SBMs: the first has a macroscopic community structure of a complete graph on 5 vertices, while the second has a macroscopic community structure of a cyclic graph on 5 vertices. Since the cyclic and complete graph on 3 vertices coincide, these two 5-community

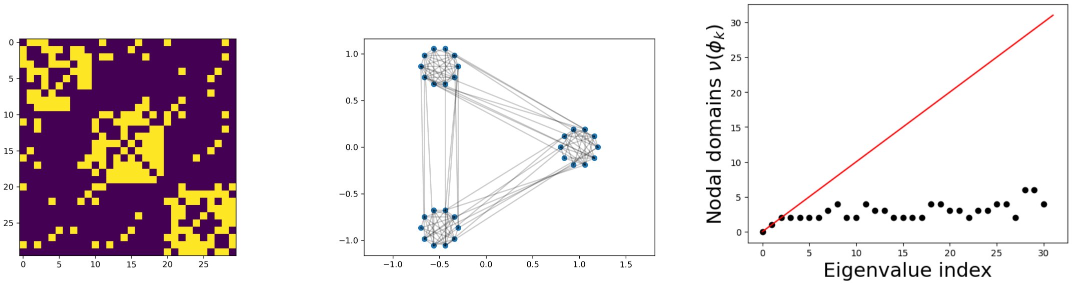

Figure 4.10: An example of a 3-community SBM, with $p_{ii} = 0.7$ and $p_{ij} = 0.1$ for $i \neq j$. Similar to Figure 4.9, the adjacency matrix (left) is displayed with yellow pixels indicating an entry of 1, and purple pixels indicating 0. Three community structures are apparent, corresponding to groups of vertices that have a high probability of being connected. The corresponding graph is also displayed (middle); the three circular groups correspond to the communities and have more edges among vertices in the same community than to vertices in the other communities. The nodal count for each eigenvector is also display (right).

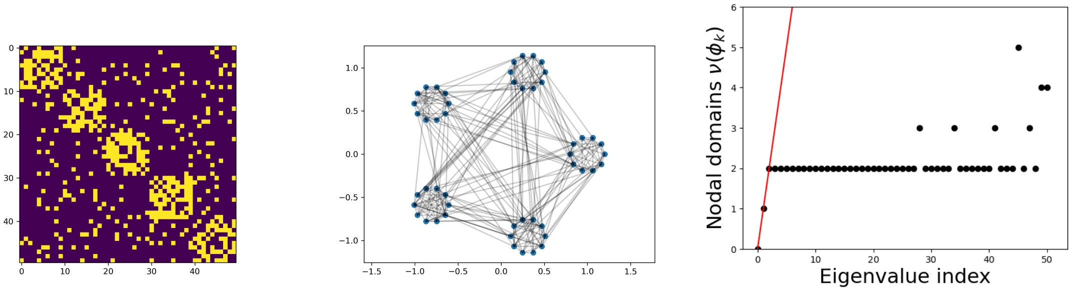

Figure 4.11: An example of a 5-community SBM, with $p_{ii} = 0.7$ and $p_{ij} = 0.1$ for $i \neq j$. Similar to Figure 4.9, the adjacency matrix (left) is displayed with yellow pixels indicating an entry of 1, and purple pixels indicating 0. Five community structures are apparent, corresponding to groups of vertices that have a high probability of being connected. The corresponding graph is also displayed (middle); the five circular groups correspond to the communities and have more edges among vertices in the same community than to vertices in the other communities. The nodal count for each eigenvector is also display (right).

constructions can be thought of as valid generalizations of the 3-community SBM considered above.

For the generalization with a complete graph macroscopic structure, we set the within-community edge probability to be $p_{ii} = 0.7$ and the between-community edge probability to be $p_{ij} = 0.1$, $i \neq j$. Figure 4.11 shows the adjacency matrix and graph corresponding to a single sample of a 5-community SBM. The five communities are visible in the adjacency matrix plot as the five square regions of primarily yellow pixels. Note that in the purple regions of the adjacency matrix there are still significant amounts of yellow pixels, corresponding to the fact that vertices are able to be connected to vertices in any community.

For the generalization with a cyclic graph macroscopic structure, we set the within-community edge probability to be $p_{ii} = 0.7$ and the between-community edge probabilities to be $p_{ij} = 0$ if $j \neq i + 1$ or $(i, j) \neq (1, 5)$, and $p_{ij} = 0.1$ otherwise. Figure 4.12 shows the adjacency matrix and graph corresponding to a

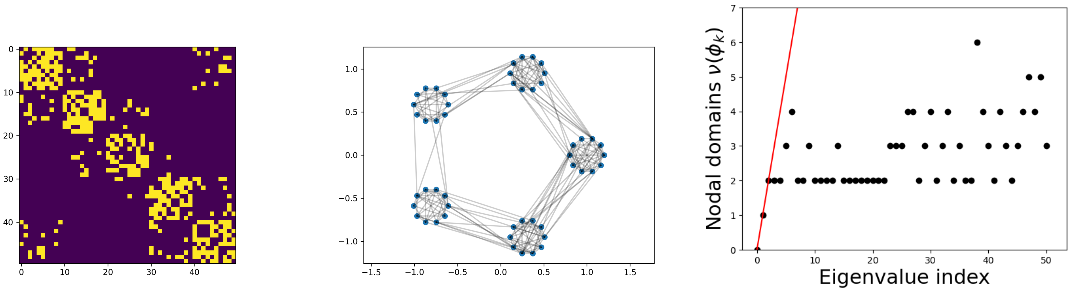

Figure 4.12: An example of a 5-community SBM, with $p_{ii} = 0.7$, $p_{ij} = 0$ if $j \neq i + 1$ or $(i, j) \neq (1, 5)$, and $p_{ij} = 0.1$ otherwise. Similar to Figure 4.9, the adjacency matrix (left) is displayed with yellow pixels indicating an entry of 1, and purple pixels indicating 0. Five community structures are apparent, corresponding to groups of vertices that have a high probability of being connected. Of note is the significant purple, corresponding to the lack of edges between "non-adjacent" communities. The corresponding graph is also displayed (middle); the five circular groups correspond to the communities and have more edges among vertices in the same community than to vertices in adjacent communities, and no edges to vertices in non-adjacent communities. The nodal count for each eigenvector is also display (right); in contrast to Figure 4.11, the nodal count suggests that this SBM sample is slightly further from a complete graph, since higher eigenvectors upwards of 7 nodal domains.

single sample of a 5-community SBM. The five communities are visible in the adjacency matrix plot as the five square regions of primarily yellow pixels. In contrast to the complete graph macroscopic structure, the adjacency matrix plot has significantly more purple pixels. This corresponds to the inability of vertices in non-adjacent communities to be connected by an edge.

Though the nodal counts displayed in Figure 4.11 and Figure 4.12 are similar, the former only has 7 eigenvectors with more than two nodal domains while the second has 25 eigenvectors with more than two nodal domains. This suggests the nodal count can be an effective statistic for networks that are sufficiently sparsified. We explore this aspect next.

Dependence of the Nodal Count on Between-Community Probabilities Here we revisit the 3-community SBM and examine the dependence of the nodal count on the between-community probability p_{ij}. Figure 4.13 shows the result of computing 3-community SBMs for various between-community probabilities. As the probability for an edge between communities increases, higher eigenvectors will have fewer nodal domains, suggesting that the underlying graph is spectrally closer to a complete graph.

Dependence of the Nodal Count on Between-Community Edge-weights Finally, we explored the dependence of nodal counts on both between-community probabilities and edge-weights. We focused on the 5-community model with probabilities $p_{ii} = 0.7$, $p_{ij} = 0.1$ for $1 \leq i < j \leq 5$, and $p_{ij} = 0$ otherwise; and with

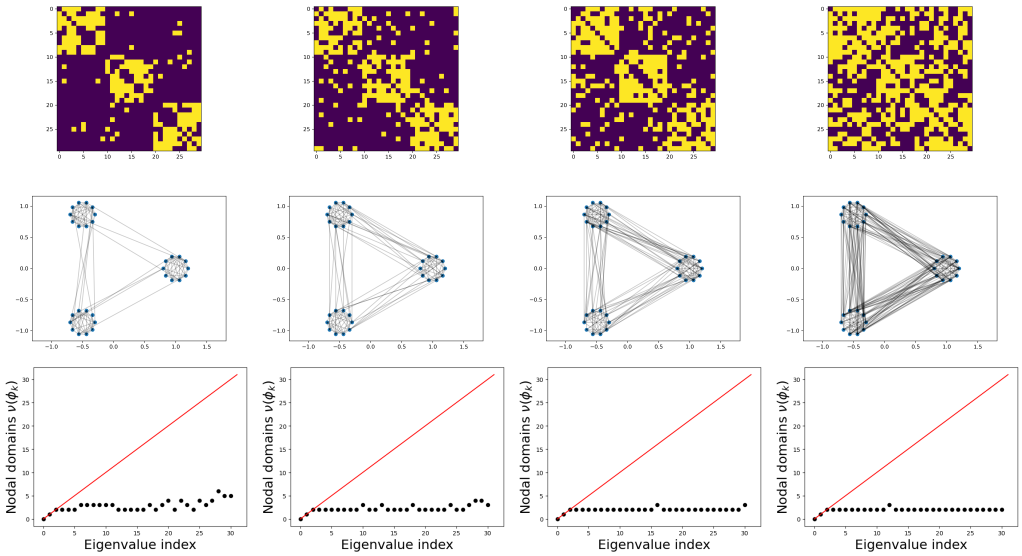

Figure 4.13: The adjacency matrices (top row), graphs (middle row), and nodal counts (bottom row) for a family of 3-community SBMs. Each SBM has a within-community edge probability of $p_{ii} = 0.7$, whereas the between-community edge probabilities are (left to right, columns) $p_{ij} = 0.05, 0.1, 0.2, 0.4$. As the between-community edge probability increases, the nodal counts become closer to the nodal count for the complete graph: the first nodal count has 17 eigenvectors with more than 2 nodal domains, the second nodal count has 8, the third nodal count has 2, and the fourth nodal count has just 1.

edge-weight $w_{ij} = 1$ for vertices i, j in the same community, and $w_{ij} = \alpha$ for edges between communities. We then set $\alpha = 1, \frac{1}{2}, \frac{1}{3}$, and $\frac{1}{4}$ to explore the dependence of nodal counts on the edge-weights. Figure 4.14 shows the underlying graph in this experiment, while Figure 4.14 shows the nodal counts for the various between-community edge-weights α.

The 5-community SBM in Figure 4.14 has an underlying structure of a 5-interval graph: each community of 10 vertices corresponds to a single vertex of the 5-interval graph, and edges between communities

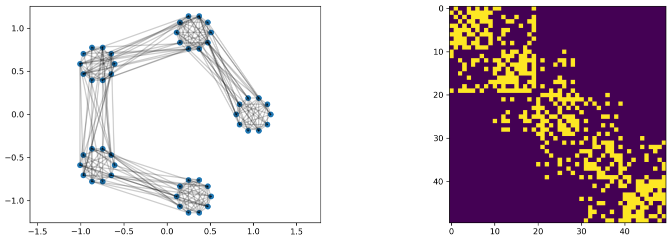

Figure 4.14: A 5-community SBM built to resemble an interval graph on 5 vertices. Left: the underlying graph. Right: the adjacency matrix for the graph, with yellow pixels corresponding to a 1 in the matrix and purple pixels corresponding to 0.

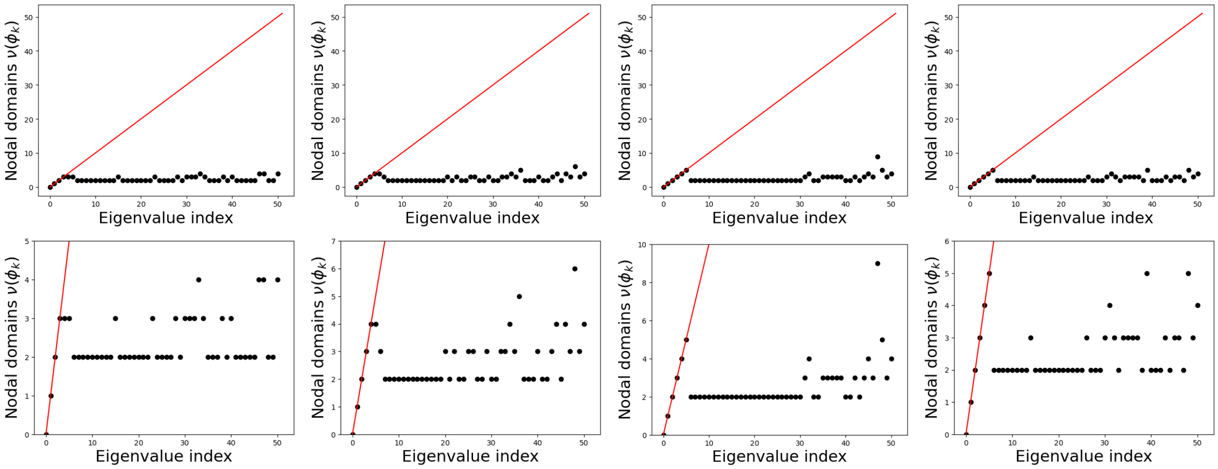

Figure 4.15: The nodal counts for the 5-community SBM in Figure 4.14. Top row, left to right: the nodal counts for $\alpha = 1, 1/2, 1/3, 1/4$. Bottom row: the same nodal counts zoomed in to the domain $[0, 50] \times [0, \max\{\text{counts}\}]$.

correspond to the 4 edges in the 5-interval graph. As the edge-weights for edges between communities, α, decreases, the corresponding nodal counts for the graphs are able to detect the underlying interval graph, as seen in the second row of Figure 4.15; just as the interval graphs are Courant sharp, so are these 5-community SBMs in the first 5 dominant eigenvectors. This is also verified by looking at the corresponding eigenvector plots in Figure 4.16; the underlying SBM graph is plotted in 3D on the plane $z = 0$, and the corresponding eigenvectors are plotted with eigenvector value as the height. The first eigenvector is constant, the second eigenvector has a sign-change "community" right near the center, etc.

Also of note is the drop to two nodal domains in the nodal count plots (Figure 4.15), right after the sharp nodal counts. As seen in the second row of Figure 4.16, the eigenvectors tend to concentrate and localize at individual vertices, giving rise to the two nodal domains seen in the nodal count plots.

4.2.3 Random kNN Geometric Graphs

The final collection of random graphs we study are random geometric graphs for which only the k-nearest neighbors are joined by an edge. These graphs are constructed by sampling a number of points from a bounded domain in $\mathbb{R}^n$ and connecting points with an edge if at least one of them is among the k nearest neighbors of the other. Edge-weights in our construction will be through a self-tuning heat kernel [36], which incorporates an exponential to replicate the heat kernel of the continuum Laplacian.

Definition 4.2.3. *A* ***random kNN geometric graph*** *is constructed from points $x_1, ..., x_N$ sampled from a bounded domain $\Omega \subset \mathbb{R}^n$, in which points are connected with an edge if one is among the k nearest neighbors*

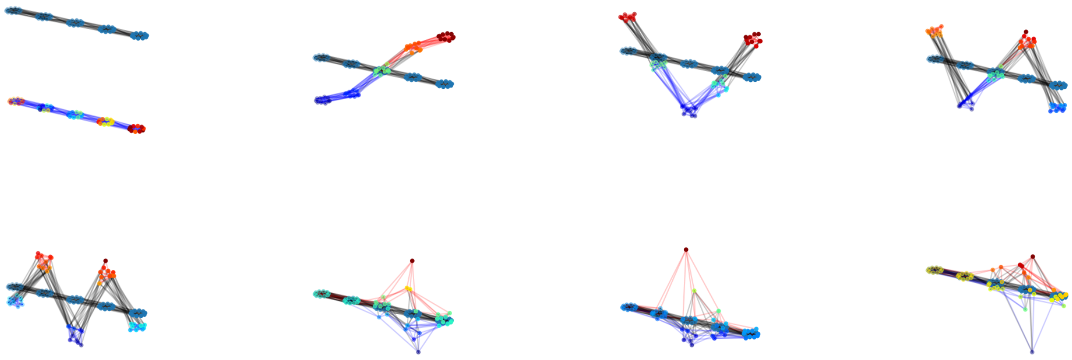

Figure 4.16: The eigenvectors for the 5-community SBM of Figure 4.14, with between-community edge-weight $\alpha = 1/4$. Top row, left to right: eigenvectors 1-4. Bottom row, left to right: eigenvectors 5-8. Except for the first eigenvector, edges are coloured red if they connect two positive vertices, and edges are coloured blue if they connect two negative vertices. Vertex colour corresponds to function value, with red vertices positive, blue vertices negative, and cyan vertices close to zero.

of the other, and edge-weights are given by

$$w_{ij} := \exp\left(\frac{-\|x_i - x_j\|^2}{\sigma_i \sigma_j}\right),$$

*where σ_i is the distance from point x_i to its Kth nearest neighbor, and K may or may not equal k. This choice of w_{ij} is the **self-tuning heat kernel** construction [36].*

Generally, practitioners like to use edge-weights $w_{ij} = \exp\left(\frac{\|x_i - x_j\|}{\epsilon}\right)$, where ϵ is a parameter that can be adapted to the data and data analytic task. The idea behind the self-tuning heat kernel is that the data can be used to effectively choose a parameter ϵ based on local connectivity between points. Moreover, relatively small values of K can provide effective constructions for a wide range of data sets, including data in very high-dimensional spaces. See [36] for more on this construction and related literature.

First, we construct a random kNN geometric graph using the self-tuning heat kernel procedure on points sampled from [0, 1]. The spectrum of the interval graph I_N and the Sturm-Liouville theorem (Theorem 2.1.1) suggest that the nodal deficiencies of eigenvectors for small index should be zero, and this is indeed the case. Figure 4.17 shows the first four non-trivial eigenvectors for $N = 50, 100, 200$ points sampled from [0, 1]. The corresponding nodal counts are shown in Figure 4.18. Note that the nodal counts are Courant sharp, meaning they have zero nodal deficiency, for the first approximately $N/4$ eigenvectors.

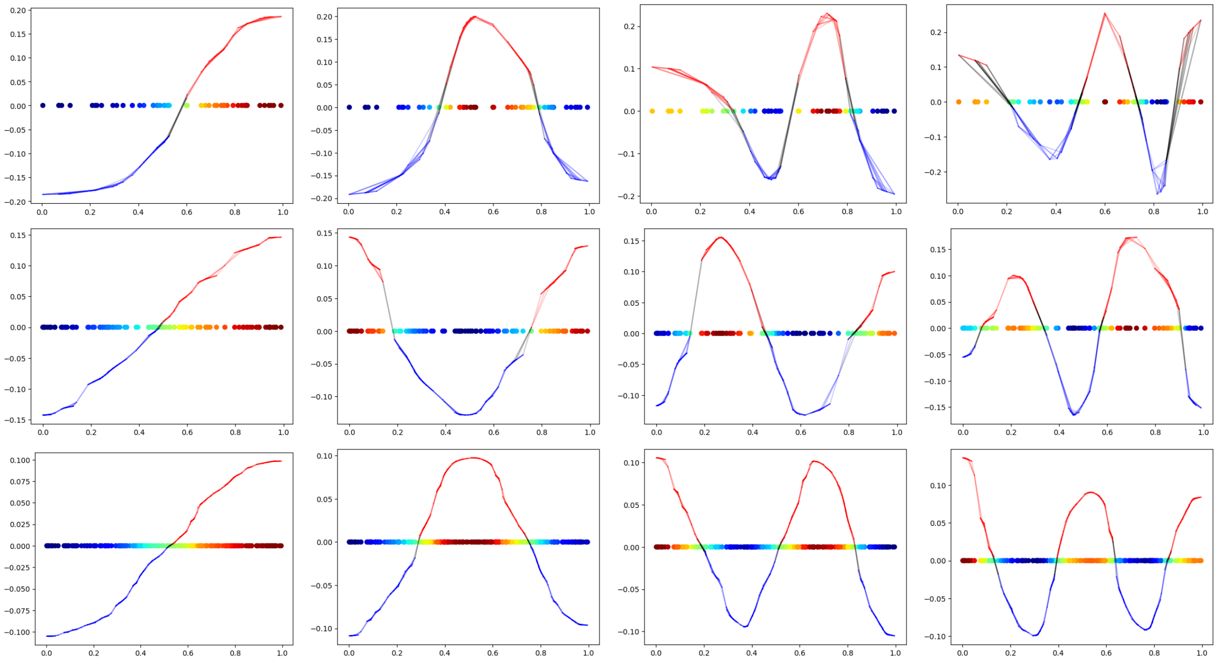

Figure 4.17: The first four non-trivial eigenvectors (left to right columns) of a random kNN geometric graph built on (top to bottom rows) $N = 50, 100, 200$ points sampled from $[0, 1]$. The eigenvector is displayed with the underlying point cloud. Red/blue edges connect vertices of the same sign, while black edges connect vertices of different signs. Though all the eigenvectors have the same qualitative properties, the 4th eigenvector on $N = 50$ points (top right) has noticeably more sign-change edges cutting across non-sign-change edges.

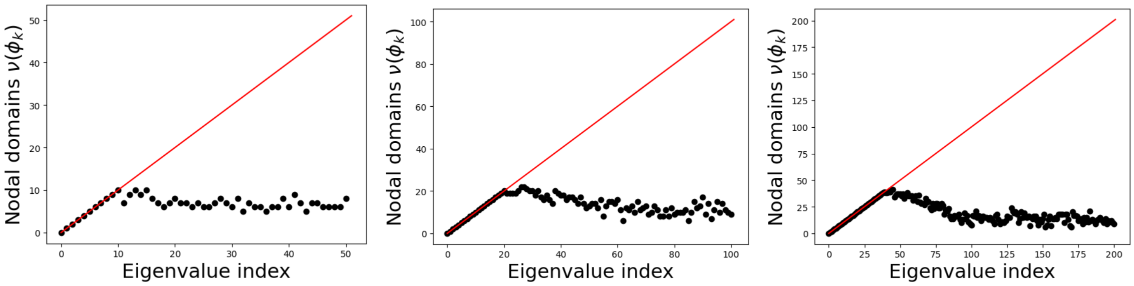

Figure 4.18: Nodal counts for random kNN geometric graphs corresponding to (left to right) $N = 50, 100, 200$ points. The nodal counts are Courant sharp, i.e. the eigenvectors have zero nodal deficiency, for indices up to approximately $N/4$, after which the nodal count decays in a qualitatively similar way for both $N = 100$ and $N = 200$.

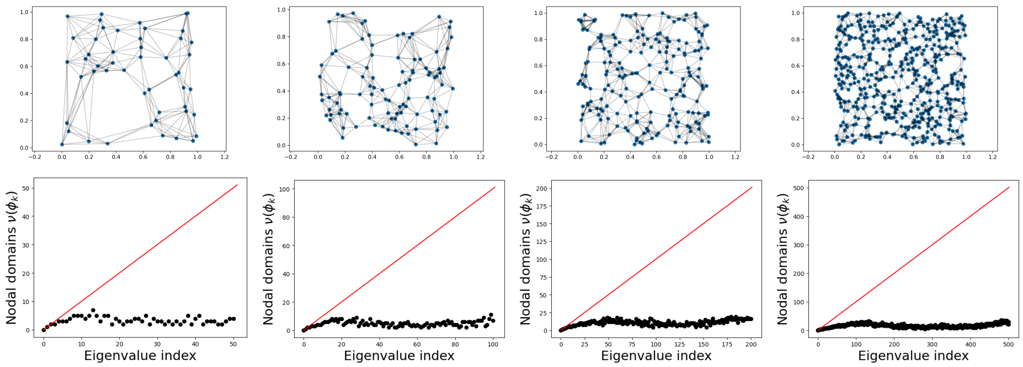

Figure 4.19: Random kNN geometric graphs corresponding to (left to right columns) $N = 50, 100, 200, 500$ points sampled from $[0, 1] \times [0, 1]$. The graphs are displayed (top row) above their respective nodal counts (bottom row). As the number of sampled points increases, the nodal counts seem to peak near the indices $N/4$ and N.

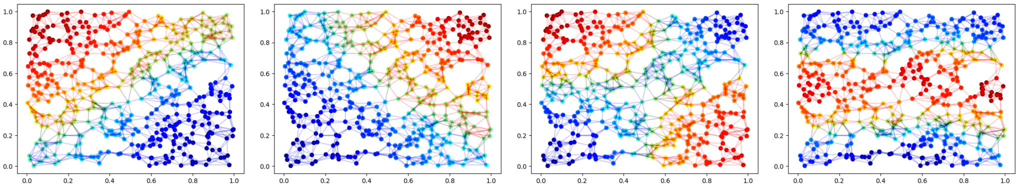

Figure 4.20: The first four non-trivial eigenvectors for the random kNN geometric graph on 500 points are displayed. Even though each point has approximately 7 neighbors, the eigenfunctions seem to match the profiles we would expect from eigenfunctions of the continuum Laplacian on $[0, 1] \times [0, 1]$.

Next we sample points from the unit square $[0, 1] \times [0, 1]$, construct a random kNN geometric graph using the self-tuning heat kernel procedure, and study the nodal counts for graphs as the number of sampled points increases. Figure 4.19 shows the results of this procedure, wherein the graph for $N = 50, 100, 200, 500$ points is displayed above its nodal counts. As the number of points increases, the nodal count tends to a curve with local maxima near indices $N/4$ and N. This behaviour is reminiscent of what we saw in the case of points sampled from $[0, 1]$, in that the nodal count behaved as expected for the first $N/4$ eigenvectors. Of course in the continuum we would expect the nodal count to get arbitrarily high even for relatively small indices of eigenfunctions (namely, eigenfunctions of the form $\sin(k_1 x)\sin(y)$), which is not readily seen in the graph Laplacian nodal counts.

4.3 Handwritten Digit Networks

The MNIST dataset is a popular collection of handwritten digits collected from both National Institute of Standards and Technology (NIST) employees and high school students [62]. In this section we show the results of computing nodal counts for each eigenvector of a self-tuned heat kernel construction using the

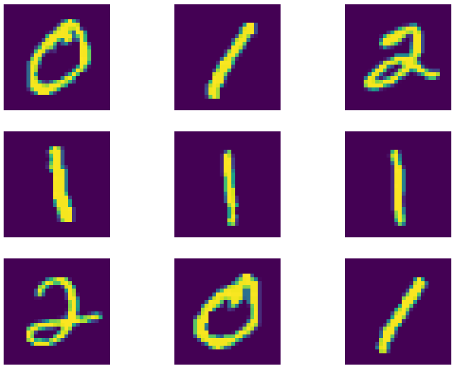

Figure 4.21: Nine samples from the MNIST dataset, in which we have restricted the data to digits with labels 0, 1, or 2.

MNIST dataset. In particular we focused on 312 handwritten digits with labels 0, 1, 2, using two different metrics on the data: Euclidean distances treating each image as a vector, and 2-Wasserstein distances treating each image as a probability measure. The restriction to three labels was for ease of visualization and interpretation. Examples of the digits can be found in Figure 4.21.

Each MNIST data object is a 28×28 array of greyscale pixel values (though Python's default colour scheme is purple to yellow), which can be treated as a vector in $\mathbb{R}^{784}$. Though this interpretation is computationally cheap, information about pixel neighbours above and below a given pixel is discarded. One approach to keeping track of neighboring pixel information is by treating each data object as a function $f(x, y)$ on the unit square $[0, 1] \times [0, 1]$. Moreover, we normalize each function so that we are working with probability distributions, namely we require that $\int_{[0,1]\times[0,1]} f(x, y)d\mu = 1$ for each image; this interpretation and framework lets us incorporate tools from optimal transport theory to study the space of MNIST digits.

In this section we review the necessary theoretical and computational aspects from optimal transport theory, before looking at how nodal counts can be used to analyze this collection of data.

4.3.1 Optimal Transport

Informally speaking, the field of optimal transport asks how best to transport mass from one domain to another, in a way that requires the least amount of effort. One instance of this problem arises when trying to dig a moat, using the removed earth to build a wall; another instance arises in trying to transport baked goods from a collection of bakeries to a collection of cafés. The mathematical framework that encompasses each of these examples utilizes probability measures on appropriate underlying spaces, and has two equivalent formulations.

Given a domain Ω, and two probability measures f_0, f_1 on Ω, *Monge's optimal transport problem* looks to minimize

$$\int_{\Omega} c(x, T(x)) df_0(x),$$

over all possible maps T such that the measures satisfy $f_1 = f_0 \circ T^{-1}$; the cost function $c\colon \Omega \times \Omega \to [0, \infty)$ is often taken to be an L^1 or L^2 cost, so that $c(x, y) = ||x - y||$ or $c(x, y) = ||x - y||^2$, but other choices are possible. While this formulation matches our interpretation of moving mass from one location to another, it turns out that such a minimizing map T may not exist. An alternative formulation is the *Kantorovich optimal transport problem*, which seeks to minimize

$$\int_{\Omega \times \Omega} c(x, y) d\pi(x, y)$$

over all possible probability couplings π between f_0 and f_1, namely π is a probability measure on $\Omega \times \Omega$ whose first and second marginals are precisely f_0 and f_1 respectively. When solutions to the Monge and Kantorovich probems exist, the total costs are the same, but the Monge problem may not have an optimal map. For more on optimal transport, both its history and mathematical analysis, we refer to [63].

In this thesis we focus on the 2-Wasserstein distance between two probability measures f_0, f_1, defined as

$$W_2^2(f_0, f_1) := \inf_{\pi \in \Gamma(f_0, f_1)} \int_{\Omega \times \Omega} \frac{1}{2} d(x, y)^2 d\pi(x, y),$$

where $\Gamma(f_0, f_1)$ is the collection of probability couplings between f_0 and f_1, and $d(x, y)$ is the Euclidean distance between x and $y \in \Omega$. An alternative characterization of the 2-Wasserstein distance is given by a slightly enlarged optimization problem:

$$W_2^2(f_0, f_1) := \inf_{f_t, v_t} \int_0^1 \int_{\Omega} \frac{1}{2} ||v_t||^2 df_t dt, \text{ such that } 0 = \frac{d}{dt} f_t + \nabla \cdot (f_t v_t),$$

where $f_t\colon [0, 1] \to \mathcal{P}(\Omega)$ is a path of probability distributions on Ω and $v_t\colon [0, 1] \times \Omega \to T\Omega$ is a time-dependent, velocity vector field on Ω; see [64] for more details and a proof of the equivalence.

To compute the 2-Wasserstein distance between MNIST digits, treated as probability measures on $\Omega = [0, 1] \times [0, 1]$, we adapt the numerical framework from [65]. Each MNIST image was interpolated onto a simplicial mesh V for Ω, and the 2-Wasserstein geodesic f_t was computed between each pair of images

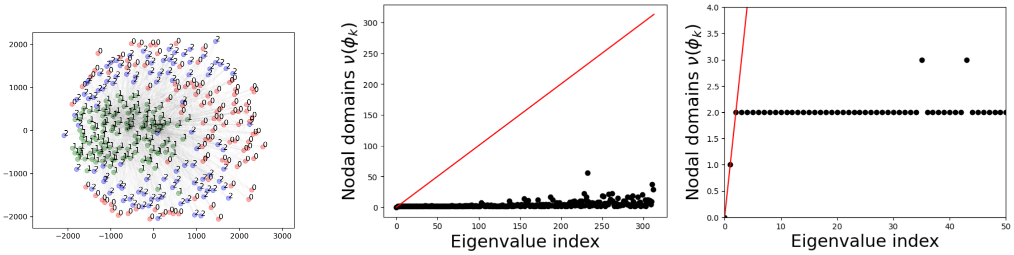

Figure 4.22: The MNIST digits dataset embedded into $\mathbb{R}^2$ using MDS on the Euclidean distance matrix. Left: each dot corresponds to an MNIST image, with the corresponding label displayed. Green corresponds to label 1, red corresponds to label 0, and blue corresponds to label 2. Middle: the nodal counts for all 312 eigenvectors. Right: the nodal counts for the first 50 eigenvectors.

along with the corresponding velocity vector field v_t. The 2-Wasserstein distance is then approximately

$$W_2^2(f_0, f_1) = \sum_{t=0}^{1} \sum_{s \in V} \frac{1}{2} |s| \|v_{t,s}\|^2 f_{t,s},$$

where the first sum is taken over the time discretization, the second sum is over all simplices s in the mesh, and $v_{t,s}$ denotes the value of the vector field v_t on the simplex s. See [65, §3] for more details on the discrete optimal transport implementation.

4.3.2 Results

As in Section 4.2.3, we use a self-tuning heat kernel construction on each distance matrix to construct a graph Laplacian, after which nodal counts are computed. Figure 4.22 shows the 2D embedding and nodal counts for the MNIST data with Euclidean distances. While the digits labeled 1 are grouped together, the digits labeled 0 and 2 seem to be uniformly distributed around the collection of 1s. Moreover, the nodal count plot indicates that the first 30 eigenvectors all have 2 nodal domains. There are two possibilities for such nodal counts: either (1) the data intrinsically lives in a very high dimensional space, and each eigenvector is cutting the data into two roughly equal-sized halves, or (2) the graph Laplacian eigenvectors localize to individual vertices. Inspecting the eigenvectors shows that, in the Euclidean setting, the first non-trivial eigenvector picks out the cluster of data objects labeled 1, but then all subsequent eigenvectors are localizing to individual vertices; these eigenvectors are displayed in Figure 4.23.

As mentioned, the MNIST digits are inherently 2D objects, and so incorporating the intrinsic 2D structure in the metric should lead to stronger clustering behaviour in the data. Indeed, when using the 2-Wasserstein distance to compare MNIST digits, we see better separation between the digits labeled 1, as well as the digits labeled 0 and 2; see Figure 4.24, left. Moreover, the nodal counts of eigenvectors seem to decompose the data

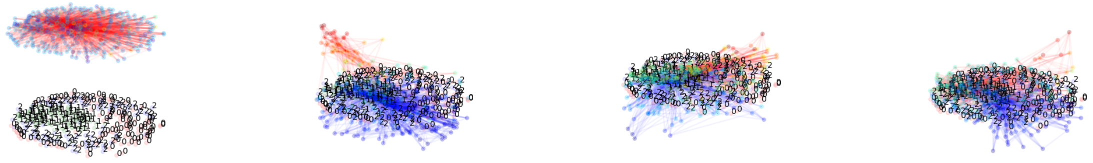

Figure 4.23: The first four eigenvectors of the self-tuning heat kernel Laplacian for the MNIST digits dataset using Euclidean distances. Top row, left to right: eigenvectors 1 and 2. Bottom row, left to right: eigenvectors 3 and 4. Height corresponds to eigenvector value, and edges are coloured red if they connect two positive vertices, or blue if they connect two negative vertices; sign-change edges are not displayed. Also, vertex labels are drawn on the plane $z = 0$. The first eigenvector is constant, the second eigenvector localizes on the cluster of digits labeled 1, and the next two eigenvectors do not exhibit readily interpretable results.

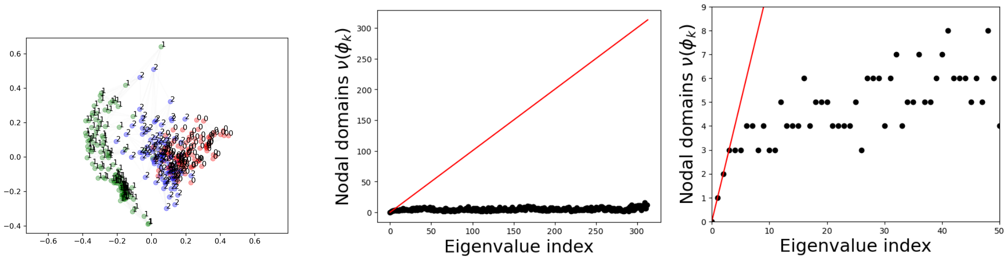

Figure 4.24: The MNIST digits dataset embedded into $\mathbb{R}^2$ using MDS on the 2-Wasserstein distance matrix. Left: each dot corresponds to an MNIST image, with the corresponding label displayed. Green corresponds to label 1, red corresponds to label 0, and blue corresponds to label 2. Using the 2-Wasserstein metric between data objects gives stronger separation between each class of digits. Middle: the nodal counts for all 312 eigenvectors. Right: the nodal counts for the first 50 eigenvectors. Unlike in the Euclidean, none of these eigenvectors (beyond the second) have 2 nodal domains, suggesting the 2-Wasserstein distance better detects the underlying manifold structure of the MNIST data manifold.

into more domains as compared to the Euclidean construction (Figure 4.24, right two plots). Indeed, plotting the first four eigenvectors shows that, as in the Euclidean setting, the second eigenvector localizes to the cluster of data objects labeled 1. Unlike the Euclidean case, however, the third and fourth eigenvectors behave as the third eigenvectors of a 3-interval graph in two distinct ways. This suggests that there is interesting structure using the 2-Wasserstein metric not detected by the Euclidean metric, and that nodal counts can be used to detect whether a metric is more or less intrinsic to the data.

4.4 Discussion

In this chapter a number of numerical examples were presented, illustrating the behaviour of the graph spectral flow for a number of simple graphs, and then transitioning to the study of nodal counts for all graph Laplacian eigenvectors for random graphs, as well as graphs arising in data contexts. The nodal counts seems to capture intrinsic structure in the data: in Section 4.2.2 and Section 4.2.3, the underlying structure

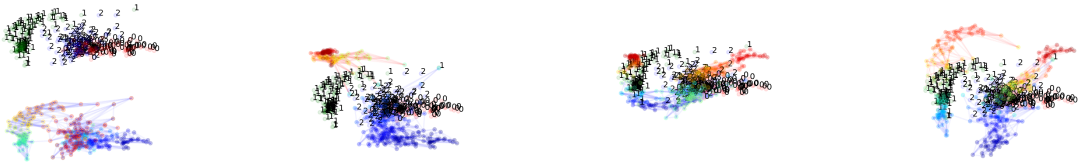

Figure 4.25: The first four eigenvectors of the self-tuning heat kernel Laplacian for the MNIST digits dataset, using 2-Wasserstein distances. Top row, left to right: eigenvectors 1 and 2. Bottom row, left to right: eigenvectors 3 and 4. Height corresponds to eigenvector value, and edges are coloured red if they connect two positive vertices, or blue if they connect two negative vertices; sign-change edges are not displayed. Also, vertex labels are drawn on the plane $z = 0$. The first eigenvector is constant, the second eigenvector seems to localize on the cluster of digits labeled 1, as in the Euclidean case, though the next two eigenvectors behave similar to eigenvectors of the 3-interval graph. Of particular note is that there seem to be two intrinsic 3-interval graphs captured by the eigenvectors and their nodal domains.

of a 5-interval graph and continuum interval, respectively, were effectively detected by the nodal counts. In Section 4.3 we took the same construction and applied it to real data, and saw that graphs constructed using intrinsic metrics saw nodal counts that more accurately captured intrinsic structure. Of course what the intrinsic structure of the MNIST data actually is remains open to debate, these examples do suggest that nodal counts can be an effective tool for detecting the presence of intrinsic structure.

We note that all of these examples saw relatively small graphs used in the analysis. One aspect of future work, discussed next, is extending these tools to work for larger, messier, data sets.

CHAPTER 5

Future Directions

In this final chapter we comment on a few ongoing avenues of research related to the graph spectral flow, including considerations theoretical and applied.

5.1 Eigenvalue Multiplicity

While the eigenvalues for weighted networks will generically be simple, the case of random geometric graphs suggests that, as more points are added to the graph, the spectrum of the graph Laplacian will converge to the spectrum of the continuum Laplacian; see Section 5.3 for a discussion of results in this direction. Hence if the underlying domain has eigenvalues with multiplicity, the graph Laplacian will have distinct eigenvalues that converge to the same value. Thus, care should be taken when studying nodal counts for eigenvectors with eigenvalues that are close in value. One possibility is to incorporate a spectral gap into the analysis. This is the subject of ongoing work.

5.2 Nodal Counts as a Statistical Tool

As suggested in Section 4.2, the first $\approx N/4$ nodal counts for a random geometric graph seem to encode information about the nodal counts for the corresponding continuum eigenfunctions. If we want to use graph nodal counts as an effective approximation for continuum nodal counts, and interpret these counts in a data context, then some sort of thresholding should occur. Higher eigenvectors for graphs tend to behave like indicator functions for individual vertices, explaining the trend to $\nu(\psi) = 2$ for the nodal counts of random geometric graphs.

Despite this, using the first few nodal counts can still be an effective statistic when studying graphs G_i, especially graphs of different sizes. One should normalize the counts to ensure they serve as a robust statistical tool, of which two possibilities stand out:

1. for a family of networks, retain the first k nodal counts and normalize the vectors by subtracting the mean and dividing by standard deviations (similar to Section 2.4.1), or

2. transform the nodal count vectors into piecewise-linear curves $\gamma_{G_i} \colon [0,1] \to \mathbb{R}$ such that $\gamma_{G_i}(0) =$

$0, \gamma_{G_i}(\frac{1}{|V|}) = \frac{1}{|V|}$, and $\gamma_{G_i}(1) = \frac{\nu(\phi_{|V|})}{|V|}$.

The former requires a choice of k dominant eigenvectors and is adapted to specific families of graphs, but is easier to work with numerically. The latter is independent of the specific family of graphs, but general properties of the construction are less clear. Note that the former preserves the fact that complete graphs have $\nu(\phi_k) = 2$ for $k \geq 2$, whereas the latter will have $\gamma_{K_n} \to 0$ pointwise as $n \to \infty$. The utility of each of these approaches is the subject of ongoing work.

5.3 Consistency

Finally, we address the aspect of consistency. Consistency in our framework refers to the convergence of graph-based operators to their continuum counterparts, and has seen extensive use in the machine learning community. In the statistics community the term consistency has been used since the 1920s [66], and generally refers to an assumption that performing a statistical procedure with unlimited data will illuminate the underlying truth. As an interesting example of our type of consistency, a continuum analogue of the well-known PageRank algorithm is proposed in [67]. Here a possibly degenerate, elliptic PDE is given as a continuum analogue to the PageRank equation, and various convergence properties are discussed.

More generally, consistency aspects of graph Laplacians are well understood [68, 69]. Let $\Omega \subset \mathbb{R}^d$ be an open, bounded, connected domain (with Lipschitz boundary) with $d \geq 2$, and suppose we have point clouds $X_n = \{x_1, ..., x_n\}$ sampled from Ω uniformly. Let ϵ_n be a sequence of positive numbers converging to 0 such that

$$\lim_{n\to\infty} \frac{(\log n)^{1/d}}{n^{1/d}} \frac{1}{\epsilon_n} = 0.$$

This requirement arises from optimal transport conditions on matching empirical measures $\nu_n = \frac{1}{n} \sum_i \delta_{x_i}$ to the uniform measure on Ω, though the results hold as well when points are sampled with respect to a (possibly non-uniform measure) μ. We also note that the original results treated ϵ_n differently depending on if $d = 2$ or $d \geq 3$, but such dependence on dimension was alleviated in recent work [70]. We note that the Suppose $\eta\colon [0, \infty) \to [0, \infty)$ is a kernel function that satisfies

1. $\eta(0) > 0$ and η is continuous at 0,

2. η is non-increasing, and

3. $\int_0^\infty \eta(r) r^{d+1} dr < \infty$.

Set $\sigma_\eta = \int_{\mathbb{R}^d} \eta(h)|h_1|^2 dh$, where $h = (h_1, ..., h_d)$, let $\lambda_k^{(n)}$ denote the kth eigenvalue of the graph Laplacian $\mathcal{L}_{n,\epsilon_n}$

for the geometric graph built on the sampled points X_n with edge weights $w_{ij} := \eta(\frac{\|x_i - x_j\|}{\epsilon_n})$, and let λ_k be the kth eigenvalue of the Laplacian Δ on Ω.

Theorem 5.3.1 ([68]). *As $n \to \infty$,*

1. *for each k,*
$$\lim_{n\to\infty} \frac{2\lambda_k^{(n)}}{n\epsilon_n^2} = \sigma_\eta \lambda_k,$$
and

2. *for each k, if $\{u_k^{(n)}\}$ is a sequence of unit norm eigenvectors of $\mathcal{L}_{n,\epsilon_n}$ associated to the eigenvalue $\lambda_k^{(n)}$, then there exists a sub-sequence that converges to u, a λ_k eigenfunction of Δ.*

Convergence in the above theorem is in the TL^2 framework: for a domain Ω define

$$TL^2(\Omega) = \{(\mu, f) \colon \mu \in \mathcal{P}, f \in L^2(\Omega, \mu)\},$$

where $\mathcal{P}$ is the space of (Borel) probability measures on Ω. $TL^2(\Omega)$ consists of pairs of probability measures μ and L^2 functions on Ω with respect to μ, and provides the right kind of space to compare geometric graphs, and functions defined on them, with the underlying continuum and its functions. This space also comes with a metric
$$d_{TL^2}((\mu, f), (\nu, g)) := \inf_{\pi\in\Gamma(\mu,\nu)} \left(\int \int_{\Omega\times\Omega} |x-y|^2 + |f(x) - g(y)|^2 d\pi(x,y) \right)^{1/2};$$
$\Gamma(\mu, \nu)$ is the collection of optimal transport plans between measures μ and ν, which can also be described as the set of all (Borel) probability measures on $\Omega \times \Omega$ that have marginals μ and ν on the first and second variable respectively.

The utility of this framework is that graphs, corresponding to the empirical measures empirical measures $\nu_n = \frac{1}{n}\sum_i \delta_{x_i}$, can be compared in an L^2 framework to L^2 functions on the domain Ω the graph was sampled from, since δ_n and μ are both measures on the domain Ω. Moreover, the proof makes use of non-local energy functionals that bridge the gap between the graph Dirichlet energy

$$\langle u, \mathcal{L}v \rangle = \sum_{(i,j)\in E} w_{ij}(u_i - u_j)(v_i - v_j)$$

and the domain's Dirichlet energy

$$\mathcal{E}(u, v) := \int_{\Omega} \nabla u \cdot \nabla v d\mu.$$

Recall the edge-based flow construction from Definition 3.3.4:

$$\begin{aligned}
B_\sigma(u, v) = & \sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i - u_j)(v_i - v_j) + \sum_{i\in V} Q_i u_i v_i \\
& + \sum_{(i,j)\in E_\pm} w_{ij} \frac{1}{1+\sigma}(u_i - u_j)(v_i - v_j) \\
& + \sum_{(i,j)\in E_\pm} w_{ij} \frac{\sigma}{1+\sigma}\Big[(1 + q_{ji})(u_i - u_{0_{ij}})(v_i - v_{0_{ij}}) \\
& \qquad\qquad + (1 + q_{ij})(u_j - u_{0_{ij}})(v_j - v_{0_{ij}})\Big] \\
& + \sigma \sum_{i\in V_{gh}} u_i v_i.
\end{aligned}$$

By Theorem 5.3.1 we can argue that

$$\sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i - u_j)(v_i - v_j) \to C \int_{\Omega\setminus\Gamma_\epsilon} \nabla u \cdot \nabla v d\mu$$

and

$$\sum_{(i,j)\in E_\pm} w_{ij} \frac{1}{1+\sigma}(u_i - u_j)(v_i - v_j) \to C \frac{1}{1+\sigma} \int_{\Gamma_\epsilon} \nabla u \cdot \nabla v d\mu$$

as more points are sampled from Ω for appropriate scaling constants C, where Γ is the nodal set of the limit of the kth eigenvectors of $\mathcal{L}_{n,\epsilon_n}$ and $\Gamma_\epsilon := \{x \colon d(x, \Gamma) < \epsilon\}$ is an ϵ neighborhood of Γ; the ϵ dependence here is through geometric properties of the sign-change edges $E_\pm$.

Of particular difficulty is the term

$$\sum_{(i,j)\in E_\pm} w_{ij} \frac{\sigma}{1+\sigma}\Big[(1 + q_{ji})(u_i - u_{0_{ij}})(v_i - v_{0_{ij}}) + (1 + q_{ij})(u_j - u_{0_{ij}})(v_j - v_{0_{ij}})\Big].$$

Note that the differences in the sum are based at the nodal set Γ, and hence the limiting energy may not necessarily be a Dirichlet energy on Γ_ϵ. Moreover, the coefficients $1 - \frac{\psi_i}{\psi_j}$ pose an issue in that harmonic functions have polynomial expansions near nodal sets [71]. Indeed, one candidate for such a non-local energy functional is

$$\int_{\Gamma}\int_{\Gamma_\epsilon^+}\int_{\Gamma_\epsilon^-}\left(1-\frac{\psi(y)}{\psi(x)}\right)(u(x)-u(z))(v(x)-v(z))+\left(1-\frac{\psi(x)}{\psi(y)}\right)(u(y)-u(z))(v(y)-v(z))dxdydz,$$

where $\Gamma_\epsilon^\pm := \Gamma_\epsilon \cap \{\pm\psi > 0\}$. Bounding such functionals to incorporate the Γ-convergence framework, which is a technique that is often used in consistency proofs, does not seem viable due to the presence of $\frac{1}{\psi}$ in the integrands, without further assumptions on the geometry of Γ_ϵ and the space of functions from which u, v are taken from.

Another approach would be to re-weight the edges appearing in the construction of $G_{\psi,\sigma}$ Definition 3.3.1 with a regularizing term, such as $c_{ij} = -\psi_i\psi_j$ for each sign-change edge $(i, j) \in E_\pm$. This would result in the bilinear form

$$\begin{aligned} B_\sigma(u,v) = &\sum_{(i,j)\in E\setminus E_\pm} w_{ij}(u_i-u_j)(v_i-v_j) + \sum_{i\in V} Q_i u_i v_i \\ &+ \sum_{(i,j)\in E_\pm} w_{ij}a_{ij,\sigma}(u_i-u_j)(v_i-v_j) \\ &+ \sum_{(i,j)\in E_\pm} w_{ij}(-\psi_i\psi_j)\frac{\sigma}{1+\sigma}\Big[(1+q_{ji})(u_i-u_{0_{ij}})(v_i-v_{0_{ij}}) \\ &\qquad\qquad + (1+q_{ij})(u_j-u_{0_{ij}})(v_j-v_{0_{ij}})\Big] \\ &+ \sigma\sum_{i\in V_{gh}} u_i v_i, \end{aligned}$$

where $a_{ij,\sigma} := \frac{1+(1-(-\psi_i\psi_j))\sigma}{1+\sigma}$ is chosen so that ψ is still in the spectrum of B_σ for all σ.

While this seems like a viable approach, one major difficulty arises. Note that B_σ corresponds to the graph Laplacian of a ψ-subdivision with edge-weights $b_{ij,\sigma} = \frac{(1+q_{ji})(-\psi_i\psi_j)\sigma}{1+\sigma}$ for each ghost vertex edge $(i, 0_{ij})$, and edge-weights $a_{ij,\sigma} = \frac{1+(1-(-\psi_i\psi_j))\sigma}{1+\sigma}$ for each sign-change edge (i, j). A key feature of the original vertex-based construction was that as $\sigma \to \infty$, the edge-weights for each sign-change edge went to 0, and the topology of the underlying graph was precisely that of the desired nodal domains with Dirichlet boundary conditions. In this altered construction though, as $\sigma \to \infty$, the sign-change edge edge-weights do not tend to zero; thus while B_σ behaves as we want it to in a spectral manner, the underlying combinatorics, which our proofs relied on, do not.

This discussion motivates the search for purely spectral characterizations of nodal counts that do not rely on the “correct” underlying graph topology. Indeed, such characterizations would make this tool more readily usable for data, and is the subject of ongoing work.

REFERENCES

[1] W. Stone, *Elementary Lessons on Sound ...* Elementary Lessons on Sound, Macmillan & Company, 1879.

[2] W. Hamilton, J. L. Marzuola, and H.-t. Wu, "On the behavior of 1-Laplacian ratio cuts on nearly rectangular domains," *Information and Inference: A Journal of the IMA*, 12 2020. iaaa034.

[3] W. Hamilton, "A graph spectral flow for computing nodal deficiencies," 2021.

[4] I. Chavel, *Eigenvalues in Riemannian geometry*, vol. 115 of *Pure and Applied Mathematics*. Academic Press, Inc., Orlando, FL, 1984. Including a chapter by Burton Randol, With an appendix by Jozef Dodziuk.

[5] L. C. Evans, *Partial differential equations*, vol. 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, second ed., 2010.

[6] B. Simon, *Sturm Oscillation and Comparison Theorems*, pp. 29–44.

[7] M. Reed and B. Simon, *Methods of modern mathematical physics. IV. Analysis of operators*. Academic Press [Harcourt Brace Jovanovich, Publishers], New York-London, 1978.

[8] A. k. Pleijel, "Remarks on Courant's nodal line theorem," *Comm. Pure Appl. Math.*, vol. 9, pp. 543–550, 1956.

[9] A. Henrot, ed., *Shape optimization and spectral theory*. De Gruyter Open, Warsaw, 2017.

[10] W. A. Strauss, *Partial differential equations*. John Wiley & Sons, Ltd., Chichester, second ed., 2008. An introduction.

[11] G. H. Hardy and E. M. Wright, *An introduction to the theory of numbers*. Oxford University Press, Oxford, sixth ed., 2008. Revised by D. R. Heath-Brown and J. H. Silverman, With a foreword by Andrew Wiles.

[12] G. Berkolaiko, G. Cox, and J. L. Marzuola, "Nodal deficiency, spectral flow, and the Dirichlet-to-Neumann map," *Lett. Math. Phys.*, vol. 109, no. 7, pp. 1611–1623, 2019.

[13] A. Girouard and I. Polterovich, "Spectral geometry of the Steklov problem (survey article)," *J. Spectr. Theory*, vol. 7, no. 2, pp. 321–359, 2017.

[14] G. Berkolaiko, P. Kuchment, and U. Smilansky, "Critical partitions and nodal deficiency of billiard eigenfunctions," *Geom. Funct. Anal.*, vol. 22, no. 6, pp. 1517–1540, 2012.

[15] G. Cox, C. K. R. T. Jones, and J. L. Marzuola, "Manifold decompositions and indices of Schrödinger operators," *Indiana Univ. Math. J.*, vol. 66, no. 5, pp. 1573–1602, 2017.

[16] T. Kato, *Perturbation theory for linear operators*. Classics in Mathematics, Springer-Verlag, Berlin, 1995. Reprint of the 1980 edition.

[17] M. F. Atiyah, V. K. Patodi, and I. M. Singer, "Spectral asymmetry and Riemannian geometry. III," *Math. Proc. Cambridge Philos. Soc.*, vol. 79, no. 1, pp. 71–99, 1976.

[18] N. Waterstraat, "Fredholm operators and spectral flow," 2016.

[19] C. Godsil and G. Royle, *Algebraic graph theory*, vol. 207 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2001.

[20] M. Newman, *Networks*. Oxford University Press, Oxford, 2018. Second edition of [MR2676073].

[21] F. R. K. Chung, *Spectral graph theory*, vol. 92 of *CBMS Regional Conference Series in Mathematics*. Published for the Conference Board of the Mathematical Sciences, Washington, DC; by the American Mathematical Society, Providence, RI, 1997.

[22] T. Bıyıkoğlu, J. Leydold, and P. F. Stadler, *Laplacian eigenvectors of graphs*, vol. 1915 of *Lecture Notes in Mathematics*. Springer, Berlin, 2007. Perron-Frobenius and Faber-Krahn type theorems.

[23] B. Nica, *A brief introduction to spectral graph theory*. EMS Textbooks in Mathematics, European Mathematical Society (EMS), Zürich, 2018.

[24] J. C. Urschel, "Nodal decompositions of graphs," *Linear Algebra and its Applications*, vol. 539, pp. 60–71, 2018.

[25] E. B. Davies, G. M. L. Gladwell, J. Leydold, and P. F. Stadler, "Discrete nodal domain theorems," *Linear Algebra Appl.*, vol. 336, pp. 51–60, 2001.

[26] Y. Colin de Verdière, "Magnetic interpretation of the nodal defect on graphs," *Anal. PDE*, vol. 6, no. 5, pp. 1235–1242, 2013.

[27] G. Berkolaiko, "Nodal count of graph eigenfunctions via magnetic perturbation," *Anal. PDE*, vol. 6, no. 5, pp. 1213–1233, 2013.

[28] G. Berkolaiko, "A lower bound for nodal count on discrete and metric graphs," *Comm. Math. Phys.*, vol. 278, no. 3, pp. 803–819, 2008.

[29] G. Herschlag, H. S. Kang, J. Luo, C. V. Graves, S. Bangia, R. Ravier, and J. C. Mattingly, "Quantifying gerrymandering in north carolina," *Statistics and Public Policy*, vol. 7, no. 1, pp. 30–38, 2020.

[30] S. Caldera, D. DeFord, M. Duchin, S. C. Gutekunst, and C. Nix, "Mathematics of nested districts: The case of alaska," *Statistics and Public Policy*, vol. 7, no. 1, pp. 39–51, 2020.

[31] D. DeFord, M. Duchin, and J. Solomon, "Recombination:a family of markov chains for redistricting," *Harvard Data Science Review*, 3 2021. https://hdsr.mitpress.mit.edu/pub/1ds8ptxu.

[32] M. Duchin and B. E. Tenner, "Discrete geometry for electoral geography," 2018.

[33] P. K. Chan, M. D. F. Schlag, and J. Y. Zien, "Spectral k-way ratio-cut partitioning and clustering," in *Proceedings of the 30th International Design Automation Conference*, DAC '93, (New York, NY, USA), pp. 749–754, Association for Computing Machinery, 1993.

[34] J. B. Kruskal, Jr., "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proc. Amer. Math. Soc.*, vol. 7, pp. 48–50, 1956.

[35] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT Press, Cambridge, MA, third ed., 2009.

[36] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proceedings of the 17th International Conference on Neural Information Processing Systems*, NIPS'04, (Cambridge, MA, USA), pp. 1601–1608, MIT Press, 2004.

[37] X. Cheng and H.-T. Wu, “Convergence of graph laplacian with knn self-tuned kernels,” 2021.

[38] N. G. Trillos, D. Slepčev, J. von Brecht, T. Laurent, and X. Bresson, “Consistency of cheeger and ratio graph cuts,” *Journal of Machine Learning Research*, vol. 17, no. 181, pp. 1–46, 2016.

[39] E. Barter and T. Gross, “Manifold cities: social variables of urban areas in the uk,” *Proc. R. Soc. A*.

[40] The US Census Bureau, “American community survey 5-year estimates,” 2019. Data retrieved from US Census, https://api.census.gov/data/2019/acs/acs5/profile.html.

[41] J. H. Ward, Jr., “Hierarchical grouping to optimize an objective function,” *J. Amer. Statist. Assoc.*, vol. 58, pp. 236–244, 1963.

[42] C. Belfoure, “Middle-class mix acts as anchor,” *The Baltimore Sun*, Apr 2001.

[43] B. Nadler, S. Lafon, R. R. Coifman, and I. G. Kevrekidis, “Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators,” pp. 955–962, 2005.

[44] R. R. Coifman and S. Lafon, “Diffusion maps,” *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 5–30, 2006. Special Issue: Diffusion Maps and Wavelets.

[45] X. Huo, X. S. Ni, and A. K. Smith, *A Survey of Manifold-Based Learning Methods*. 2008.

[46] A. V. Little, M. Maggioni, and L. Rosasco, “Multiscale geometric methods for data sets i: Multiscale svd, noise and curvature,” *Applied and Computational Harmonic Analysis*, vol. 43, no. 3, pp. 504–567, 2017.

[47] A. V. Little, M. Maggioni, and J. M. Murphy, “Path-based spectral clustering: Guarantees, robustness to outliers, and fast algorithms,”

[48] A. Little, D. McKenzie, and J. Murphy, “Balancing geometry and density: Path distances on high-dimensional data,” 2020.

[49] J. M. Murphy and M. Maggioni, “Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 3, pp. 1829–1845, 2019.

[50] L. van der Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008.

[51] W. Hamilton, J. E. Borgert, T. Hamelryck, and J. S. Marron, “Persistent topology of protein space,” 2021.

[52] E. Schubert and M. Gertz, “Intrinsic t-stochastic neighbor embedding for visualization and outlier detection,” in *Similarity Search and Applications* (C. Beecks, F. Borutta, P. Kröger, and T. Seidl, eds.), (Cham), pp. 188–203, Springer International Publishing, 2017.

[53] G. C. Linderman and S. Steinerberger, “Clustering with t-sne, provably,” *SIAM Journal on Mathematics of Data Science*, vol. 1, no. 2, pp. 313–332, 2019.

[54] A. E. Brouwer and W. H. Haemers, *Spectra of graphs*. Universitext, Springer, New York, 2012.

[55] J. Friedman, “Some geometric aspects of graphs and their eigenfunctions,” *Duke Math. J.*, vol. 69, no. 3, pp. 487–525, 1993.

[56] R. Gera and P. Stănică, “The spectrum of generalized Petersen graphs,” *Australas. J. Combin.*, vol. 49, pp. 39–45, 2011.

[57] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide*. Philadelphia, PA: Society for Industrial and Applied Mathematics, third ed., 1999.

[58] P. Erdős and A. Rényi, “On random graphs. I,” *Publ. Math. Debrecen*, vol. 6, pp. 290–297, 1959.

[59] E. N. Gilbert, “Random Graphs,” *The Annals of Mathematical Statistics*, vol. 30, no. 4, pp. 1141 – 1144, 1959.

[60] B. Bollobás, *Random Graphs*. Cambridge Studies in Advanced Mathematics, Cambridge University Press, 2 ed., 2001.

[61] P. W. Holland, K. B. Laskey, and S. Leinhardt, “Stochastic blockmodels: First steps,” *Social Networks*, vol. 5, no. 2, pp. 109–137, 1983.

[62] Y. LeCun and C. Cortes, “MNIST handwritten digit database,” 2010.

[63] C. Villani, *Optimal transport*, vol. 338 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 2009. Old and new.

[64] J.-D. Benamou and Y. Brenier, “A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem,” *Numer. Math.*, vol. 84, no. 3, pp. 375–393, 2000.

[65] H. Lavenant, S. Claici, E. Chien, and J. Solomon, “Dynamical optimal transport on discrete surfaces,” *ACM Trans. Graph.*, vol. 37, Dec. 2018.

[66] G. Upton and I. Cook, *A Dictionary of Statistics*. Oxford Paperback Reference, OUP Oxford, 2008.

[67] A. Yuan, J. Calder, and B. Osting, “A continuum limit for the pagerank algorithm,” 2021.

[68] N. García Trillos and D. Slepčev, “A variational approach to the consistency of spectral clustering,” *Applied and Computational Harmonic Analysis*, vol. 45, no. 2, pp. 239–281, 2018.

[69] J. Calder, N. G. Trillos, and M. Lewicka, “Lipschitz regularity of graph laplacians on random data clouds,” 2020.

[70] M. Caroccia, A. Chambolle, and D. Slepčev, “Mumford-shah functionals on graphs and their asymptotics,” *Nonlinearity*, vol. 33, pp. 3846–3888, Jun 2020.

[71] S. Zelditch, *Eigenfunctions of the Laplacian on a Riemannian manifold*, vol. 125 of *CBMS Regional Conference Series in Mathematics*. Published for the Conference Board of the Mathematical Sciences, Washington, DC; by the American Mathematical Society, Providence, RI, 2017.