# Bounds on eigenvalues of graphs and some applications of neural networks

by

Abhinav B. Chand

B.S., University of Louisiana at Lafayette, 2016

M.S., University of Louisiana at Lafayette, 2020

---

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Mathematics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2025

# Abstract

Many real-world data come with an inherent relationship structure between the data points that is encoded in the form of graphs/networks. Whereas graph theory studies the topology and combinatorics of graphs, graph signal processing studies signals on graphs and their variations. These two fields are intimately connected by the Laplacian operator, which is an operator acting on signals, and its eigenvalues provide information about the graph structure. The pseudo-inverse of the Laplacian is closely related to effective resistance on the graphs. We introduce a switch identity to simplify the effective resistance computation, similar to series and parallel rules in circuits. We consider the strength of graphs, which is the solution to an optimization problem, and find bounds on the eigenvalues of the Laplacian operator using the strength.

We also train graph neural network models to compute the strength. For this task, we use various node features and, using saliency analysis, show that resistance curvature has a strong connection with the strength of the graphs. To further illustrate the applications of graph neural networks in mathematics, we train graph neural networks to learn the influence probabilities for the site percolation model.

# Bounds on eigenvalues of graphs and some applications of neural networks

by

Abhinav B. Chand

B.S., University of Louisiana at Lafayette, 2016

M.S., University of Louisiana at Lafayette, 2020

---

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Mathematics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2025

Approved by:

Major Professor
Dr. Pietro Poggi-Corradini

# Copyright

# Abstract

Many real-world data come with an inherent relationship structure between the data points that is encoded in the form of graphs/networks. Whereas graph theory studies the topology and combinatorics of graphs, graph signal processing studies signals on graphs and their variations. These two fields are intimately connected by the Laplacian operator, which is an operator acting on signals, and its eigenvalues provide information about the graph structure. The pseudo-inverse of the Laplacian is closely related to effective resistance on the graphs. We introduce a switch identity to simplify the effective resistance computation, similar to series and parallel rules in circuits. We consider the strength of graphs, which is the solution to an optimization problem, and find bounds on the eigenvalues of the Laplacian operator using the strength.

We also train graph neural network models to compute the strength. For this task, we use various node features and, using saliency analysis, show that resistance curvature has a strong connection with the strength of the graphs. To further illustrate the applications of graph neural networks in mathematics, we train graph neural networks to learn the influence probabilities for the site percolation model.

# Table of Contents

# List of Figures

# List of Tables

# Acknowledgments

I am grateful to Dr. Pietro Poggi-Corradini for everything he has done for me. His guidance, suggestions of problems, recommendations, and constant encouragement have shaped my academic career. I thank him for the time, patience, and effort that he devoted to my development without which this degree would not have been possible.

I also thank Dr. Nathan Albin, who has helped me immensely in my journey. A lot of the machine learning that I used in my work was inspired by the materials from his Deep Learning course and was influential in my development. I am indebted to Dr. Gabe Kerr for always being supportive and providing encouragement when I needed it the most. His advice and suggestions have always worked out well for me and I appreciate the kindness he showed me. I would like to thank Dr. Lino Amorim for always being there when I needed help with anything. Whenever I had questions on any topic, he was the first person I approached and I enjoyed the many hours I spent discussing with him. Also, a special thanks go to Dr. Bharat Ratra for his time and effort in serving on my committee and always being there for all the checkpoints during the completion of the degree.

I express my sincere gratitude to Dr. Kim Klinger-Logan who helped me greatly in building my academic portfolio. I appreciate her for all the valuable feedback and advice. I thank Dr. Andy Bennett for his time and patience in working with me, and for all the suggestions that he gave me during my time here. I am grateful to Dr. Leonel Robert for being an inspiration and for helping me get to where I am today. I appreciate the support and advice of Subhash Subedi, which has been influential in my development. Special thanks go to my fellow graduate students who helped me whenever I needed them and were always open to suggestions. I also thank the professors of the math department who supported me in this journey. I appreciate the help of Dr. Sarah Reznikoff, Dr. Marianne Korten, Dr. Virginia Naibo, Tina Orwig, Debra Webb, and Courtney Albin every step of the way.

I would also like to thank the Department of Mathematics, College of Arts and Sciences, and the Graduate School at Kansas State University for providing financial, academic and career development support, and giving me the opportunity to complete my doctoral degree in this revered institution. This research was partially supported by "Interactions among Analysis, Optimization, and Network Science", under NSF award number 2154032. Some of the computing for this project was performed on the Beocat Research Cluster at Kansas State University, which is funded in part by NSF grants CNS-1006860, EPS-1006860, EPS-0919443, ACI-1440548, CHE-1726332, and NIH P20GM113109.

# Dedication

To my parents for their love, support and sacrifice.

# Chapter 1

# Introduction

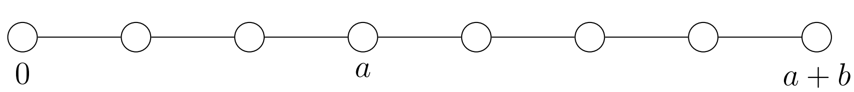
## 1.1 Common thread

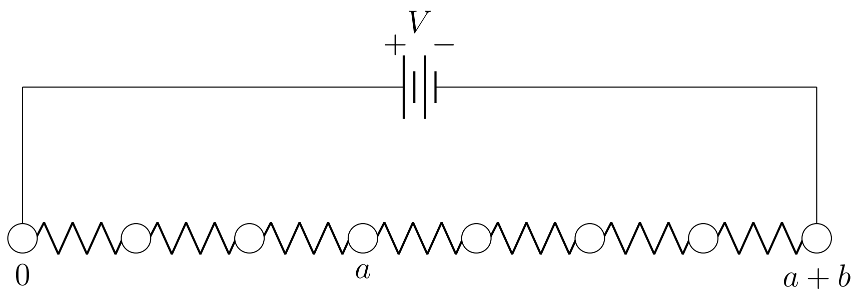
Consider the following simplified version of the Gambler's Ruin problem posed by Pascal to Fermat in 1656 [1]:

> Alice and Bob have initial amounts of money \$$a$ and \$$b$, respectively. They play a game in rounds, where in each round they toss a fair coin. If it lands heads, Alice wins a dollar from Bob, and if it lands tails, Bob gets a dollar from Alice. What is the probability that Alice will eventually get all of Bob's money?

The beauty of this problem lies in the fact that it can be viewed from many perspectives and that each new perspective sheds light on a connection that seems befitting only with hindsight. A new perspective is provided by the following problem.

> An agent performs a random walk on the path graph $G$ labelled by integers from 0 to $a+b$, starting at an interior node $a$. The probability of walking to any neighbor is $\frac{1}{2}$. What is the probability that the agent will reach $a+b$ before visiting 0?






Another beautiful viewpoint is the theory of electric circuits:

> Create an electric circuit in the above path graph where the terminals of a battery are connected to the endpoints 0 and $a+b$, and a unit resistor is placed on each of the edges (see Section 1.2 for definition of graph, nodes and edges). What is the induced voltage on the node $a$?

All three of the above problems are equivalent. The connection between random walks and electric networks essentially arises from the theory of harmonic functions on graphs. A harmonic function on the nodes of a graph is a function where the value at a node is the average of the values of its neighbors. The probabilities on the nodes and the voltage function are both harmonic functions.

Using the theory of electric circuits, the third problem can be solved by adding resistors in series. The effective resistance between two nodes in a network is a metric on the network. Effective resistance also has a variational formulation called Dirichlet's principle. Essentially, the voltage function is the minimizer of the energy functional: the quadratic form associated with the Laplacian of the graph, namely $x^T Lx$, where $L = D - A$ is the Laplacian matrix of the graph. Here $D$ denotes the degree matrix defined as $D_{ij} = \deg(i)$ if $i = j$ and 0 otherwise. $A$ denotes the adjacency matrix defined as $A_{ij} = \delta_{i\sim j}$ where $i \sim j$ if $i$ and $j$ are adjacent nodes.

From a variational point of view, this problem is asking for the values on the unlabeled nodes, given the values on the labeled nodes to minimize the energy of the graph, $\sum_{x\sim y}(v(x) - v(y))^2$. Here, $x \sim y$ if there is an edge connecting them. This problem is a special case of semi-supervised learning on graphs, and this leads to connection with graph representation learning and graph neural networks. The equivalent formulation of the above three problems in semi-supervised learning is the following:

> Consider a path graph $G$ with node labels 0 and 1 which are attached to the left and right endpoints, respectively. Infer the node labels on the interior nodes to make the label function "smooth" (minimize the Laplacian quadratic form).

The four viewpoints- probability theory, random walk on graphs, circuit theory, and machine learning–have led to important advancements in the fields of mathematics, computer science, and machine learning. To briefly summarize this thesis, we introduce a transformation rule such as the series-rule in the circuit problem above, introduce new bounds on the eigenvalues of the Laplacian $L$ and use graph machine learning to infer missing values on the nodes and edges of graphs for certain combinatorial tasks.

## 1.2 Graphs and signals

Graphs are a form of abstraction for real-world networks such as road networks, electric networks, friendship networks, communication networks, and molecular structures. A graph $G$ consists of a set of nodes (vertices) called $V$ and a set of edges $E$ that are ordered or unordered pairs of nodes in $V$ according to whether the graph is directed or undirected. An edge represents a connection or relationship between two nodes in the graph. The graph is denoted by $G = (V, E)$. For example, a social network could be directed if it has a follower/followed relationship (X network), whereas it could be undirected or bidirected if it has only mutual relationships (Facebook network). Loops in graphs are edges that connect the same node. A multigraph is a generalization of graphs in which there are several distinct edges connecting two vertices.

A weighted graph $G$ is a graph along with a function $w : E \to [0, \infty)$ and is denoted as $G = (V, E, w)$. The class of weighted graphs allows for the encoding of more information on the edges than the class of graphs. For example, road networks can have capacity of travel on the edges and electric networks have conductance on the edges. An unweighted graph can be considered as a weighted graph with all edge weights equal to 1.

Graph signals are functions on the nodes of a graph:

**Definition 1** (Signal)**.** A graph signal is a function $f : V \to \mathbb{R}$. We denote signals by elements of $\mathbb{R}^{|V|}$ or $\ell^2(V)$.

Examples of graph signals include the voltage function of an electric network, real valued opinions of individuals in a social network, pixel intensity of an image considered as a grid graph.

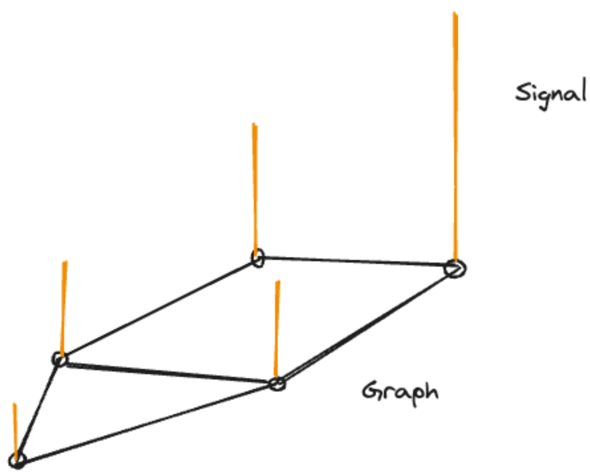



**Figure 1.1**: *Illustration of a graph signal.*

**Definition 2** (Shift operator)**.** [2, 3] Let $G = (V, E)$ be a graph with $|V| = n$. An operator $S : \mathbb{R}^n \to \mathbb{R}^n$ is called a *graph shift operator* or a *one hop operator* if for $i \neq j$, $[S]_{ij} = 0$ if $(i, j) \notin E$. Examples include

- Adjacency
- Laplacian
- Markov matrix

and their normalized versions. The shift operators act on graph signals as follows:

$$Sf_i = \sum_{j=1}^{n} S_{ij} f_j.$$

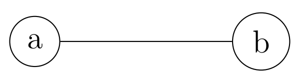
The output $Sf$ is another graph signal where the value at each node is the weighted linear combination of the values of $f$ at the neighbors of the node, weighted by the entries of the matrix $S$.

## 1.3 Importance of Laplacian eigenvalues and bounds on them

Let $G = (V, E, w)$ be a weighted graph. The Laplacian matrix is the matrix $L$ which acts on signals on graphs, i.e. $l^2(V)$ and is defined by the following quadratic form:

$$f^T L f = \sum_{\{a,b\} \in E} w_{a,b}(f(a) - f(b))^2$$

. Note the sum is over unordered edges. Consider the graph with two vertices and one edge of weight 1:




We have

$$\begin{aligned}
(f(a) - f(b))^2 &= ((\delta_a - \delta_b)^T f)^2 \\
&= f^T(\delta_a - \delta_b)(\delta_a - \delta_b)^T f \\
&= f^T \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} f
\end{aligned}$$

The Laplacian for a graph $G$ is given by

$$L = \sum_{(a,b) \in E} w_{a,b}(\delta_a - \delta_b)(\delta_a - \delta_b)^T.$$

This shows that the Laplacian is a positive semi-definite matrix. Hence, its eigenvalues are in $[0, \infty)$.

**Remark 3.** The Laplacian defined above is also called the *combinatorial Laplacian*.

The idea of inferring the geometry of a manifold from the Laplacian eigenvalues is an old problem in analysis and has a rich history. Mark Kac popularized the study of this subject with an article titled with the question "Can one hear the shape of a drum?" which is attributed to Lipman Bers [4]. Although [5] prove that there exist non-isomorphic but isospectral manifolds, i.e., there can be two different drums that produce the same vibrational frequencies, the question can be posed with various restrictions where uniqueness holds.

Similar questions have been asked for graphs [6], but in the case of graphs, examples of isospectral graphs were known much earlier [7]. For example, the graph union $C_4 \cup K_1$, where $C_n$ is the cycle graph on $n$ vertices and $K_1$ is the complete graph on $n$ vertices, and the star graph $S_5$ have identical spectrum [8].

There are many different questions and variations that can be asked under the theme of inferring characteristics of the geometry and topology of graphs from their eigenvalues and vice-versa. One in particular is the question of bounding the expansion of graphs from its eigenvalues, which most likely started from its origins in the study of expanders [9, 10]. These relations between the eigenvalues and isoperimetric constants or expansion constants are not mere mathematical curiosities. Indeed, computing these constants is NP-hard in general and hence, eigenvalues can serve as computable approximations or bounds. Apart from serving a rich playground for computational complexity theory, graph algorithms based on eigenvalues play a crucial role in parallel processing [11], VLSI circuit design [12] and clustering/partitioning [13, 14, 15, 16].

The idea of relating the eigenvalues of the Laplacian and the isoperimetric constant comes from the following theorem:

**Theorem 4.** *Let $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ denote the Laplacian eigenvalues of a graph. The graph has $k$ connected components if and only if $\lambda_k = 0$ and $\lambda_l > 0$ for $l > k$.*

The idea of the proof is as follows: if the graph has $k$ connected components, then the span of the characteristic functions of the $k$ components form a $k$ dimensional subspace of $\ell^2(V)$ and the Rayleigh quotient of any function in this subspace is 0. Hence $\lambda_k = 0$ by (2.15). On the other hand if $\lambda_k = 0$, again by (2.15), there is a $k$ dimensional subspace $S$ such that the Rayleigh quotient is zero on $S$. However, because the quadratic form $\langle Lf, f\rangle = 0$, we must have that for $f \in S$, $f$ is constant within each connected component. Since the dimension of $S$ is equal to the number of connected components, we must have exactly $k$ connected components.

Now when $\lambda_2$ is close to 0, we would expect there to be two components with few bridges between them. Similarly when $\lambda_k$ is close to 0, we would expect there to be $k$ components with few bridges between them (See Figure 1.2).

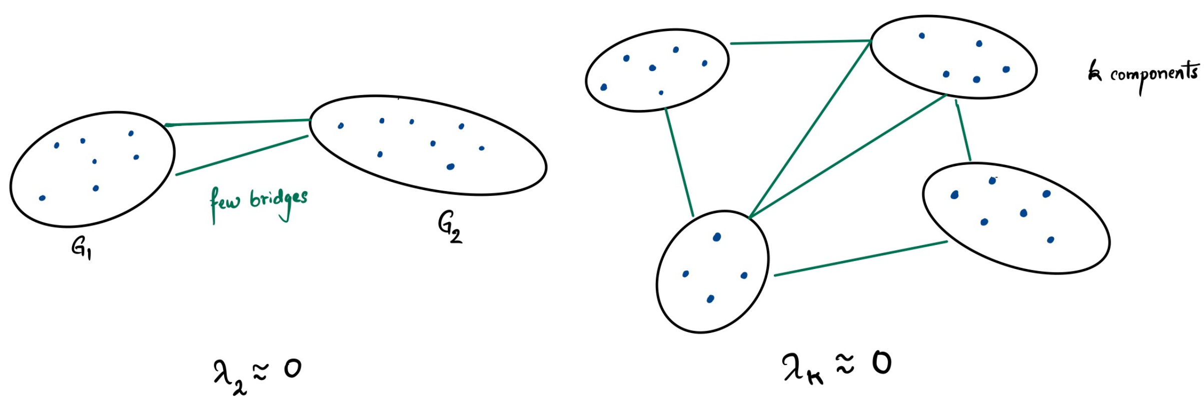



**Figure 1.2**: *Illustration of cases when eigenvalues are close to 0.*

The Cheeger inequality confirms this intuition.

**Theorem 5.** *[17, 18, 19, Cheeger Inequality] Let $G = (V, E)$ be a finite graph and let $d_{\max}$ denote the maximum degree among the vertices in $V$. Let $h(G)$ denote the Cheeger constant* *of G (Definition 33). Then we have the following:*

$$\frac{\lambda_2}{2} \leq h(G) \leq \sqrt{2 d_{\max} \lambda_2}$$

Note that if $\lambda_2$ is small, the Cheeger constant $h(G)$ is small; hence, there exists a subset with a small Cheeger constant that would result in two disjoint components with a bottleneck between them.

Bounding eigenvalues of the Laplacian have many more applications than just providing insights into the connectivity of a graph. Many applications have been discussed in [20]. We list some of the applications mentioned in [20]:

- The max-cut problem is an NP-hard problem in graph algorithms. The largest eigenvalue $\lambda_n$ provides an upper bound on the maximum weight of the edge cuts.
- The second largest eigenvalue of the random walk Laplacian determines the mixing rate of the random walks on a graph. The random walk Laplacian is defined as $L_{\mathrm{rw}} := I - D^{-1}A = D^{-1}L$, i.e., the Laplacian normalized via left multiplication by the degree matrix. Its eigenvalues are in the range $[0, 1]$.
- Families of expanders correspond to families of graphs with $\lambda_2$ uniformly bounded away from zero [20, 21]. Expanders have applications in network design and error-correcting codes.

## 1.4 Graph Neural Networks for combinatorial optimization

Computers have played an influential role in mathematics since the early years of their development. John von Neumann used the ENIAC for complex calculations in fields such as weather prediction and hydrodynamics [22],[23]. A major breakthrough was made in 1976 with the computer-assisted proof of the Four-Color Theorem [24]. Although many computational fields such as number theory and differential equations have always found computing useful for many purposes, it has always been unclear how to utilize computers for many other theoretical parts of mathematics other than for formal verification or algorithmic computation. With the advent of machine learning and artificial intelligence in recent years, AI systems are now being used to discover new mathematical patterns and conjectures, potentially accelerating the pace of mathematical discovery [25, 26].

Graph neural networks (GNNs) have recently gained popularity for solving classical combinatorial optimization problems. Cappart et al. [27] say "Classical algorithms solve problem instances in isolation, while GNNs can learn from patterns in related data distributions." Some applications mentioned in [27] are as follows:

- GNNs can be used to find feasible solutions while solving graph algorithms, such as the traveling salesman problem.
- GNNs can guide branching decisions in branch-and-bound algorithms.
- Neural algorithmic reasoners can be trained to execute classical algorithms, such as Bellman-Ford and Dijkstra's algorithms.

GNNs have permutation invariance and can learn useful representations of graph-structured data. These models can not only solve classical graph algorithms but also provide insights into connections between different graph features, as shown in Chapter 4.

## 1.5 Outline and Contributions

This thesis focuses on three fundamental themes in network science: effective resistance, Laplacian eigenvalues, and graph representation learning. In Chapter 2, we introduce the $p$-modulus of paths between vertices, which generalizes both the shortest path distance and the effective resistance distance. We introduce and prove the switch identity, which we derived from the work of Lawler [28]. This identity can be interpreted as a circuit transformation rule, such as series rule, parallel rule, or star-mesh transformation. Then, we introduce a well-known upper bound on the second Laplacian eigenvalue of a graph in terms of the effective resistance diameter and investigate conditions when equality holds in the inequality.

In Chapter 3, we introduce bounds on the eigenvalues of the Laplacian in terms of the strength of a graph. We illustrate the bounds for certain classes of graphs.

In Chapter 4, we apply graph neural networks to two problems: predicting the edges in the minimum feasible partition problem which gives rise to the strength and predicting the probability of influence for the site percolation model. This chapter shows that graph neural networks can be trained to learn intricate problems in network science and provide insights into connections between various quantities.

In Chapter 5, we discuss the implications of our results and present several problems for future work.

Our main contributions are as follows:

- In Section 2.4 we state and prove the switch identity which was originally stated in the language of loop erased random walks by Lawler [29]. We give a proof in terms of modulus of paths and identify its implications. We show that it can be used as a reduction technique for computing the effective resistance.

- We prove equality conditions for the inequality in Section 2.6 which gives an upper bound on the algebraic connectivity in terms of the effective resistance diameter.

- As shown in Section 3.4, we formulate a lower bound on the strength of a graph in terms of Laplacian eigenvalues.

- As shown In Section 3.5, we formulate a lower bound on the algebraic connectivity in terms of the strength. In order to get this result we use the work of Lee et al. [30] to prove the weaker version of the higher-order Cheeger inequality for the eigenvalues of the combinatorial Laplacian and isoperimetric ratio as shown in Section 3.3.

- We train graph neural networks to learn the minimum feasible partition problem and used saliency analysis to show the connection to discrete curvature in Section 4.3.

- As shown in Section 4.4, we train graph neural networks to learn the site percolation model, which can be used as a benchmark for graph neural network architectures.

# Chapter 2

# Modulus on Graphs, Strength and Effective Resistance

## 2.1 Motivation

The theory of modulus on graphs is a vast generalization of classical graph quantities like min-cut, max-flow, and effective resistance [31]. Modulus on graphs is inspired by conformal capacity which is a concept studied in complex analysis and geometric function theory [32, 33]. For a given family of "objects" in a graph, for example, the family of paths between two vertices, the family of spanning trees, the family of loops, etc., the $p$-modulus of the family quantifies the richness of the family.

Consider the case of the shortest path distance between two vertices $s$ and $t$ in a graph (see Figure 2.1). In contrast, the effective resistance distance is more expressive as it considers both the number of paths between the two vertices as well as their lengths. The effective resistance between $s$ and $t$ is the reciprocal of the 2-modulus of the family of paths between them.

The theory of $p$-modulus on graphs began with the study of the family of walks between two vertices [34]. There are many applications of this theory: introduction of new graph metrics [35], connection to epidemic processes on networks [36, 37], network clustering [38],


Shortest Path Distance = 2    Effective Resistance Distance = 1

**Figure 2.1**: *Comparison of shortest path distance (left) and effective resistance distance (right) between vertices s and t.*

community detection [39], and secure broadcast games [40].

The modulus of the family of paths between two vertices gives rise to the effective resistance, and the modulus of the family of spanning trees gives rise to the strength of graphs via the minimum feasible partition problem.

## 2.2 Definition of the modulus of a family

Let $G = (V, E)$ be a graph, where $V$ is the set of vertices and $E$ is the set of edges. Consider a family of objects $\Gamma$ for which each $\gamma \in \Gamma$ can be assigned an associated function $\mathcal{N}(\gamma, \cdot) : E \to \mathbb{R}_{\geq 0}$ that measures the usage of edge $e$ by $\gamma$. For example, if we consider the family of walks, then for a walk $\gamma = x_0 e_1 x_1 \cdots e_n x_n$, $\mathcal{N}(\gamma, \cdot)$ is equal to the number of times $\gamma$ traverses $e$. The matrix $\mathcal{N}$ is called the usage matrix for the family $\Gamma$. The density on $G$ is a non-negative function on the edge set: $\rho : E \to [0, \infty)$. This is also sometimes referred to as the cost of using an edge. For an object $\gamma \in \Gamma$, we define

$$\ell_\rho(\gamma) := \sum_{e \in E} \mathcal{N}(\gamma, e)\rho(e) = (\mathcal{N}\rho)(\gamma),$$

which represents the total usage cost for $\gamma$ with the given edge costs $\rho$. A density $\rho \in \mathbb{R}_{\geq 0}^E$ is called admissible for $\Gamma$, if

$$\ell_\rho(\gamma) \geq 1, \quad \forall \gamma \in \Gamma.$$

We define the set

$$\mathrm{Adm}(\Gamma) = \{\rho \in \mathbb{R}^E_{\geq 0} : \mathcal{N}\rho \geq 1\}$$

to be the set of admissible densities. For $1 \leq p \leq \infty$ we define the $p$-energy of a density $\rho$ as

$$\mathcal{E}_{\rho,\sigma}(\rho) := \sum_{e \in E} \sigma(e)\rho(e)^p \text{ and } \mathcal{E}_{\infty,\sigma}(\rho) := \max_{e \in E} \sigma(e)\rho(e).$$

**Definition 6** ($p$-modulus)**.** The $p$-modulus of $\Gamma$ is defined to be

$$\mathrm{Mod}_{p,\sigma}(\Gamma) = \inf_{\rho \in Adm(\Gamma)} \mathcal{E}_{\rho,\sigma}(\rho). \tag{2.1}$$

## 2.3 Effective Resistance

The following definitions can be found in Nachmias's book [41]. Let $G = (V, E)$ be a graph with positive edge weights $c_e$ called conductances. The reciprocals $r_e = \frac{1}{c_e}$ are called the edge resistances. In order to define the effective resistance between two nodes, we will need the definition of harmonic function and voltage:

**Definition 7** (Harmonic function)**.** Let $G = (V, E)$ be a graph. A function $f : V \to \mathbb{R}$ is said to be harmonic at $u \in V$ if $f(u) = \frac{1}{\deg(u)} \sum_{u \sim v} f(v)$, where $u \sim v$ if there exists an edge between $u$ and $v$.

**Definition 8** (Voltage function)**.** Let $G = (V, E)$ be a graph with two distinct vertices $s, t \in V$. A voltage is a function $h : V \to \mathbb{R}$ which is harmonic at any $u \in V \backslash \{s, t\}$.

To define the current flow, we first define the notion of an $s$–$t$ flow:

**Definition 9** ($s$-$t$ flow)**.** Let $G = (V, E)$ be a graph, and $\vec{E}$ be the set of oriented edges. An $s$-$t$ flow in $G$ is a function $I : \vec{E} \to \mathbb{R}$ satisfying

1. For any $\{u, v\} \in E$ we have $I(uv) = -I(vu)$.

2. For all $u \notin \{s,t\}$, $\sum_{v\sim u} I(uv) = 0$.

**Definition 10** (Current flow)**.** Let $G = (V, E, c)$ be a weighted graph with edge conductance weights $c : E \to \mathbb{R}_{\geq 0}$ and two distinct vertices $s, t \in V$. Given a voltage $h$, the current flow $I = I_h$ associated with $h$ is defined by $I(uv) = c_{xy}(h(v) - h(u))$.

**Definition 11** (Strength of flow)**.** The strength of a $s$-$t$ flow $I$ is given by

$$\|I\| := \sum_{v\sim s} I(sv).$$

**Definition 12** (Energy of flow)**.** The energy of a flow $I$ from $s$ to $t$ in $G$, denoted by $\mathcal{E}_2(I)$ is defined to be

$$\mathcal{E}_2(I) = \sum_{e\in E} I(e)^2 r_e.$$

The energy of a voltage $h$ is given by

$$\mathcal{E}_2(h) = \sum_{\{u,v\}\in E} (h(v) - h(u))^2 c_e.$$

Finally, we define the effective resistance as follows:

**Definition 13** (Effective resistance)**.** For every non-constant voltage $h$ and a corresponding current flow $I$, the ratio $\frac{h(t)-h(s)}{\|I\|}$ is a positive constant that does not depend on $h$. This ratio is called the effective resistance between $s$ and $t$ in $G$, and is denoted by $\mathcal{R}_{\text{eff}}(s,t)$. The reciprocal is called the effective conductance between $s$ and $t$ and is denoted by $\mathcal{C}_{\text{eff}}(s,t)$.

Dirichlet's principle states that the effective conductance (resistance) can be obtained from the solution of a minimization problem without reference to the current flow. A proof of this principle can be found in [42].

**Theorem 14** (Dirichlet's principle)**.** *Let $G = (V, E)$ be a graph and let $s, t \in V$ be vertices with $s \neq t$. Then*

$$\mathcal{C}_{\mathit{eff}}(s,t) = \inf\{\mathcal{E}_2(h) | h : V \to \mathbb{R}, h(s) = 0, h(t) = 1\}. \tag{2.2}$$

Effective resistance can also be calculated using the pseudoinverse Laplacian, as in 15. In fact, many authors use the following definition for effective resistance:

**Theorem 15.** *The effective resistance between vertices $s$ and $t$ satisfies the following condition:*

$$\mathcal{R}_{\mathit{eff}}(s,t) = (\delta_t - \delta_s)^T L^+(\delta_t - \delta_s).$$

The following theorem shows that effective resistance (conductance) arises from path modulus.

**Theorem 16.** *[43, Theorem 4.2] Let $G = (V, E, \sigma)$ be an undirected graph, let $\Gamma = \Gamma(s,t)$ be the connecting family of walks from $s$ to $t$, two distinct vertices in $V$, and let $1 < p < \infty$. Let $\rho^*$ be the extremal density for (2.1). Then there exists a vertex potential $\phi^* : V \to \mathbb{R}$ such that $\phi^*(s) = 0$, $\phi^*(t) = 1$, and*

$$\rho^*(x,y) = |\phi^*(x) - \phi^*(y)| \qquad \forall (x,y) \in E. \tag{2.3}$$

*Moreover, this $\phi^*$ solves the optimization problem*

$$\begin{aligned} &\textit{minimize} \quad \sum_{(x,y)\in E} \sigma(x,y)|\phi(x) - \phi(y)|^p \\ &\textit{subject to} \quad \phi(s) = 0, \quad \phi(t) = 1. \end{aligned}$$

*When $p = 2$, it follows that $\mathrm{Mod}_2(\Gamma(s,t)) = \mathcal{C}_{\mathit{eff}}(s,t)(s,t)$.*

## 2.4 Switch Identity

The goal of this section is to derive the identity (2.4) that is derived from the work of Lawler in [28] on loop erased random walks (LERW), which were introduced in the process of trying to understand the self-avoiding walk [29]. While Lawler used the product of Green's function to derive a distribution on self-avoiding walks, from which (2.4) follows. We will use the theory of modulus to prove (2.4). Let $\Gamma$ be the family of paths connecting $s$ and $t$. We will prove the switch identity for the 2 modulus of paths below. The corresponding identity for $p$-modulus is not known.

**Theorem 17.** *(switch identity) Let $G = (V, E, w)$ be a weighted graph with $n$ vertices, where $n \geq 3$. Then for distinct $a, b, c \in V$,*

$$\mathrm{Mod}_2(\Gamma(a,b))\,\mathrm{Mod}_2(\Gamma(c,\{a,b\})) = \mathrm{Mod}_2(\Gamma(b,c))\,\mathrm{Mod}_2(\Gamma(a,\{b,c\})). \tag{2.4}$$

To prove this identity, we will need some transformation rules in graphs that preserve the 2-modulus of paths: series rule, parallel rule, and star-mesh transformation. These are used for simplifying the calculation of the 2-modulus. These are well known in circuit theory, but we will use the modulus on networks to derive them below.

First we state the $p$-Dirichlet principle:

**Theorem 18.** *[42] Let $G = (V, E, w)$ be a simple finite connected graph with edge conductances $w \in \mathbb{R}^E_{>0}$. Fix a source and a target: $s \neq t \in V$. The $p$-Dirichlet principle is the following optimization problem*

$$\begin{aligned} &\textit{minimize} \quad \mathcal{E}_{p,w}(h) := \sum_{e=\{x,y\}\in E} w(x,y)|h(x)-h(y)|^p \\ &\textit{subject to} \quad h(s) = 0, h(t) = 1. \end{aligned} \tag{2.5}$$

*The $p$-Dirichlet problem is solves by a function $u \in \mathbb{R}^V$ such that*

$$\begin{aligned} L_{p,w}u(x) &= 0, \\ u(s) = 0 \textit{ and } u(t) &= 1, \end{aligned} \tag{2.6}$$

*where*

$$(L_{p,w}u)(x) := \sum_{y\sim x} w(x,y)|u(y)-u(x)|^{p-2}(u(y)-u(x))$$

*is the weighted* $(p,w)$*-Laplacian. Moreover, when* $\Gamma$ *is the family of all paths connecting* $s$ *and* $t$*,*

$$\mathcal{E}_{p,w}(u) = \operatorname{Mod}_{p,w}\Gamma$$

*where* $u$ *solves* (2.5)*.*

**Series Rule:** Consider a weighted path graph with three nodes. The weights on $(a,x)$ and

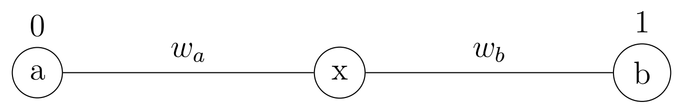



**Figure 2.2**: *Path graph with two edges for series rule.*

$(x,c)$ are $w_a$ and $w_b$ respectively. Using Theorem 18, we need to solve the $(p,w)$-Dirichlet problem for this graph, i.e., find a function $u(x)$ that solves

$$L_{p,\sigma}u(x) = 0, \tag{2.7}$$

$$u(a) = 0 \text{ and } u(b) = 1. \tag{2.8}$$

Note that $L_{p,\sigma}u(x) = -w_a u(x)^{p-1} + w_b(1-u(x))^{p-1}$. Setting this equal to 0 we get

$$\begin{aligned}
w_a u(x)^{p-1} &= w_b(1-u(x))^{p-1}\\
\Big(\frac{u(x)}{1-u(x)}\Big)^{p-1} &= \frac{w_b}{w_a}\\
\frac{u(x)}{1-u(x)} &= \frac{w_b^{\frac{1}{p-1}}}{w_a^{\frac{1}{p-1}}}\\
u(x) &= \frac{w_b^{\frac{1}{p-1}}}{w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}}}.
\end{aligned}$$

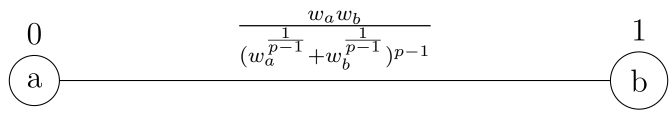



**Figure 2.3**: *Series rule for p-modulus.*

Hence, the $p$-Dirichlet energy is equal to

$$w_a \left( \frac{w_b^{\frac{1}{p-1}}}{w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}}} \right)^p + w_b \left( 1 - \frac{w_b^{\frac{1}{p-1}}}{w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}}} \right)^p = \frac{w_a w_b^{\frac{p}{p-1}} + w_b w_a^{\frac{p}{p-1}}}{(w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}})^p}$$

$$= \frac{w_a w_b (w_b^{\frac{1}{p-1}} + w_a^{\frac{1}{p-1}})}{(w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}})^p}$$

$$= \frac{w_a w_b}{(w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}})^{p-1}}.$$

Thus, we can replace the above graph with a single-edge weighted graph with weight

$$\frac{w_a w_b}{(w_a^{\frac{1}{p-1}} + w_b^{\frac{1}{p-1}})^{p-1}}$$

to get the same $p$-Dirichlet energy.

**Parallel Rule:** We consider a graph with two nodes and two edges with weights $w_1$ and

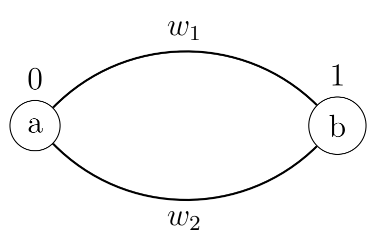



**Figure 2.4**: *Graph on two nodes with two edges for parallel rule.*

$w_2$. We set the potential of a function $h$ on this graph to be to 0 at $a$ and 1 at $b$. For the $p$-Dirichlet problem we need to minimize $\mathcal{E}_{p,w}(h) = w_1 + w_2$. If we want an equivalent minimization problem on a single-edge graph with nodes $a$ and $b$, we can assign the weight $w_1 + w_2$ on that edge.

**Star-mesh transformation for 2-modulus:**

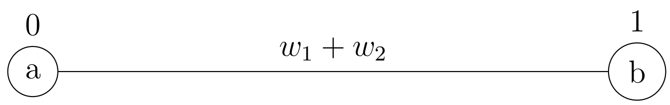



**Figure 2.5**: *Parallel rule for p-modulus.*

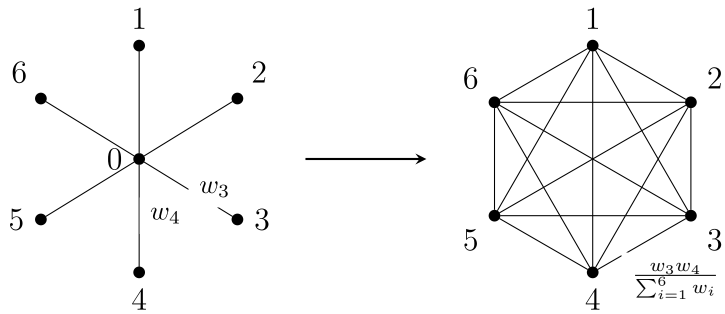



**Figure 2.6**: *Illustration of star mesh transformation.*

**Lemma 19** (star-mesh transformation)**.** *Let $G = (V, E, w)$ be a complete weighted graph. We allow a weight of 0, which means that there is no edge connecting the two vertices. Let $V = \{x_0, ..., x_n\}$. For $x \in V$, let $G/x$ be the graph where we remove the vertex $x$ and all the edges incident to it, and add the following weights to each edge $(x_i, x_j)$ of the graph $G/x$:*

$$\frac{w(x, x_i)w(x, x_j)}{\sum_{k=0}^{n} w(x, x_k)}. \tag{2.9}$$

*Then,* $\mathrm{Mod}_2(\Gamma(s,t); G) = \mathrm{Mod}_2(\Gamma(s,t); G/x)$ *for all vertices $s, t$ in $G \backslash \{x\}$.*

*Proof of Lemma 19.* Without loss of generality, we assume that $x = x_1, s = x_0$ and $t = x_n$. Let $h$ be the potential where we fix $h(x_0) = 0$ and $h(x_n) = \mathcal{R}_{\text{eff}}(x_0, x_n)$. Then the induced current is 1 unit, and hence we have, $Lh = \delta_{x_n} - \delta_{x_0}$, where $L$ denotes the weighted combinatorial Laplacian. Recall that $L(x_i, x_j) = -w(i,j)$ and $L(x_i, x_i) = \sum_{i \sim j} L(x_i, x_j)$. Thus,

$$L(x_j, x_j)h(x_j) + \sum_{i=0, i \neq j}^{n} L(x_j, x_i)h(x_i) = i_{\text{ext}}(j) \tag{2.10}$$

where $i_{\text{ext}}(j)$ denotes the net current out of $j$. For $j = 1$ we have

$$h(x_1) = -\frac{\sum_{k=0, k\neq 1}^{n} L(x_1, x_k)h(x_k)}{L(x_1, x_1)}.$$

Substituting this into the left hand side of (2.10), for $j \neq 1$ we have

$$\begin{aligned}
i_{\text{ext}}(j) &= L(x_j,x_j)h(x_j) + \sum_{i=0,i\neq j,1}^{n} L(x_j,x_i)h(x_i) + L(x_j,x_1)h(x_1) \\
&= L(x_j,x_j)h(x_j) + \sum_{i=0,i\neq j,1}^{n} L(x_j,x_i)h(x_i) - \frac{\sum_{k=0,k\neq 1}^{n} L(x_j,x_1)L(x_1,x_k)h(x_k)}{L(x_1,x_1)} \\
&= \sum_{i=0,i\neq 1}^{n} \Big( L(x_j,x_i) - \frac{L(x_1,x_j)L(x_1,x_k)}{L(x_1,x_1)} \Big) h(x_i) \\
&= \sum_{i=0,i\neq 1}^{n} \Big( w(x_j,x_i) - \frac{w(x_1,x_j)w(x_1,x_k)}{\sum_{k=0}^{n} w(x_1,x_1)} \Big) h(x_i).
\end{aligned}$$

This reduced system satisfies $L_{\text{red}}h = \delta_{x_n} - \delta_{x_0}$ with $h(x_0) = 0$ and $h(x_n) = \mathcal{R}_{\text{eff}}(x_0,x_n)$. The new weights are $L_{\text{red}}(x_j,x_i) = w(x_j,x_i) - \frac{w(x_1,x_j)w(x_1,x_i)}{\sum_{k=0}^{n} w(x_1,x_k)}$ for each edge $(i,j)$ of the graph $G/x_1$. Since $s$ and $t$ are arbitrary, we have $\text{Mod}_2(\Gamma(s,t);G) = \text{Mod}_2(\Gamma(s,t);G/x)$ for all vertices $s,t$ in $G/x$. $\square$

*Proof of Theorem 17.* We prove the theorem using induction on the number of vertices $n$:

Base Case ($n = 3$):

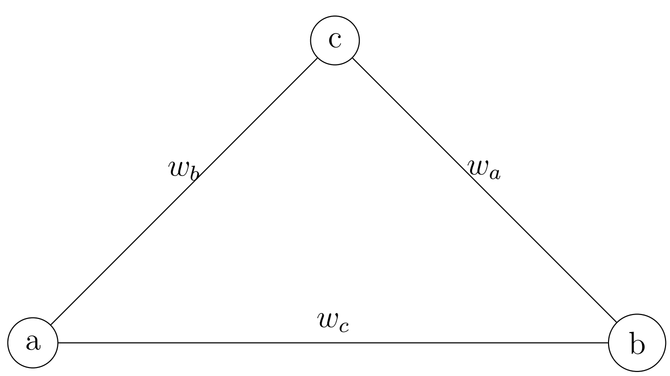



**Figure 2.7**: *Base case for switch identity.*

Consider an arbitrary graph on three vertices with $V = \{a,b,c\}$ and weights $w_a, w_b, w_c$ on the edges $(b,c),(a,c),(a,b)$ respectively. Since we allow for zero weights, this comprises all connected weighted graphs on 3 vertices. By the series and parallel rules for effective conductance,

$$\begin{aligned}\mathrm{Mod}_2(\Gamma(a,b)) &= \frac{1}{\frac{1}{w_a}+\frac{1}{w_b}} + w_c \\ &= \frac{w_a w_b}{w_a + w_b} + w_c \\ &= \frac{w_a w_b + w_a w_c + w_b w_c}{w_a + w_b},\end{aligned} \tag{2.11}$$

$$\mathrm{Mod}_2(\Gamma(c,\{a,b\})) = w_a + w_b, \tag{2.12}$$

$$\begin{aligned}\mathrm{Mod}_2(\Gamma(b,c)) &= \frac{1}{\frac{1}{w_b}+\frac{1}{w_c}} + w_a \\ &= \frac{w_b w_c}{w_b + w_c} + w_a \\ &= \frac{w_b w_c + w_a w_b + w_a w_c}{w_b + w_c},\end{aligned} \tag{2.13}$$

$$\text{and } \ \mathrm{Mod}_2(\Gamma(a,\{b,c\})) = w_b + w_c \tag{2.14}$$

From the above expressions, we have

$$\begin{aligned}\mathrm{Mod}_2(\Gamma(a,b))\,\mathrm{Mod}_2(\Gamma(c,\{a,b\})) &= \frac{w_a w_b + w_a w_c + w_b w_c}{w_a + w_b}\cdot(w_a + w_b) \\ &= w_a w_b + w_a w_c + w_b w_c,\end{aligned}$$

$$\begin{aligned}\text{and } \ \mathrm{Mod}_2(\Gamma(b,c))\,\mathrm{Mod}_2(\Gamma(a,\{b,c\})) &= \frac{w_b w_c + w_a w_b + w_a w_c}{w_b + w_c}\cdot(w_b + w_c) \\ &= w_b w_c + w_a w_b + w_a w_c.\end{aligned}$$

Thus, identity (2.4) holds for any weighted graph on three nodes.

Assume that (2.4) is true for any weighted graph on $k$ vertices where $k \geq 3$. We want to show that this is true for any graph on $k+1$ vertices. Let $G$ be a weighted graph on $k+1$ vertices. Let $x$ be a node that is not equal to either $a, b$ or $c$. By the star-mesh transformation (Lemma 19) we can remove the vertex $x$ by readjusting the weights on the complete graph of $x$'s neighbors to get a graph $G'$ such that $\text{Mod}_2(\Gamma(u,v); G') = \text{Mod}_2(\Gamma(u,v); G)$ for all pairs of vertices $u, v$ in $G'$. Hence by induction, (2.4) is true for all graphs. $\square$

The switch identity (2.4) is useful in when we have three nodes $a, b, c \in V$ and we require the ratio $\frac{\text{Mod}_2(\Gamma(a,b))}{\text{Mod}_2(\Gamma(c,b))}$ or to check whether $\text{Mod}_2(\Gamma(b,a))$ is equal to or bigger than $\text{Mod}_2(\Gamma(b,c))$. (2.4) implies that $\frac{\text{Mod}_2(\Gamma(a,b))}{\text{Mod}_2(\Gamma(c,b))} = \frac{\text{Mod}_2(\Gamma(a,\{b,c\})}{\text{Mod}_2(\Gamma(c,\{a,b\}))}$. The right hand side reduces arithmetic complexity of the 2-modulus as it involves the computation of the 2-modulus in graphs where a pair of nodes are merged which reduces the number of nodes, but at the same time removes any self loops formed after merging. Moreover, the merged node $\{b,c\}$ or $\{b,a\}$ can become a cut vertex, and hence as a result, when computing $\text{Mod}_2(\Gamma(a,\{b,c\})$ or $\text{Mod}_2(\Gamma(c,\{a,b\}))$, it is sufficient to focus on the relevant biconnected component. We illustrate this in the example below.

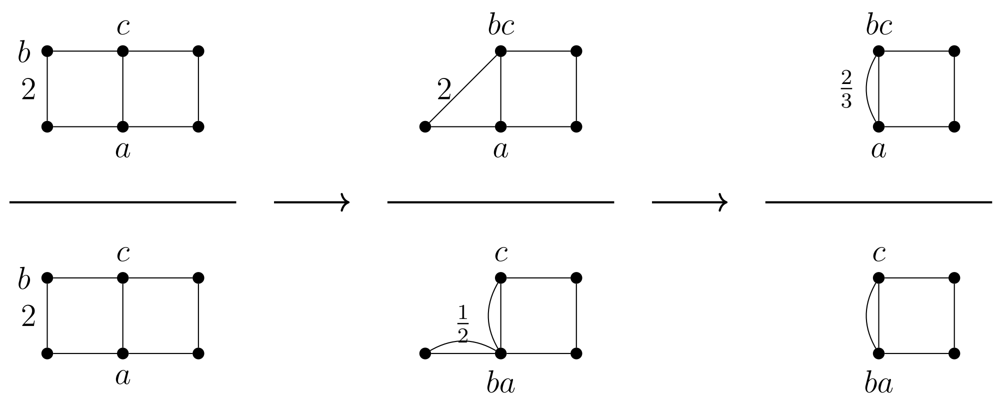



**Figure 2.8**: *Illustration of switch identity for the ladder graph.*

**Example 20.** Consider the ladder graph with two squares as shown in the figure. Unless labelled otherwise, the conductance on the edges is 1. For this problem we ask if $\text{Mod}_2(\Gamma(b,a))$ is bigger, equal to, or smaller than$\text{Mod}_2(\Gamma(b,c))$. Consider the ratio $\frac{\text{Mod}_2(\Gamma(b,a))}{\text{Mod}_2(\Gamma(b,c))}$. By (2.4) this ratio is equal to $\frac{\text{Mod}_2(\Gamma(a,\{b,c\})}{\text{Mod}_2(\Gamma(c,\{a,b\}))}$. The two corresponding graphs with merged nodes is shown in the second column. Using series rule and removing any self-loops the ratio is reduced to computation in the two graphs shown in the third column. It is clear that the denominator is bigger and hence, the ratio is less than 1.

## 2.5 Spectral theory of the Laplacian

Let $G = (V, E)$ be a finite, undirected graph. For $v \in V$, let $d(v)$ denote the degree of $v$. Let $\ell^2(V)$ denote the Hilbert Space of functions $f : V \to \mathbb{R}$ with the standard inner product

$$\langle f, g\rangle := \sum_{v\in V} f(v)g(v)$$

for $f, g \in \ell^2(V)$. Since $V$ is finite $\ell^2(V)$ is the same as $\mathbb{R}^n$. The corresponding norm is given by $\|f\|^2_{\ell^2(V)} = \langle f, f\rangle_{\ell^2(V)} = \sum_{v\in V} f(v)^2$.

For $f : V \to \mathbb{R}$ we have

$$\begin{aligned}\frac{\langle Lf, f\rangle}{\langle f, f\rangle} &= \frac{\sum_{\{u,v\}\in E}(f(u) - f(v))^2}{\sum_{u\in V} f(u)^2}\\ &:= \mathcal{R}_G(f),\end{aligned}$$

where $\mathcal{R}_G(f)$ denotes the Rayleigh quotient of $f$. For the eigenvalues $0 = \lambda_1 \le \lambda_2 \le ... \le \lambda_n$ of $L$, by the Courant–Fischer–Weyl min-max principle, we have

$$\lambda_k = \min_{\substack{f_1,...,f_k\in\ell^2(V)\\ dim(\text{ span}\{f_1,...,f_k\})=k}} \max_{f\neq 0} \left\{\mathcal{R}_G(f) : f \in \text{ span}\{f_1, ..., f_k\}\right\}. \tag{2.15}$$

**Theorem 21.** *[44] Let $G = (V, E)$ be a graph, and let $0 = \lambda_1 \le \lambda_2 \cdots \le \lambda_n$ be the eigenvalues of its Laplacian matrix $L$. Then, $\lambda_2 > 0$ if and only if $G$ is connected.*

**Remark 22.** The second eigenvalue $\lambda_2$ is also called the algebraic connectivity of the graph.

## 2.6 Bounds on eigenvalues from effective resistance

Since effective resistance can be defined in terms of the Laplacian (see Theorem 15), we can expect a relation between the eigenvalues of the Laplacian and the effective resistance (conductance). We begin with the inequality (2.16) on the algebraic connectivity of a graph in terms of the effective resistance diameter: $\max_{s \neq t} \mathcal{R}_{\text{eff}}(s,t) = \frac{1}{\min_{s \neq t} \mathcal{C}_{\text{eff}}(s,t)}$. The following results are implicit in [45] where bounds were stated for the eigenvalues of the random walk Laplacian. We use [45] to make the relationship between the combinatorial Laplacian with effective resistance (conductance) explicit in the following theorems:

**Theorem 23.** *Let $G = (V, E)$ be a graph and let $\lambda_2$ be the second smallest eigenvalue of the Laplacian. Then*

$$\lambda_2 \leq 2 \min_{s \neq t} \mathcal{C}_{\mathit{eff}}(s,t). \tag{2.16}$$

*Proof.* Let $s$ and $t$ be distinct vertices. Assume that $G$ is disconnected. Then $\lambda_2 = 0$. If $s$ and $t$ are in different components of $G$, then $\mathcal{C}_{\text{eff}}(s,t) = 0$, while if $s$ and $t$ are in the same component of $G$, then $\mathcal{C}_{\text{eff}}(s,t) > 0$. Hence, the above inequality is true when $G$ is disconnected.

Now assume that $G$ is connected. Then $\lambda_2 > 0$. Note that $\frac{1}{\lambda_2} = \lambda_{\max}(L^+)$. Since $L^+$ is symmetric

$$\frac{1}{\lambda_2} = \max_{v \neq 0} \frac{v^T L^+ v}{v^T v}. \tag{2.17}$$

Taking $v = \delta_t - \delta_s$ for arbitrary $s \neq t \in V$, we get

$$\begin{aligned} \frac{1}{\lambda_2} &\geq \frac{(\delta_t - \delta_s)^T L^+ (\delta_t - \delta_s)}{(\delta_t - \delta_s)^T (\delta_t - \delta_s)} \\ &= \frac{\mathcal{R}_{\text{eff}}(s,t)}{2}. \end{aligned}$$

Since $\mathcal{R}_{\text{eff}}(s,t) = \frac{1}{\mathcal{C}_{\text{eff}}(s,t)}$, we have

$$\lambda_2 \leq 2\, \mathcal{C}_{\text{eff}}(s,t).$$

Since $s$ and $t$ are arbitrary,

$$\lambda_2 \leq 2 \min_{s \neq t} \mathcal{C}_{\text{eff}}(s,t).$$

$\square$

**Theorem 24.** *Equality holds in (2.16) if and only if either $G$ is a complete graph $K_{n-2}$ connected to two nodes that are adjacent to all vertices of $K_{n-2}$ or $G$ is a complete graph.*

*Proof.* Now, we investigate equality. Let $G$ be a connected graph for which $\lambda_2 = 2\,\mathcal{C}_{\text{eff}}(s,t)$ where $s$ and $t$ are distinct vertices in $G$. First, assume that $s$ and $t$ are not adjacent. Then $\delta_t - \delta_s$ is an eigenvector of $L$ with eigenvalue $\lambda_2$. Note that

$$L(\delta_t - \delta_s) = \deg(t)\delta_t - \deg(s)\delta_s - \sum_{u \sim t, u \nsim s} \delta_u + \sum_{v \sim s, v \nsim t} \delta_v.$$

Hence, $L(\delta_t - \delta_s) = \lambda_2(\delta_t - \delta_s)$ if and only if for every $v \in V$ such that $v \neq s,t$, $v \sim s$ and $v \sim t$, and $\deg(s) = \deg(t) = \lambda_2$. Hence, this degree, call it $d$, is equal to $\lambda_2$. Thus, we must have

$$\mathcal{R}_{\text{eff}}(s,t) = \frac{2}{d}.$$

Let $H$ be the intermediate induced subgraph on $V \backslash \{s,t\}$. If $H$ is not complete, adding edges will strictly decrease $\mathcal{R}_{\text{eff}}(s,t)$. When graph $H$ is complete, we have $\mathcal{R}_{\text{eff}}(s,t) = \frac{2}{d}$. Thus, (2.16) holds only if $H$ is complete.

Now assume that $s$ and $t$ are adjacent. Then $\delta_t - \delta_s$ is an eigenvector of $L$ with eigenvalue $\lambda_2$. Note that

$$L(\delta_t - \delta_s) = (\deg(t)+1)\delta_t - (\deg(s)+1)\delta_s - \sum_{u \sim t, u \nsim s} \delta_u + \sum_{v \sim s, v \nsim t} \delta_v.$$

Hence, $L(\delta_t - \delta_s) = \lambda_2(\delta_t - \delta_s)$ if and only if for every $v \in V$ such that $v \neq s,t$, $v \sim s$ and

$v \sim t$, and $\deg(s) + 1 = \deg(t) + 1 = \lambda_2$. This implies $d := \deg(s) = \deg(t)$ and $\lambda_2$ is equal to $d + 1$. Thus, we must have

$$\mathcal{R}_{\text{eff}}(s, t) = \frac{2}{d + 1}.$$

Let $H$ be the intermediate induced subgraph on $V \backslash \{s, t\}$. If $H$ is not complete, adding edges will strictly decrease $\mathcal{R}_{\text{eff}}(s, t)$. In the case when this graph $H$ is complete, we have $\mathcal{R}_{\text{eff}}(s, t) = \frac{2}{d+1}$. Thus, (2.16) holds only if $G$ is complete. $\square$

The following theorem gives a lower bound on $\lambda_k$ in terms of the effective resistance diameter. Although it was proved in terms of the eigenvalues of the normalized Laplacian, we state the theorem corresponding to [45, Proposition 4.2] in terms of eigenvalues of the combinatorial Laplacian.

**Theorem 25.** *Let $G = (V, E)$ be a graph and let $\lambda_k$ denote the $k^{th}$ smallest eigenvalue of the Laplacian $L$. Then we have the following:*

$$\lambda_k \geq \frac{1}{N} \cdot k \min_{s \neq t} \mathcal{C}_{\mathit{eff}}(s, t).$$

# Chapter 3

# Bounds on Laplacian eigenvalues using strength

This chapter is based on the preprint [46].

## 3.1 Overview and Motivation

In this chapter we derive bounds on the Laplacian eigenvalues using the strength of graphs. This is inspired by the Cheeger inequality and its higher-order versions which give bounds on the expansion of graphs using the normalized Laplacian eigenvalues.

Similar to how $\lambda_2$ measures how easy it is to partition the vertices into two sets with small isoperimetric ratios via the Cheeger inequality, the $k$-th eigenvalue measures how easy it is to partition the vertices into $k$ components with small isoperimetric ratios via the higher-order Cheeger inequality:

**Theorem 26.** *[30, Theorem 1.1.] Let $G = (V, E)$ be a finite graph and let $\tilde{\lambda}_1 \leq \tilde{\lambda}_2 \leq \ldots \leq \tilde{\lambda}_n$ denote the eigenvalues of the normalized Laplacian $\tilde{L} = D^{-1/2} L D^{-1/2}$ where $L$ is the combinatorial Laplacian. Let $\tilde{\rho}_k(G)$ denote the $k$-way expansion constant of $G$ (Definition*

*35). Then we have the following:*

$$\frac{\tilde{\lambda}_k}{2} \leq \rho_k(G) \leq O(k^2)\sqrt{\tilde{\lambda}_k(G)}. \tag{3.1}$$

Note that (3.1) can be interpreted as a special result on upper and lower bounds on eigenvalues of the normalized Laplacian. We will take this viewpoint and try to obtain bounds on the eigenvalues of the combinatorial Laplacian using the strength of graphs defined below.

A *feasible partition* is a partition of the vertex set of the graph, such that each part induces a non-empty connected subgraph. The minimum feasible partition problem [47] is the following optimization problem:

$$\begin{aligned} &\text{minimize} \quad \frac{|E(P)|}{k_P - 1} \\ &\text{subject to} \quad P \text{ is a feasible partition.} \end{aligned} \tag{3.2}$$

We say that $w(P) := \frac{|E_P|}{k_P - 1}$ is the *weight* of the feasible partition $P$.

**Definition 27** (Strength)**.** The *strength of a graph* is the solution to (3.2) and is denoted by $m(G)$.

It was introduced by Gusfield [48] and it originated from the following theorem due to Tutte and Nash-Williams:

**Theorem 28.** *[49, 50] Let $G = (V, E)$ be a graph. A necessary and sufficient condition for $G$ to have $k$ mutually edge-disjoint spanning trees is that*

$$k(c(E \backslash S) - 1) \leq |S| \tag{3.3}$$

*for all subsets $S$ of $E$, where for $A \subset E$, $c(A)$ denote the number of components in the subgraph $(V, A)$. So $c(E \backslash S)$ denotes the number of components in the partition resulting after removing $S$ from the graph.*

**Remark 29.** Note that in (3.3) dividing both sides by $c(E\backslash S) - 1$, we get $k \leq m(G)$, and hence, the graph $G$ has $k$ edge-disjoint spanning trees if and only if $k \leq m(G)$. Hence the maximum number of edge-disjoint spanning trees in a graph is equal to $\lfloor m(G) \rfloor$.

**Definition 30** (Homogenous graphs)**.** [47] A graph is homogeneous if the solution to (3.2) is the trivial partition (partition containing singleton nodes). In this case the strength $m(G) = \frac{|E|}{|V|-1}$.

There are polynomial time algorithms to find the strength of graphs. Cunningham [51] provided an algorithm to compute $m(G)$ with $O(|V|^4|E|)$ or $O(|V|^2|E|^2 log|V|)$ time complexity for weighted graphs. Cunningham considered the optimal attack problem [51] using which he solved the problem of computing the strength. The strength and the edges of a minimum feasible partition can also be deduced from the spanning tree modulus [47]. There is an exact algorithm to compute the spanning tree modulus in Albin et al. [52].

## 3.2 Preliminaries

In the section we give the definition of various isoperimetric constants studied in spectral graph theory. For $S \subset V$, let $\bar{S}$ denote the complement $V \backslash S$.

**Definition 31** (Expansion)**.** Let $G = (V, E)$ be a finite graph. Let $\mathrm{Vol}(S) := \sum_{v \in S} \deg(v)$ denote the volume of $S$. The expansion of $S$ is defined as

$$\phi_G(S) := \frac{|E(S, \bar{S})|}{\mathrm{Vol}(S)}, \tag{3.4}$$

where $\bar{S}$ is the complement of $S$ in $V$.

**Definition 32** (Isoperimetric ratio)**.** Let $G = (V, E)$ be a finite graph. The isoperimetric ratio of $S$ is defined as

$$\rho_G(S) := \frac{|E(S, \bar{S})|}{|S|}, \tag{3.5}$$

where $|S|$ denotes the cardinality of $S$.

While Lee et al. [30] work with expansion (Definition 31) in their paper, we will primarily need the isoperimetric ratio (Definition 32) in order to relate the eigenvalues to strength.

**Definition 33.** (Cheeger constant) Let $G = (V, E)$ be a finite graph. The Cheeger constant of $G$ is defined as

$$h(G) := \min_{X:|X| \leq \frac{|V(G)|}{2}} \frac{|\partial X|}{|X|}. \tag{3.6}$$

**Definition 34** ($k$-Cheeger constant)**.** Let $G = (V, E)$ be a finite graph. The $k$-Cheeger constant of $G$ is defined as

$$h_k(G) := \min_{V_1, V_2, \ldots, V_k} \max_{i=1,2,\ldots,k} \frac{|\partial V_i|}{|V_i|}, \tag{3.7}$$

where the minimum is taken over all feasible partitions $V_1, V_2, \ldots, V_k$ of $V$ with exactly $k$ parts.

**Definition 35** ($k$-way expansion constant)**.** [30] Let $G = (V, E)$ be a finite graph. The $k$-way expansion constant of $G$ is defined as

$$\tilde{\rho}_G(k) = \min_{S_1, S_2, \ldots, S_k} \max\{\phi(S_i) : i = 1, 2, \ldots, k\}, \tag{3.8}$$

where the minimum is taken over all collections of $k$ non-empty, disjoint subsets $S_1, S_2, \ldots, S_k \subseteq V$.

Note that in Definition 34 we partition $V$ into exactly $k$ parts, while in Definition 35 we introduce the $k$-way isoperimetric constant where we look at $k$ disjoint sets as opposed to a proper partition:

**Definition 36** ($k$-way isoperimetric constant)**.** Let $G = (V, E)$ be a finite graph. The $k$-isoperimetric constant of $G$ is defined as

$$\rho_k(G) := \min_{\{S_i\} \in \mathcal{S}_k} \max_{i=1,2,\ldots,k} \frac{|\partial S_i|}{|S_i|} = \min_{\{S_i\} \in \mathcal{S}_k} \max_{i=1,2,\ldots,k} \rho_G(S_i), \tag{3.9}$$

where the minimum is taken over all family $\mathcal{S}_k$ of sets $\{S_i\}_{i=1}^k$, where $\emptyset \neq S_i \subset V$ for every $i = 1, \ldots, k$, and $S_i \cap S_j = \emptyset$ for $i \neq j$.

Now we extend the definition of Rayleigh quotients (see Section 2.5) to vector valued functions as follows:

**Definition 37** (Higher-dimensional Rayleigh quotient)**.** [30] For any $f : V \to \mathbb{R}^m$ we define the Rayleigh quotient of $f$ as

$$\mathcal{R}_G(f) := \frac{\sum_{\{u,v\}\in E} \|f(u) - f(v)\|^2}{\sum_{u\in V} \|f(u)\|^2}.$$

The following lemma states that for the spectral embedding, i.e.,
$F(v) := (f_1(v), f_2(v), ..., f_k(v))$, the Rayleigh quotient is bounded above by $\lambda_k$.

**Lemma 38.** *Let $f_1, f_2, ...., f_k : V \to \mathbb{R}$ be orthonormal vectors in $\ell^2(V)$. Let $F(v) := (f_1(v), f_2(v), ..., f_k(v))$. Then $\mathcal{R}_G(F) \leq \lambda_k$.*

*Proof.* We have

$$\begin{aligned}
\mathcal{R}_G(F) &= \frac{\sum_{\{u,v\}\in E} \|F(u) - F(v)\|^2}{\sum_{v\in V} \|F(v)\|^2} \\
&= \frac{\sum_{\{u,v\}\in E} \sum_{i=1}^k (f_i(u) - f_i(v))^2}{\sum_{v\in V} \sum_{i=1}^k (f_i(v))^2} \\
&= \frac{\sum_{i=1}^k \sum_{\{u,v\}\in E} (f_i(u) - f_i(v))^2}{k} \quad \text{(each } f_i \text{ is a unit vector)} \\
&= \frac{\sum_{i=1}^k \mathcal{R}_G(f_i)}{k} \quad \Big(\mathcal{R}_G(f_i) = \sum_{\{u,v\}\in E} (f_i(u) - f_i(v))^2 \text{ by definition}\Big) \\
&= \frac{\sum_{i=1}^k \lambda_k}{k} \\
&\leq \lambda_k.
\end{aligned}$$

$\square$

## 3.3 Higher-order Cheeger Inequalities

In this section we will prove the following weaker version of the higher-order Cheeger inequality.

**Theorem 39.** *Let $G = (V, E)$ be a finite graph and let $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ denote the eigenvalues of the combinatorial Laplacian $L := D - A$. Let $h_k(G)$ denote the $k$-Cheeger constant of $G$ (Definition 34). Then we have the following:*

$$\frac{\lambda_k}{2} \leq h_k(G) \leq O(k^4)\sqrt{d_{\max}\lambda_k}. \tag{3.10}$$

**Remark 40.** The constant $O(k^4)$ is a universal constant as it does not depend on $n$ or the graph. We also use the notation $h_k(G) \lesssim k^4\sqrt{d_{\max}\lambda_k}$ to denote the dependence on a constant.

This section closely follows [30] and the results there are modified wherever needed to get the corresponding results for the $k$-Cheeger constant (Definition 34) of $G$ as opposed to the $k$-way expansion constant of $G$, and the unnormalized combinatorial Laplacian eigenvalues of $G$ as opposed to the normalized Laplacian eigenvalues of $G$. These modifications cannot be inferred as a special case of results from [30], but as mentioned in Section 3.1, we require bounds on the $k$-Cheeger constant because it relates more directly to the strength of graphs.

The following lemma states that if we can bound the Rayleigh quotients of $k$ disjoint functions, then we can also bound the $k^{\text{th}}$ eigenvalue. Using this lemma we can get the left inequality of (3.10).

**Lemma 41.** *[30, Lemma 2.1] Suppose $\psi_1, \psi_2, ..., \psi_k : V \to \mathbb{R}$ are disjointly supported functions with $\mathcal{R}_G(\psi_i) \leq c$ for each $i = 1, 2, ..., k$. Then, $\lambda_k \leq 2c$.*

*Proof of Lemma.* Let $f = \sum_{i=1}^{k} \alpha_i \psi_i$ be arbitrary. For $u, v \in V$ we have

$$
\begin{aligned}
(f(u) - f(v))^2 &= \Big( \sum_{i=1}^{k} \alpha_i (\psi_i(u) - \psi_i(v)) \Big)^2 \\
&\leq 2 \sum_{i=1}^{k} \alpha_i^2 (\psi_i(u) - \psi_i(v))^2
\end{aligned}
$$

since $(a+b)^2 \leq 2(a^2 + b^2)$ for $a, b \in \mathbb{R}$. Thus,

$$
\begin{aligned}
\mathcal{R}_G(f) &= \frac{\sum_{\{u,v\} \in E} (f(u) - f(v))^2}{\sum_{v \in V} f(v)^2} \\
&\leq \frac{2 \sum_{i=1}^{k} \alpha_i^2 \sum_{\{u,v\} \in E} (\psi_i(u) - \psi_i(v))^2}{\sum_{i=1}^{k} \alpha_i^2 \sum_{v \in V} \psi_i(v)^2} \\
&= 2 \sum_{i=1}^{k} \frac{\alpha_i^2 \sum_{\{u,v\} \in E} (\psi(u) - \psi(v))^2}{\alpha_i^2 \sum_{v \in V} \psi_i(v)^2} \cdot \frac{\alpha_i^2 \sum_{v \in V} \psi_i(v)^2}{\sum_{i=1}^{k} \alpha_i^2 \sum_{v \in V} \psi_i(v)^2} \\
&\leq 2c \sum_{i=1}^{k} \frac{\alpha_i^2 \sum_{v \in V} \psi_i(v)^2}{\sum_{i=1}^{k} \alpha_i^2 \sum_{v \in V} \psi_i(v)^2} \\
&\leq 2c.
\end{aligned}
$$

□

Now in Lemma 41, we take the characteristic functions of $V_i$ for the minimum feasible partition $V_1, V_2, \ldots, V_k$ of $V$ solving the minimization problem (Definition 34):

$$
h_k(G) := \min_{V_1, V_2, \ldots, V_k} \max_{i=1,2,\ldots,k} \frac{|\partial V_i|}{|V_i|}.
$$

Note that $\mathcal{R}(\mathbb{1}_{V_i}) = \frac{|\partial V_i|}{|V_i|}$. Hence we have $\mathcal{R}(\mathbb{1}_{V_i}) \leq h_k(G)$. Thus, by Lemma 41, $\lambda_k \leq 2h_k(G)$.

Below we give an alternative proof to the left hand inequality in (3.10):

**Proposition 42.** *If $G$ is a finite graph,*

$$
2h_k(G) \geq \lambda_k(G).
$$

*Proof.* Let $\{V_1, V_2, \ldots, V_k\}$ be a partition of the nodes of $G$. Up to relabeling, the Laplacian can be block decomposed as

$$L = \begin{bmatrix} D_{11} & & & & \\ & D_{22} & & & \\ & & D_{33} & & \\ & & & \ddots & \\ & & & & D_{kk} \end{bmatrix} - \begin{bmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{1k} \\ A_{12}^T & A_{22} & A_{23} & \cdots & A_{2k} \\ A_{13}^T & A_{23}^T & A_{33} & \cdots & A_{3k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{1k}^T & A_{2k}^T & A_{3k}^T & \cdots & A_{kk} \end{bmatrix}.$$

Consider the $k$-dimensional subspace of $\mathbb{R}^V$ spanned by $\{\mathbb{1}_{V_1}, \mathbb{1}_{V_2}, \ldots \mathbb{1}_{V_k}\}$. Maximizing $x^T L x$ over all normal vectors in this subspace will provide an upper bound on $\lambda_k(L)$. Let $x = \sum_i c_i \mathbb{1}_{V_i}$ be a vector in the subspace. Then

$$x^T x = \sum_i c_i^2 |V_i|,$$

and

$$x^T L x = \sum_{i,j} c_i c_j \mathbb{1}_{V_i}^T L \mathbb{1}_{V_j}.$$

Note that

$$\mathbb{1}_{V_i}^T L \mathbb{1}_{V_i} = \mathbf{1}^T D_{ii} \mathbf{1} - \mathbf{1}^T A_{ii} \mathbf{1} = \sum_{v \in V_i} \deg(v) - 2|E(V_i)| = |\partial V_i|$$

and, when $i \neq j$,

$$\mathbb{1}_{V_i}^T L \mathbb{1}_{V_j} = -\mathbf{1}^T A_{ij} \mathbf{1} = -|E(V_i, V_j)|.$$

Here, $E(V_i)$ denotes the set of edges within the component $V_i$ and $E(V_i, V_j)$ denotes the set of edges connecting $V_i$ to $V_j$. In this way, the optimization problem we wish to solve can be written in the form

$$\begin{aligned} \text{maximize} \quad & c^T Q c \\ \text{subject to} \quad & c^T P c = 1, \end{aligned}$$

with $P = \operatorname{diag}(|V_i|)$ and

$$Q_{ij} = \begin{cases} |\partial V_i| & \text{if } i = j, \\ -|E(V_i, V_j)| & \text{if } i \neq j. \end{cases}$$

The matrix $Q$ can be interpreted as the Laplacian matrix for the multigraph $G'$ formed by collapsing each of the sets $V_i$ into a single node and discarding any resulting loops. The corresponding normalized Laplacian is

$$\tilde{Q} := D^{-\frac{1}{2}} Q D^{-\frac{1}{2}} \qquad \text{where} \qquad D = \operatorname{diag}(|\partial V_i|).$$

Now, we need the following lemma:

**Lemma 43.** *[53] Let $G = (V, E)$ be a graph. Let $\tilde{L} := D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$ denote the normalized Laplacian of $G$. Let $\tilde{\lambda}_1 \leq \tilde{\lambda}_2 \leq \ldots \leq \tilde{\lambda}_n$ denote the eigenvalues of $\tilde{L}$. Then for every $1 \leq k \leq n$ we have*

$$\tilde{\lambda}_k \leq 2$$

*with $\tilde{\lambda}_k = 2$ if and only if a connected component of $G$ is bipartite and nontrivial.*

Using Lemma 43, we observe that,for any vector $c$ satisfying $\|P^{\frac{1}{2}} c\|^2 = c^T P c = 1$,

$$\begin{aligned} c^T Q c &= (D^{\frac{1}{2}} c)^T \tilde{Q} (D^{\frac{1}{2}} c) \\ &\leq 2\|D^{\frac{1}{2}} c\|^2 \\ &= 2\|D^{\frac{1}{2}} P^{-\frac{1}{2}} P^{\frac{1}{2}} c\|^2 \\ &\leq 2\|D^{\frac{1}{2}} P^{-\frac{1}{2}}\|^2. \end{aligned}$$

The matrix $D^{\frac{1}{2}}P^{-\frac{1}{2}}$ is a diagonal matrix with entries

$$\left(D^{\frac{1}{2}}P^{-\frac{1}{2}}\right)_{ii} = \sqrt{\frac{|\partial V_i|}{|V_i|}},$$

yielding the upper bound

$$c^T Q c \leq 2 \max_i \frac{|\partial V_i|}{|V_i|} = 2h_k.$$

The result follows. □

The rest of this section is devoted to proving the right hand inequality in (3.10).

### 3.3.1 Disjoint sets to disjoint functions

We begin with the following lemma which was proved by Chung [53] for the expansion of a graph (see [30]). We prove it for the isoperimetric ratio.

**Lemma 44.** *Let $d_{\max}$ denote the maximum degree in $G$. For any $\psi : V \to \mathbb{R}^m$, there exists a subset $S \subset supp(\psi)$ with*

$$\rho_G(S) \leq \sqrt{2d_{\max}\mathcal{R}_G(\psi)}. \tag{3.11}$$

*Proof.* If $\text{supp}(\psi) = V$, we can take $S = V$ and the inequality is true as $\rho_G(S) = 0$ in this case. So we assume that $\text{supp}(\psi) \neq V$. For $t \in [0, \infty)$, define $S_t = \{u \in V : ||\psi(u)||^2 > t\}$. We have

$$\begin{aligned}
\int_0^\infty |S_t| dt &= \int_0^\infty \sum_{v \in V} \mathbf{1}_{v \in S_t} dt \\
&= \sum_{v \in V} \int_0^\infty \mathbf{1}_{\{\|\psi(v)\|^2 > t\}} dt \\
&= \sum_{v \in V} \int_0^{\|\psi(v)\|^2} 1 dt \\
&= \sum_{v \in V} \|\psi(v)\|^2 .
\end{aligned}$$

Also, without loss of generality assuming that we order $\{u, v\} \in E$, such that $\|\psi(u)\|^2 \geq \|\psi(v)\|^2$, we get

$$\begin{aligned}
\int_0^\infty |E(S_t, \bar{S}_t|)dt &= \int_0^\infty \sum_{\{u,v\}\in E} \mathbf{1}_{u\in S_t}\mathbf{1}_{v\in \bar{S}_t} dt \\
&= \sum_{\{u,v\}\in E} \int_0^\infty \mathbf{1}_{\{\|\psi(u)\|^2>t\}}\mathbf{1}_{\{\|\psi(v)\|^2<t\}} dt \\
&= \sum_{\{u,v\}\in E} \|\psi(u)\|^2 - \|\psi(v)\|^2 \\
&= \sum_{\{u,v\}\in E} \langle\psi(u), \psi(u)\rangle - \langle\psi(u), \psi(v)\rangle + \langle\psi(u), \psi(v)\rangle - \langle\psi(v), \psi(v)\rangle \\
&= \sum_{\{u,v\}\in E} \langle\psi(u) - \psi(v), \psi(u) + \psi(v)\rangle \\
&\leq \sum_{\{u,v\}\in E} \|\psi(u) - \psi(v)\| \quad \|\psi(u) + \psi(v)\| \\
&\leq \sqrt{\sum_{\{u,v\}\in E} \|\psi(u) - \psi(v)\|^2}\sqrt{\sum_{\{u,v\}\in E} \|\psi(u) + \psi(v)\|^2} \\
&\leq \sqrt{\sum_{\{u,v\}\in E} \|\psi(u) - \psi(v)\|^2}\sqrt{2\sum_{\{u,v\}\in E} \|\psi(u)\|^2 + \|\psi(v)\|^2} \\
&\leq \sqrt{\sum_{\{u,v\}\in E} \|\psi(u) - \psi(v)\|^2}\sqrt{2d_{\max}\sum_{v\in V} \|\psi(v)\|^2}.
\end{aligned}$$

From the above we have

$$\frac{\int_0^\infty |E(S_t, \bar{S}_t)|dt}{\int_0^\infty |S_t|dt} \leq \sqrt{2d_{\max}\mathcal{R}_G(\psi)}.$$

Hence,

$$\int_0^\infty |E(S_t, \bar{S}_t)| - |S_t|\sqrt{2d_{\max}\mathcal{R}_G(\psi)}dt \leq 0.$$

This implies that there exists a $\bar{t} \in [0, \infty]$ such that

$$|E(S_t, \bar{S}_{\bar{t}})| - |S_{\bar{t}}|\sqrt{2d_{\max}\mathcal{R}_G(\psi)} \leq 0$$

and hence,

$$\frac{|E(S_{\bar{t}}, \bar{S}_{\bar{t}})|}{|S_{\bar{t}}|} \leq \sqrt{2d_{\max}\mathcal{R}_G(\psi)}.$$

Thus (3.11) follows. □

### 3.3.2 Random partitions of metric spaces

The following results are referenced from [30]. Let $(X, d)$ be a finite metric space. Let $B(x, R) = \{y \in X : d(x, y) \leq R\}$ denote the closed ball of radius $R$ about $x$. We write a partition $P$ of $X$ as a function $P : X \to 2^X$ mapping a point $x \in X$ to the unique set in $2^X$ that contains $x$.

For $\Delta > 0$, we say that $P$ is $\Delta$-bounded if $\operatorname{diam}(S) \leq \Delta$ for every $S \in P$. We will consider distributions over random partitions. If $\mathcal{P}$ is a random partition of $X$, we say that $\mathcal{P}$ is $\Delta$-bounded if this property holds with probability one.

A random partition $\mathcal{P}$ is $(\Delta, \alpha, \delta)$-padded if $\mathcal{P}$ is $\Delta$-bounded, and for every $x \in X$, we have

$$\mathbb{P}(B(x, \Delta/\alpha) \subset \mathcal{P}(x)) \geq \delta.$$

We need the following theorem:

**Theorem 45** ([54, 55])**.** *If $X \subset \mathbb{R}^k$, then for every $\Delta > 0$ and $\delta > 0$, $X$ admits a $(\Delta, O(k/\delta), 1 - \delta)$-padded random partition.*

### 3.3.3 Localizing eigenfunctions

Let $F : V \to \mathbb{R}^m$ be a function.

**Definition 46** (Radial projection distance)**.** The radial projection distance is defined as

$$d_F(u,v) := \left\| \frac{F(u)}{\|F(u)\|} - \frac{F(v)}{\|F(v)\|} \right\|.$$

**Definition 47** (Spreading function)**.** Let $F : V \to \mathbb{R}^m$ be a function. $F$ is called $(\Delta, \eta)$ spreading if for every $S \subset V$, $\mathrm{diam}_{d_F}(S) \leq \Delta$ implies that $\sum_{u \in S} \|F(u)\|^2 \leq \eta \sum_{u \in V} \|F(u)\|^2$.

**Lemma 48.** *Let $F : V \to \mathbb{R}^m$. For all $u, v \in V$,*

$$d_F(u,v) \|F(u)\| \leq 2 \|F(u) - F(v)\|.$$

*Proof.* We have

$$\begin{aligned} d_F(u,v) \|F(u)\| &= \left\| F(u) - \|F(u)\| \frac{F(v)}{\|F(v)\|} \right\| \\ &\leq \|F(u) - F(v)\| + \left\| F(v) - \|F(u)\| \frac{F(v)}{\|F(v)\|} \right\| \\ &\leq 2 \|F(u) - F(v)\|. \end{aligned}$$

□

**Lemma 49.** *[30] Let $f_1, f_2, ...., f_k : V \to \mathbb{R}$ be orthonormal vectors in $\ell^2(V)$. Let $F(v) := (f_1(v), f_2(v), ..., f_k(v))$. For all $\Delta > 0$, $F$ is $\left(\Delta, \frac{1}{k(1-\Delta^2)}\right)$ spreading.*

*Proof.* For $x \in \mathbb{R}^k$ a unit vector, we define $U : \mathbb{R}^k \to \ell^2(V)$ as

$$(Ux)(v) := \sum_{i=1}^{k} x_i f_i(v).$$

Note that $U$ maps the standard orthonormal basis to $f_1, f_2, ..., f_k$ which is an orthonormal system in $\ell^2(V)$, and hence $U$ is unitary. Thus,

$$\sum_{v\in V} \langle x, F(v)\rangle^2 = \langle Ux, Ux\rangle = 1.$$

Let $S \subset V$ such that $\operatorname{diam}_{d_F}(S) \leq \Delta$. For any $u \in S$ we have

$$\begin{aligned}
1 &= \sum_{v\in V} \left\langle F(v), \frac{F(u)}{\|F(u)\|}\right\rangle^2 \\
&= \sum_{v\in V} \|F(v)\|^2 \left\langle \frac{F(v)}{\|F(v)\|}, \frac{F(u)}{\|F(u)\|}\right\rangle^2 \\
&= \sum_{v\in V} \|F(v)\|^2 \left(\frac{2 - d_F(u,v)^2}{2}\right)^2 \\
&\geq \sum_{v\in S} \|F(v)\|^2 \left(\frac{2 - d_F(u,v)^2}{2}\right)^2 \\
&\geq \sum_{v\in S} \|F(v)\|^2 \left(1 - \frac{\Delta^2}{2}\right)^2 \\
&\geq \sum_{v\in S} \|F(v)\|^2 \left(1 - \Delta^2\right) \quad \text{since } \left(1 - \frac{x^2}{2}\right)^2 \geq 1 - x^2.
\end{aligned} \tag{3.12}$$

We also have

$$\begin{aligned}
\sum_{v\in V} \|F(v)\|^2 &= \sum_{v\in V}\sum_{i=1}^{k} f_i(v)^2 \\
&= \sum_{i=1}^{k} \|f_i\|^2 \\
&= k.
\end{aligned} \tag{3.13}$$

Hence, from (3.12) and (3.13) we get

$$\frac{1}{k\Big(1-\Delta^2\Big)} \sum_{v\in V} \|F(v)\|^2 \geq \sum_{v\in S} \|F(v)\|^2$$

. ∎

Now we proceed to localizing the partitions by constructing functions analogous to characteristic functions of components in a partition. We consider the following:

**Definition 50** (Induced shortest-path metric)**.** For a graph $G = (V, E)$ we defined the metric $\hat{d}_F$ to be the induced shortest-path metric on $G$, where the length of an edge $\{u, v\}$ is given by $d_F(u, v)$. Since $d_F$ is a pseudo-metric (Definition 51 ), $\hat{d}_F \geq d_F$. We define

$$N_\epsilon(S, \hat{d}_F) := \{v \in V : \hat{d}_F(v, S) < \epsilon\}.$$

**Definition 51** (Pseudo-metric)**.** Let $X$ be a set. A non-negative function $X \times X \to \mathbb{R}$ is a pseudo-metric is it satisfies the following:

1. For all $x \in X$, $d(x, x) = 0$.
2. For all $x, y \in X$, $d(x, y) = d(y, x)$.
3. For all $x, y, z \in X$, $d(x, z) + d(z, y) \leq d(x, y)$.

The following lemma constructs the corresponding disjoint function for each subset $S \subset V$:

**Lemma 52.** *[30, Lemma 3.3] Let $F : V \to \mathbb{R}^m$. For every subset $S \subset V$ and $\epsilon > 0$, there exists a mapping $\psi : V \to \mathbb{R}^m$ such that*

1. *$\psi|_S = F|_S$.*
2. *supp$(\psi) \subset N_\epsilon(S, \hat{d}_F)$.*
3. *For $\{u, v\} \in E$, $\|\psi(u) - \psi(v)\| \leq (1 + \frac{2}{\epsilon}) \|F(u) - F(v)\|$.*

**Lemma 53.** *[30, Lemma 3.4] Let $F : V \to \mathbb{R}^m$. Suppose that for $\beta, \delta > 0$ and $r \in \mathbb{N}$, there exist $r$ disjoint subsets $T_1, T_2, ..., T_r \subset V$ such that $\hat{d}(_F T_i, T_j) \geq \beta$ for $i \neq j$, and for every $i = 1, 2, ..., r$,*

$$\sum_{v \in T_i} \|F(v)\|^2 \geq \delta \sum_{v \in V} \|F(v)\|^2 .$$

*Then there exist disjointly supported functions* $\psi_1, \psi_2, ..., \psi_r : V \to \mathbb{R}$ *such that*

$$\mathcal{R}_G(\psi_i) \leq \frac{2}{\delta(r-i+1)}\Big(1+\frac{4}{\beta}\Big)^2 \mathcal{R}_G(F).$$

**Lemma 54.** *[30, Lemma 3.5] Let* $r, k \in \mathbb{N}$ *be such that* $\frac{k}{2} \leq r \leq k$*, and suppose that the map* $F : V \to \mathbb{R}^m$ *is* $(\Delta, \frac{1}{k} + \frac{k-r+1}{8kr})$*-spreading for some* $\Delta > 0$*. Suppose additionally there is a random partition* $\mathcal{P}$ *with the following properties:*

1. *For every* $S \in \mathcal{P}$*, diam*$(S, d_F) \leq \Delta$*.*

2. *For every* $v \in V$*,* $\mathbb{P}[B_{\hat{d}_F}(v, \frac{\Delta}{\alpha}) \subset \mathcal{P}(v)] \geq 1 - \frac{k-r+1}{4r}$*.*

*Then there exist* $r$ *disjoint subsets* $T_1, T_2, ..., T_r \subset V$ *such that for each* $i \neq j$*, we have* $\hat{d}_F(T_i, T_j) \geq 2\frac{\Delta}{\alpha}$*, and for every* $i = 1, 2, ..., k$*,*

$$\sum_{v \in T_i} \|F(v)\|^2 \geq \frac{1}{2k} \sum_{v \in V} \|F(v)\|^2 .$$

**Corollary 55.** *[30, Corollary 3.6] Let* $k \in \mathbb{N}$ *and* $\delta \in (0, 1)$ *be given. Suppose the map* $F : V \to \mathbb{R}^m$ *is* $(\Delta, \frac{1}{k} + \frac{\delta}{48k})$*- spreading for some* $\Delta \leq 1$*, and there is a random partition* $\mathcal{P}$ *with the following properties:*

1. *For every* $S \in \mathcal{P}$*, diam*$(S, d_F) \leq \Delta$*.*

2. *For every* $v \in V$*,* $\mathbb{P}[B_{\hat{d}_F}(v, \frac{\Delta}{\alpha}) \subset \mathcal{P}(v)] \geq 1 - \frac{\delta}{24}$*.*

*Then there are at least* $r \geq \lceil (1-\delta)k \rceil$ *disjointly supported functions* $\psi_1, \psi_2, ..., \psi_r : V \to \mathbb{R}$ *such that*

$$\mathcal{R}_G(\psi_i) \lesssim \frac{\alpha^2}{\delta \Delta^2} \mathcal{R}_G(F).$$

### 3.3.4 Cheeger Inequalities

The following theorem and its proof follow [30, Theorem 3.7], with modifications to fit the context of combinatorial Laplacian eigenvalues.

**Theorem 56.** *For any $\delta \in (0,1)$, and graph $G = (V, E)$, there exist $r \geq \lceil (1-\delta)k \rceil$ disjointly supported functions $\psi_1, \psi_2, ..., \psi_r : V \to \mathbb{R}$ such that*

$$\mathcal{R}_G(\psi_i) \lesssim \frac{k^2}{\delta^4} \lambda_k \tag{3.14}$$

*where $\lambda_k$ is the $k$th smallest eigenvalue of $L$.*

*Proof.* Let $f_1, f_2, \dots, f_k$ be an orthonormal system of eigenvectors for the first $k$ eigenvalues of the Laplacian $L$ and for $v \in V$, define $F(v) = (f_1(v), f_2(v), \dots, f_k(v))$.

Choose $\Delta \asymp \sqrt{\delta}$ so that $\frac{1}{1-\Delta^2} \leq 1 + \frac{\delta}{48}$. From Lemma 49, $F$ is $(\Delta, 1 + \frac{\delta}{48})$ spreading. From Theorem 45, $(F(V), d_F)$ admits a $(\Delta, O(k/\delta), 1 - \delta/24)$-padded random partition $\mathcal{P}$, i.e., $\mathcal{P}$ is $\Delta$-bounded and for ever $v \in V$,

$$\mathbb{P}(B(x, \Delta/\alpha) \subset \mathcal{P}(x)) \geq 1 - \delta/24$$

for $\alpha \asymp \frac{k}{\delta}$. Since $\hat{d}_F \geq d_F$, $B_{\hat{d}_F}(v, \Delta/\alpha) \subset B_{d_F}(v, \Delta/\alpha)$, and hence we have the required assumptions for Corollary 55. Hence,

$$\mathcal{R}_G(\psi_i) \lesssim \frac{k^2}{\delta^2 \cdot \delta \cdot \delta} \mathcal{R}_G(F)$$

Hence by Lemma 38 we get (3.14). $\square$

The following theorem and its proof follow [30, Theorem 3.8], with modifications to fit the context of combinatorial Laplacian eigenvalues.

**Theorem 57.** *For any graph $G = (V, E)$, there exists a partition $V = S_1 \cup S_2 \cup ... \cup S_k$ such* *that*

$$\rho_G(S_i) \lesssim k^4 \sqrt{d_{\max}\lambda_k}$$

*where $\lambda_k$ is the $k$th smallest eigenvalue of $L$ and $d_{\max}$ is the maximum degree in $G$.*

*Proof.* We use $\delta = \frac{1}{2k}$ in Theorem 56 to get $\psi_1, \psi_2, ..., \psi_r : V \to \mathbb{R}$ such that

$$\mathcal{R}_G(\psi_i) \lesssim 16k^2k^4\lambda_k \lesssim k^6\lambda_k \tag{3.15}$$

Using Lemma 44 we can find sets $S_1 \cup S_2 \cup ... \cup S_k$ with $S_i \subset \operatorname{supp}(\psi_i)$ and $\rho_G(S_i) \leq \sqrt{2d_{\max}\mathcal{R}_G(\psi_i)} \lesssim k^3\sqrt{d_{\max}\lambda_k}$ for each $i = 1, 2, \ldots, k$.

Now we reorder the sets so that $|S_i| \leq |S_2| \leq \ldots |S_k|$ and form the set $S_k^{'} = V \backslash (S_1 \cup S_2 \cup ... \cup S_{k-1})$ so that $V = S_1 \cup S_2 \cup \ldots \cup S_k^{'}$. Now note that

$$\begin{aligned}
\rho_G(S_k^{'}) &= \frac{|E(S_k^{'}, \bar{S_k^{'}})|}{|S_k^{'}|} \\
&\leq \frac{\sum_{i=1}^{k-1} |E(S_i, \bar{S}_i)|}{|S_k^{'}|} \\
&\leq k \max_{i=1}^{k-1} \rho_G(S_i) \\
&\lesssim k^4\sqrt{d_{\max}\lambda_k}
\end{aligned}$$

□

The right inequality of (3.10) follows from Theorem 57.

## 3.4 Lower bounds for $m_k(G)$

In this section, we bound the weights of the feasible partitions of a graph from below with its Laplace eigenvalues.

**Theorem 58.** *Let $k \geq 2$ be an integer. If $\lambda_k(G)$ is the $k$-th eigenvalue of the combinatorial Laplacian of a finite graph $G = (V, E)$, which is $r$-edge-connected, then*

$$m_k(G) \geq \frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2}. \tag{3.16}$$

*It follows that*

$$m(G) \geq \min_{k=2,\ldots,N} \frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2}. \tag{3.17}$$

*Moreover, if equality holds in (3.16) then $G$ is an $r$-regular, $r$-edge connected, bipartite graph.*

*Proof of Theorem 58.* Let $\{V_1, V_2, \ldots, V_k\}$ be a partition of the nodes of $G$. We need to find an upper bound on $\lambda_k$ using this partition. So following the proof of Theorem 42 we can write $\lambda_k$ as the solution to the optimization problem:

$$\begin{aligned} &\text{maximize} \quad c^T Q c \\ &\text{subject to} \quad c^T P c = 1, \end{aligned}$$

with $P = \operatorname{diag}(|V_i|)$ and

$$Q_{ij} = \begin{cases} |\partial V_i| & \text{if } i = j, \\ -|E(V_i, V_j)| & \text{if } i \neq j. \end{cases}$$

Introducing the change of variables $c = P^{-1/2} d$ transforms the problem into

$$\begin{aligned} &\text{maximize} \quad d^T \tilde{Q} d \\ &\text{subject to} \quad d^T d = 1, \end{aligned}$$

with

$$\tilde{Q}_{ij} = \begin{cases} \frac{|\partial V_i|}{|V_i|} & \text{if } i = j, \\ -\frac{|E(V_i, V_j)|}{\sqrt{|V_i|\,|V_j|}} & \text{if } i \neq j. \end{cases}$$

From here, we see that the goal is to find an upper bound on $\lambda_{\max}$, the largest eigenvalue of $\tilde{Q}$.

Gershgorin's theorem shows that

$$\lambda_{\max} \le \max_i \left\{ \frac{|\partial V_i|}{|V_i|} + \sum_{j \ne i} \frac{|E(V_i, V_j)|}{\sqrt{|V_i|\,|V_j|}} \right\} \le 2 \max_i |\partial V_i|. \tag{3.18}$$

Since the partition was arbitrary, this shows that

$$\lambda_k = \lambda_k(L) \le 2 \min_{\{V_1,\ldots,V_k\}} \max_i |\partial V_i|, \tag{3.19}$$

where the minimum is taken over all partitions of $V$ into $k$ nonempty subsets.

Now, suppose $G$ is $r$-edge-connected and let $P = \{V_1, \ldots, V_k\}$ be a minimum feasible partition satisfying

$$\frac{|E_P|}{k-1} = m_k(G).$$

Note that

$$2|E_P| = \sum_i |\partial V_i|,$$

and, by the connectivity hypothesis, $|\partial V_i| \ge r$ for all $i$. Thus, for any $j$,

$$|\partial V_j| = 2|E_P| - \sum_{j \ne i} |\partial V_j| \le 2|E_P| - (k-1)r. \tag{3.20}$$

So, from (3.19) and (3.20),

$$\frac{\lambda_k}{2} \le \max_i |\partial V_i| \le 2|E_P| - (k-1)r.$$

Hence dividing by $2(k-1)$ and rearranging we get

$$m_k(G) \ge \frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2}.$$

Minimizing over $k$ yields

$$m(G) \geq \min_{k=2,\ldots,N} \frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2}.$$

Now we show the equality condition in (3.16). Assume that $G$ is a $r$-edge connected graph with Laplacian eigenvalues $\lambda_1, \ldots, \lambda_n$, such that for some $2 \leq k \leq n$, there exists a minimum feasible partition $P = \{V_1, \ldots, V_k\}$ with $\frac{|E_P|}{k-1} = m_k(G)$ and

$$m_k(G) = \frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2}.$$

Then $\frac{\lambda_k}{2} = 2|E_P| - (k-1)r$ and by (3.20), $\lambda_k \geq 2|\partial V_i|$ for all $i$. But by (3.19), $\lambda_k \leq 2\max_i |\partial V_i|$. Hence, $\lambda_k = 2\max |\partial V_i|$. But equality holds in the second inequality of (3.18) only if $|V_i| = 1$ for all $i$. Thus, $k = n$ and the graph is homogeneous. Now equality holds in (3.20) only if degree of each vertex is equal to $r$. Thus, we have $\lambda_n = 2r$ and we know that this implies the graph is bipartite by Lemma 43.

On the other hand, if the graph $G$ is an $r$-regular, $r$-edge connected, bipartite graph, then $G$ is homogeneous since for any feasible partition $P$ with $k$ components,

$$\frac{|E(P)|}{k-1} \leq \frac{kr}{2(k-1)},$$

and since $\frac{k}{k-1}$ is decreasing, $\frac{|E(P)|}{k-1}$ is minimized for $k = n$. Hence $G$ is homogeneous. Now we have

$$\begin{aligned} \frac{\lambda_n(G)}{4(n-1)} + \frac{r}{2} &= \frac{2r}{4(n-1)} + \frac{r}{2} \\ &= \frac{nr}{2(n-1)}. \end{aligned}$$

Since $G$ is homogenous, $m(G) = \frac{nr}{2(n-1)}$. Thus $m_n(G) = \frac{\lambda_n(G)}{4(n-1)} + \frac{r}{2}$ in this case. $\square$

**Remark 59.** Note that equality in (3.16) does not imply equality in (3.17). Consider the cube graph $Q_3$ which is 3-regular and bipartite. The minimum of the right hand side in (3.17) is achieved when $k = 4$ and $k = 7$ (see Figure 3.1), but equality holds in (3.16) when

$k = 8$.

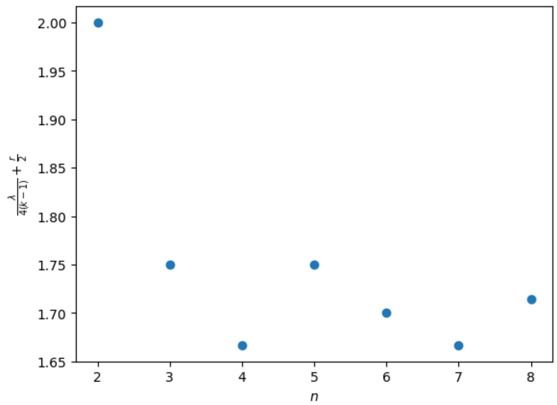



**Figure 3.1**: $\frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2}$ *for the cube graph.*

**Remark 60.** For the sake of comparison with other inequalities involving eigenvalues of the Laplacian we restate (3.17) as an upper bound on $\lambda_k$:

$$\min_{k=2,\ldots,N} \frac{\lambda_k(G)}{4(k-1)} \leq m(G) - \frac{r}{2}. \tag{3.21}$$

**Remark 61.** While proving the equality condition in Theorem 58 we also proved that a $d$-regular, $d$-edge connected graph is homogeneous. Since a $d$-regular, $d$-vertex connected graph is $d$-edge connected, this gives a new proof of [47, Theorem 6.3].

## 3.5 Upper Bounds for $m_k(G)$

First we will prove the following theorem:

**Theorem 62.** *For every finite, connected graph $G$, there is a universal constant $\eta$ such that*

$$m_k(G) \leq \eta \cdot \frac{|V|}{2(k-1)} \cdot k^4 \sqrt{d_{\max} \lambda_k(G)} \tag{3.22}$$

*for all $k \in \mathbb{N}$ with $2 \leq k \leq |V|$. Hence taking minimum on both sides, there is a universal constant $\eta$ such that*

$$m(G) \leq \eta \cdot |V| \cdot \sqrt{d_{\max} \lambda_2(G)}. \tag{3.23}$$

In order to prove this theorem, we will need the following lemma which gives an upper bound on $m_k(G)$ in terms of $h_k(G)$.

**Lemma 63.** *If $G$ is a finite, connected graph, then*

$$m_k(G) \leq \frac{|V|}{2} \frac{h_k(G)}{k-1}. \tag{3.24}$$

*Proof of Lemma 63.* Let $P = \{V_1, V_2, \ldots, V_k\}$ be an optimal partition for (Definition 34) so that

$$h_k(G) = \max_{i=1,\ldots,k} \frac{|\partial V_i|}{|V_i|}.$$

Then, for each $i = 1, 2, \ldots, k$, we have

$$|\partial V_i| \leq h_k(G)|V_i|.$$

Therefore, we have that

$$2|E_P| = \sum_{i=1}^{k} |\partial V_i| \leq \sum_{i=1}^{k} h_k(G)|V_i| = h_k(G)|V|.$$

It follows that

$$m_k(G) \leq \frac{|V|}{2} \frac{h_k(G)}{k-1}.$$

□

*Proof of Theorem 62.* From Lemma 63

$$m_k(G) \leq \frac{|V|}{2} \frac{h_k(G)}{k-1}.$$

From Theorem 57, $h_k(G) \leq \eta k^4 \sqrt{d_{\max}\lambda_k}$ for some universal constant $\eta$. Hence, we get

$$m_k(G) \leq \eta \cdot \frac{|V|}{2(k-1)} \cdot k^4 \sqrt{d_{\max}\lambda_k(G)}.$$

Taking minimum with respect to $k = 2, \ldots, |V|$, we get

$$m(G) \leq \min_{k=2,\ldots,N} \eta \cdot \frac{|V|}{k(k-1)} \cdot 2^4 \sqrt{d_{\max}\lambda_k(G)}.$$

Since $\frac{k^4}{k-1}\sqrt{\lambda_k(G)}$ is increasing with respect to $k$, we have

$$\begin{aligned} m(G) &\leq \eta \cdot \frac{|V|}{2(2-1)} \cdot 2^4 \sqrt{d_{\max}\lambda_k(G)} \\ &\leq \tilde{\eta} \cdot |V| \cdot \sqrt{d_{\max}\lambda_2(G)} \end{aligned}$$

for some universal constant $\tilde{\eta}$. $\square$

**Remark 64.** For the sake of comparison with other similar bounds, we rewrite (3.23) as a lower bound on the algebraic connectivity $\lambda_2$:

$$\frac{m(G)^2}{\eta^2 \cdot |V|^2 \cdot d_{\max}} \leq \lambda_2(G). \tag{3.25}$$

In Theorem 62 we used $h_k(G)$ to get an upper bound on $m_k(G)$. The main result of [30] is obtaining bounds on $\rho_k(G)$. Recall that (Definition 36)

$$\rho_k(G) := \min_{\{S_i\}\in\mathcal{S}_k} \max_{i=1,2,\ldots,k} \frac{|\partial S_i|}{|S_i|}. \tag{3.26}$$

Using $\rho_k(G)$ instead of $h_k(G)$ we will prove the following theorem:

**Theorem 65.** *For every finite, connected graph $G$, there is a universal constant $\eta$ such that*

$$m_k(G) \leq \frac{|V|}{4(k-1)} \cdot \left[\eta \cdot k^2 \sqrt{d_{\max}\lambda_k(G)}(|V|-1) - 1\right] \tag{3.27}$$

*for all $k \in \mathbb{N}$ with $2 \le k \le |V|$. It follows that*

$$\begin{aligned} m(G) &\le \eta \cdot \min_{k=2,\dots,N} \frac{|V|}{4(k-1)} \cdot \left[k^2\sqrt{d_{\max}\lambda_k(G)}(|V|-1)-1\right] \\ &\le \eta \cdot \min_{k=2,\dots,N} \frac{|V|}{2} \cdot \left[k\sqrt{d_{\max}\lambda_k(G)}(|V|-1)-1\right], \end{aligned}$$

*where the final inequality uses that $\frac{k}{k-1} \le 2$.*

To prove Theorem 65, we will use the work of Lee, Oveis Gharan, and Trevisan [56]. Some authors define an isoperimetric constant which does not require that $\cup_{i=1}^{k} V_i = V$ in the definition of $h_k(G)$. Instead, they consider the family $\mathcal{S}_k$ of sets $\{S_i\}_{i=1}^{k}$, where $\emptyset \neq S_i \subset V$ for every $i = 1, \dots, k$, and $S_i \cap S_j = \emptyset$ for $i \neq j$.

While it is natural to compare $m_k(G)$ with $h_k(G)$, as in Lemma 63, results of Lee, Oveis Gharan, and Trevisan relate $\tilde{\rho}_k(G)$ to $\tilde{\lambda}_k(G)$ [56, Theorem 1.1]. Similar to how we proved Theorem 39, we can use [56] to get the following:

**Theorem 66.** *For every graph $G$ with $d_{\max}$ equal to the highest degree, for every $k \in \mathbf{N}$, we have that*

$$\rho_k(G) \le O(k^2) \cdot \sqrt{d_{\max}\lambda_k(G)}. \tag{3.28}$$

Since we have an upper bound for $m_k(G)$ in terms of $h_k(G)$, and by Theorem 66 we get an upper bound on $\rho_k(G)$ in terms of $\lambda_k(G)$, we complete our estimate by proving an upper bound on $h_k(G)$ in terms of $\rho_k(G)$.

**Lemma 67.** *For a connected, finite graph $G$ and natural number $2 \le k \le |V|$, there are exactly two possible relationships between $\rho_k(G)$ and $h_k(G)$, which are mutually exclusive. These possible relationships are*

1. $\rho_k(G) = h_k(G)$,

2. $\rho_k(G) < h_k(G) \le \frac{\rho_k(G)(|V|-1)-1}{2}$.

*Proof.* The lower bound $\rho_k(G) \le h_k(G)$ is a direct consequence of the definitions. For the upper bound, let the minimum $\rho_k(G)$ be attained by the disjoint subsets $\{V_i\}_{i=1}^k \in \mathcal{S}_k$ with

$$\frac{|\partial V_1|}{|V_1|} \leq \frac{|\partial V_2|}{|V_2|} \leq \cdots \leq \frac{|\partial V_k|}{|V_k|} = \rho_k(G),$$

and define $W := V \setminus \left[\cup_{i=1}^k V_i\right]$. If there are multiple candidate families $\{V_i\} \in \mathcal{S}_k$ that attain the minimum in $\rho_k(G)$, we choose one that yields a smallest set $W$. Then, if $W = \emptyset$, we can conclude that $\rho_k(G) = h_k(G)$.

Assume that $W \neq \emptyset$. In this case, $\rho_k(G) < h_k(G)$. Since $G$ is connected, there is a $j$ so that $E(W, V_j) \geq 1$. Define $V_j' := V_j \cup W$. Then,

$$\begin{aligned}
|\partial V_j'| &= |\partial W| + |\partial V_j| - 2|E(W, V_j)| \\
&= \sum_{i=1}^k |E(W, V_i)| + |\partial V_j| - 2|E(W, V_j)| \\
&= \sum_{\substack{i=1,\ldots,k \\ i \neq j}} |E(W, V_i)| + |\partial V_j| - |E(W, V_j)| \\
&\leq \sum_{i=1}^k |\partial V_i| - |E(W, V_j)| \\
&\leq \rho_k(G) \sum_{i=1}^k |V_i| - |E(W, V_j)| \\
&= \rho_k(G)(|V| - |W|) - |E(W, V_j)| \\
&\leq \rho_k(G)(|V| - |W|) - 1.
\end{aligned}$$

Thus, since $|W| \geq 1$, we have that

$$\frac{|\partial V_j'|}{|V_j'|} \leq \frac{\rho_k(G)(|V| - |W|) - 1}{|V_j| + |W|} \leq \frac{\rho_k(G)(|V| - 1) - 1}{|V_j| + 1}.$$

The last inequality follows from the fact that the function $f(x) := \frac{\rho_k(G)(|V|-x)-1}{|V_j|+x}$ is decreasing for $x \geq 0$. To see this note that $f$ is decreasing for $x \geq 0$ as long as $\rho_k(G) \geq 1/(|V| + |V_j|)$.

However, since $\rho_k(G)$ is an isoperimetric ratio of subsets of $V$ and $G$ is connected, we know that

$$\rho_k(G) \geq 1/|V| > 1/(|V| + |V_j|).$$

In fact, we have the following stronger inequality, where $r$ is the edge-conectivity of $G$:

$$\rho_k(G) \geq \frac{r}{|V| - k + 1} \geq \frac{1}{|V| - 1}. \tag{3.29}$$

To see this note that $\rho_K(G) = |\partial V_k|/|V_k|$ and $|\partial V_K| \geq r$, while $|V_k| \leq |V| - k + 1$.

Therefore, for every $j$ such that $E(W, V_j) \geq 1$, we have

$$\frac{|\partial V_j'|}{|V_j'|} \leq \frac{\rho_k(G)(|V| - 1) - 1}{|V_j| + 1} \leq \frac{\rho_k(G)(|V| - 1) - 1}{2}.$$

Note that the right hand side of the estimate does not depend on $j$. It follows that

$$\begin{aligned}
h_k(G) &\leq \min_{j: E(V_j, W) \geq 1} \left\{ \max \left\{ \frac{|\partial V_1|}{|V_1|}, \ldots, \frac{|\partial V_j'|}{|V_j'|}, \ldots, \frac{|\partial V_k|}{|V_k|} \right\} \right\} \\
&= \max \left\{ \frac{|\partial V_k|}{|V_k|}, \frac{\rho_k(G)(|V| - 1) - 1}{2} \right\} \\
&= \max \left\{ \rho_k(G), \frac{\rho_k(G)(|V| - 1) - 1}{2} \right\} \\
&= \frac{\rho_k(G)(|V| - 1) - 1}{2},
\end{aligned}$$

where the last equality follows from the fact that $\rho_k(G) < h_k(G)$. □

**Remark 68.** From the proof of Lemma 67 we see that $\rho_k(G) < h_k(G)$ implies that $h_k(G) \leq \frac{\rho_k(G)(|V|-1)-1}{2}$ and therefore also

$$\rho_k(G) > \frac{1}{|V| - 3}.$$

The following example shows that the upper bound in Lemma 67 cannot be improved.

**Example 3.5.1.** Consider the star graph $G_{k+1}$ on $k+1$ vertices, where vertices $v_1, v_2, \ldots, v_k$ have degree one and $v_i \sim w$ for $1 \leq i \leq k$, resulting in vertex $w$ having degree $k$. We claim that $\rho_k(G_{k+1}) = 1$. To see that $\rho_k(G_{k+1}) \geq 1$, let $W_1, W_2, \ldots, W_k$, be any partition of non-empty disjoint vertex subsets of $G_{k+1}$. The fact that $G_{k+1}$ is connected implies that $|\partial W_i| \geq 1$ for every $i = 1, 2, \ldots, k$. Further, since each $W_i$ is non-empty and $G_{k+1}$ has $k+1$ vertices, we must have $|W_i| = 1$ for at least $k-1$ of the indices in $\{1, 2, \ldots, k\}$. It follows that

$$\max_{i=1,2,\ldots,k} \left\{ \frac{|\partial W_i|}{|W_i|} \right\} \geq \max_{i=1,2,\ldots,k} \left\{ \frac{1}{|W_i|} \right\} = 1.$$

Since the partition $\{W_1, \ldots, W_k\}$ was arbitrary, we conclude that $\rho_k(G_{k+1}) \geq 1$. Conversely, if we define $V_i := \{v_i\}$, for $i = 1, \ldots, k$, then $\rho_k(G_{k+1}) \leq \frac{|\partial V_k|}{|V_k|} = 1$. Hence, we have shown that $\rho_k(G_{k+1}) = 1$. Furthermore, if we let $V_1' := V_1 \cup \{w\}$ and $V_j' := \{v_j\}$ for $j = 2, \ldots, k$, then we get an upper bound for $h_k(G_{k+1})$ given by

$$h_k(G_{k+1}) \leq \frac{|\partial V_1'|}{|V_1'|} = \frac{k-1}{2} = \frac{\rho_k(G_{k+1})(|V|-1)-1}{2}.$$

See Figure 3.2 for the example $G_{8+1}$.

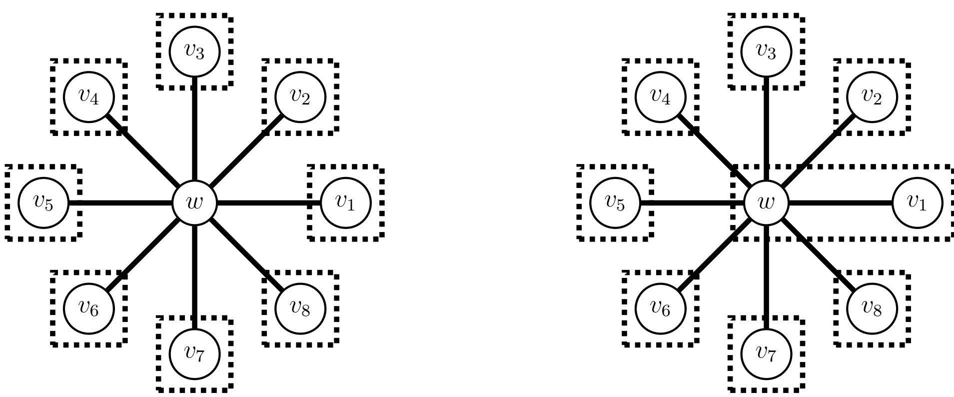



**Figure 3.2**: *Two partitions for the star graph $G_{8+1}$.*

*Proof of Theorem 65.* Let $\eta > 0$ be a universal constant such that

$$\rho_k(G) \leq \eta \cdot k^2 \cdot \sqrt{d_{\max}\lambda_k(G)},$$

which follows from Theorem 66. Therefore, we have that

$$\begin{aligned}
m_k(G) &\leq \frac{|V|}{2(k-1)} \cdot h_k(G), && \text{by Lemma 63,} \\
&\leq \frac{|V|}{4(k-1)} \cdot \left[\rho_k(G)(|V|-1) - 1\right], && \text{by Lemma 67,} \\
&\leq \frac{|V|}{4(k-1)} \cdot \left[\eta \cdot k^2 \sqrt{d_{\max}\lambda_k(G)}(|V|-1) - 1\right], && \text{by Lemma 66.}
\end{aligned}$$

This proves the bounds in Equation 3.27.

The bounds for $m(G)$ follow from taking the minimum of each term of Equation (3.27) over $k = 2, \ldots, N$. This completes the proof of Theorem 65. □

## 3.6 Examples

In the examples below we demonstrate the two inequalities: (3.17)(lower bound on strength in terms of eigenvalues) and (3.25) (lower bound on $\lambda_2$ in terms of strength).

**Example 69** (Complete graph)**.** Consider the complete graph $K_n = (V, E)$ on $n$ nodes, where $n \geq 2$. First, we claim that $K_n$ is homogeneous, i.e., the trivial partition is a minimum feasible partition. Suppose we have a feasible partition $P = \{S_1, ..., S_k\}$ with $S_1 \cup ... \cup S_k = V$ and $|S_1| \geq \ldots \geq |S_k|$. Say $|S_i| = a_i$ for $i = 1, ..., k$. Then we have $a_1 + ... + a_k = n$. Note that if we merge all the nodes in each set $S_i$ of the partition and remove all the multi-edges, we get a complete graph on $k$ nodes. Since $K_n$ has $\frac{n(n-1)}{2}$ edges, we have

$$\begin{aligned}
|E(P)| &= \frac{n(n-1)}{2} - \frac{a_1(a_1-1)}{2} - \ldots - \frac{a_n(a_n-1)}{2} \\
&= \frac{n^2 - n - (a_1^2 + \ldots a_k^2) + (a_1 + \ldots + a_k)}{2} \\
&= \frac{n^2 - n - (a_1^2 + \ldots a_k^2) + n}{2} \\
&= \frac{n^2 - (a_1^2 + \ldots a_k^2)}{2}.
\end{aligned} \tag{3.30}$$

Now, we claim that $a_1^2 + \dots a_k^2$ is maximized when $a_1 = n - (k-1)$ and $a_j = 1$ for $j = 2, \dots k$. In fact, note that given some $a_1, \dots, a_k$ with $a_1 + \dots a_k = n$, for $a_i \geq a_j$, we have

$$\begin{aligned} (a_i+1)^2 + (a_j-1)^2 - a_i^2 - a_j^2 &= a_i^2 + 2a_i + 1 + a_i^2 - 2a_j + 1 - a_i^2 - a_j^2 \\ &= 2(a_i - a_j) + 2 \geq 0. \end{aligned} \tag{3.31}$$

Hence, repeated application of (3.31) shows that $a_1^2 + \dots a_k^2$ is maximized when $a_1 = n-(k-1)$ and $a_j = 1$ for $j = 2, \dots k$. Hence,

$$\begin{aligned} a_1^2 + \dots a_k^2 &\leq (n-(k-1))^2 + 1^2 \dots + 1^2 \\ &= n^2 - 2n(k-1) + (k-1)^2 + k - 1 \\ &= n^2 - 2nk + 2n + k^2 - 2k + 1 + k - 1 \\ &= n^2 - 2nk + 2n + k^2 - k. \end{aligned}$$

From (3.30) the weight of the partition $P$ is $\frac{n^2 - (a_1^2 + \dots a_k^2)}{2(k-1)}$ and from above

$$\begin{aligned} \frac{|E(P)|}{k-1} &\geq \frac{n^2 - (n^2 - 2nk + 2n + k^2 - k)}{2(k-1)} \\ &= \frac{2nk - 2n - k^2 + k}{2(k-1)} \\ &= \frac{2n(k-1) - k(k-1)}{2(k-1)} \\ &= \frac{2n-k}{2} \\ &\geq \frac{2n-n}{2} \\ &= \frac{n}{2}. \end{aligned}$$

But the weight of the trivial partition is

$$\frac{n(n-1)}{2(n-1)} = \frac{n}{2}.$$

Thus, $K_n$ is homogeneous, i.e., $m(K_n) = \frac{n}{2}$.

We know that for $K_n$ the Laplacian eigenvalues are 0 with multiplicity 1 and $n$ with multiplicity $n-1$. Hence, the lower bound from (3.17) in this case is as follows:

$$\begin{aligned}\min_{k=2,\ldots,n} \frac{\lambda_k(G)}{4(k-1)} + \frac{r}{2} &= \frac{n}{4(n-1)} + \frac{n-1}{2} \\ &= \frac{n + 2(n^2 - 2n + 1)}{4(n-1)} \\ &= \frac{2n^2 - n + 1}{4(n-1)}.\end{aligned}$$

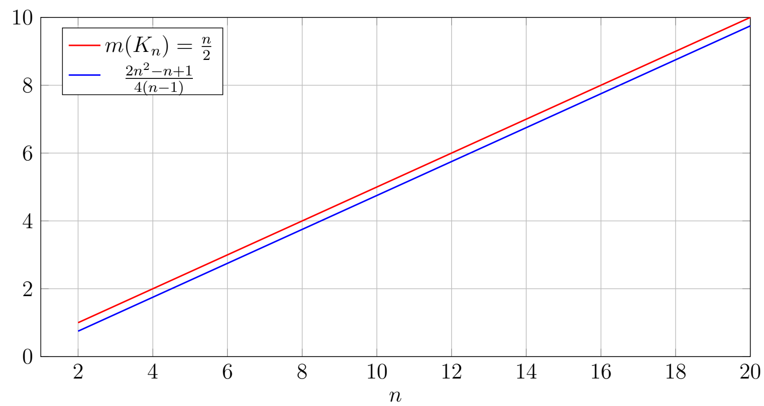



**Figure 3.3**: *Lower bound on $m(K_n)$.*

Ignoring the constant $\eta$, the lower bound on $\lambda_2$ from (3.25) in the case of $K_n$ is the following:

$$\frac{m(G)^2}{|V|^2 \cdot d_{\max}} = \frac{n^2/4}{n^2 \cdot (n-1)} = \frac{1}{4(n-1)}.$$

The comparison of this lower bound and $\lambda_2$ is illustrated in Figure 3.4.

**Example 70** (Path Graph)**.** Consider the path graph $P_n = (V, E)$ on $n$ nodes, where $n \geq 2$. Note that removing any $k$ edges from $P_n$ for $k \leq n$ results in $k+1$ components. Hence $m(P_n) = \frac{k}{k+1-1} = 1$.

Now we know that for $P_n$ the Laplacian eigenvalues are $2 - 2\cos(\frac{(k-1)\pi}{n})$ [44]. Also since

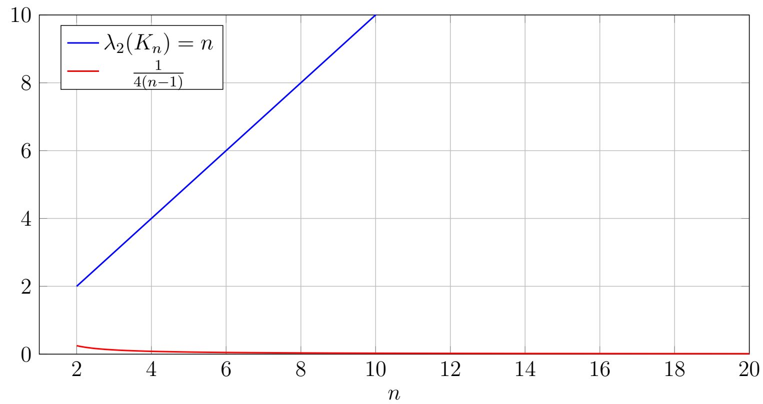



**Figure 3.4**: *Lower bound on $\lambda_2(K_n)$.*

$P_n$ is 1 edge-connected, the lower bound from (3.17) in this case is as follows:

$$\min_{k=2,\dots,n} \frac{2-2\cos(\frac{(k-1)\pi}{n})}{4(k-1)} + \frac{1}{2}.$$

It is clear that the minimum is achieved when $k = n$, and hence,

$$\min_{k=2,\dots,n} \frac{2-2\cos(\frac{(k-1)\pi}{n})}{4(k-1)} + \frac{1}{2} = \frac{2-2\cos(\frac{(n-1)\pi}{n})}{4(n-1)} + \frac{1}{2}.$$

Ignoring the constant $\eta$, the lower bound on $\lambda_2$ from (3.25) in the case of $P_n$ is the following:

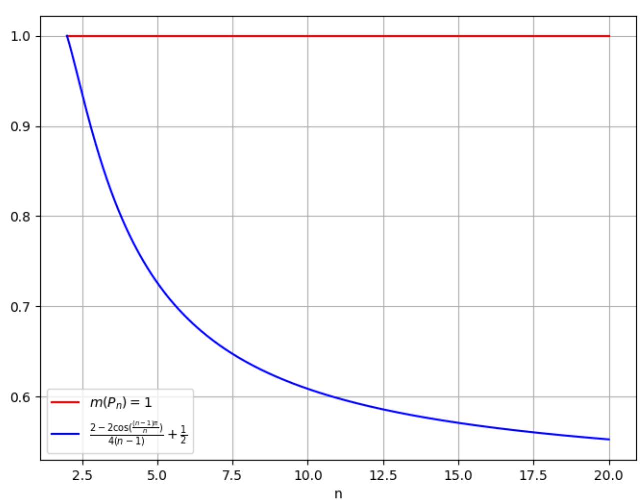



**Figure 3.5**: *Lower bound on $m(P_n)$.*

$$\frac{m(G)^2}{|V|^2 \cdot d_{\max}} = \frac{1}{n^2 \cdot 2} = \frac{1}{2n^2}.$$

The comparison of this lower bound and $\lambda_2$ is illustrated in Figure 3.6.

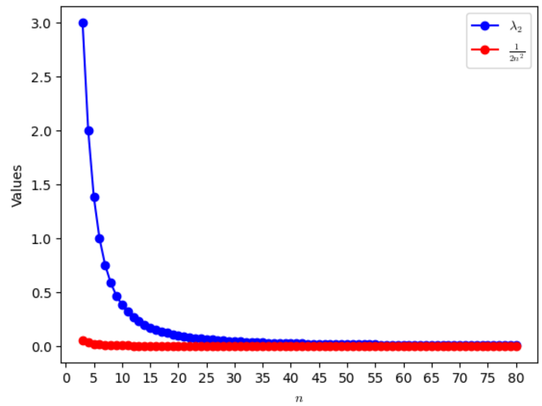



**Figure 3.6**: *Lower bound on $\lambda_2(P_n)$.*

**Example 71** (Star Graph)**.** Consider the star graph $S_n = (V, E)$ on $n$ nodes, where $n \geq 3$. Note that removing any $k$ edges from $S_n$ for $k \leq n$ results in $k+1$ components. Hence the strength $m(S_n) = \frac{k}{k+1-1} = 1$.

Now we know that for $S_n$ the Laplacian eigenvalues are 0 with multiplicty 1, 1 with multiplicity $n-2$ and $n$ with multiplicty 1 [44]. Also since $P_n$ is 1 edge-connected, the lower bound from (3.17) in this case is as follows:

$$\begin{aligned}\min_{k=2,\ldots,n} \frac{\lambda_k}{4(k-1)} + \frac{1}{2} &= \frac{1}{4(n-1-1)} + \frac{1}{2} \\ &= \frac{1}{4(n-2)} + \frac{1}{2}.\end{aligned}$$

Ignoring the constant $\eta$, the lower bound on $\lambda_2$ from (3.25) in the case of $P_n$ is the following:

$$\frac{m(G)^2}{|V|^2 \cdot d_{\max}} = \frac{1}{n^2 \cdot (n-1)} = \frac{1}{n^2(n-1)}.$$

The comparison of this lower bound and $\lambda_2$ is illustrated in Figure 3.8.

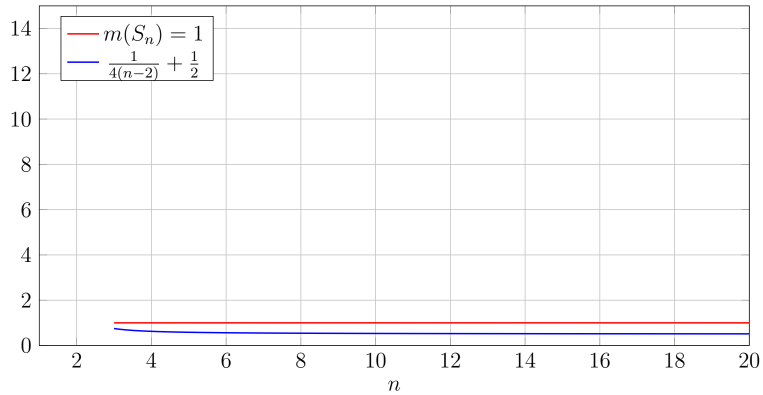



**Figure 3.7**: *Lower bound on $m(S_n)$.*

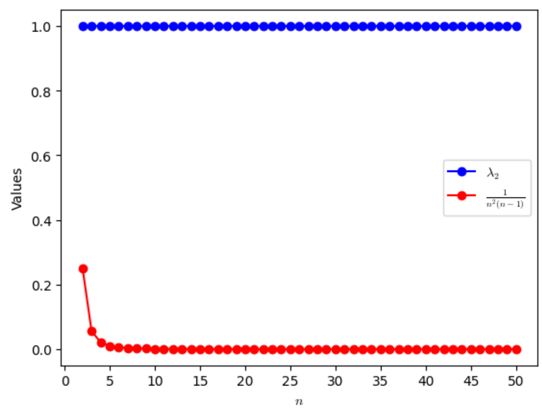



**Figure 3.8**: *Lower bound on $\lambda_2(S_n)$.*

**Example 72** (Cycle Graph)**.** Consider the cycle graph $C_n = (V, E)$ on $n$ nodes, where $n \geq 3$. Note that to disconnect $C_n$ we have to remove at least 2 edges. Once we remove 2 edges, we get a disjoint union of path graphs. Once we remove 2 edges, removing every $k$ edges results in $k$ new components. Hence, $m(C_n) = \frac{2+k}{1+k} = 1 + \frac{1}{1+k}$. Since $k$ can be at most $n-2$,

$$\begin{aligned} m(C_n) &= 1 + \frac{1}{1+n-2} \\ &= 1 + \frac{1}{n-1}. \end{aligned}$$

Now we know that for $C_n$ the Laplacian eigenvalues are $2 - 2\cos(\frac{2(k-1)\pi}{n})$ [44]. Also since

$C_n$ is 2 edge-connected, the lower bound from (3.17) in this case is as follows:

$$\min_{k=2,\ldots,n} \frac{2 - 2\cos(\frac{2(k-1)\pi}{n})}{4(k-1)} + \frac{2}{2}.$$

It is clear that the minimum is achieved when $k = n$, and hence,

$$\min_{k=2,\ldots,n} \frac{2 - 2\cos(\frac{2(k-1)\pi}{n})}{4(k-1)} + \frac{2}{2} = \frac{2 - 2\cos(\frac{2(n-1)\pi}{n})}{4(n-1)} + 1.$$

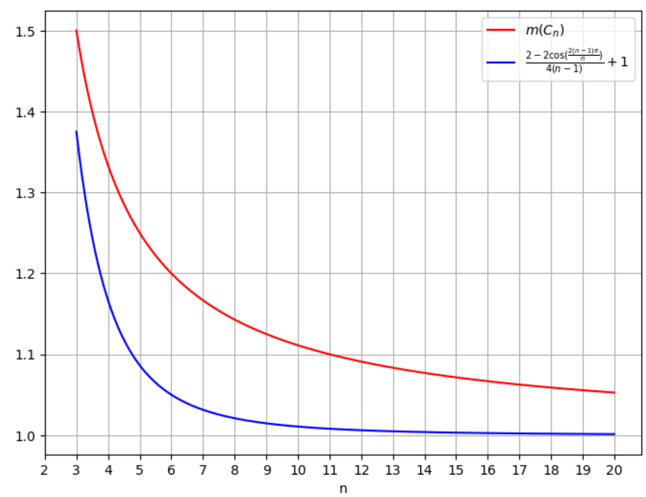



**Figure 3.9**: *Lower bound on $m(C_n)$.*

Ignoring the constant $\eta$, the lower bound on $\lambda_2$ from (3.25) in the case of $C_n$ is the following:

$$\frac{m(G)^2}{|V|^2 \cdot d_{\max}} = \frac{(1 + \frac{1}{n-1})^2}{n^2 \cdot 2}.$$

The comparison of this lower bound and $\lambda_2$ is illustrated in Figure 3.10.

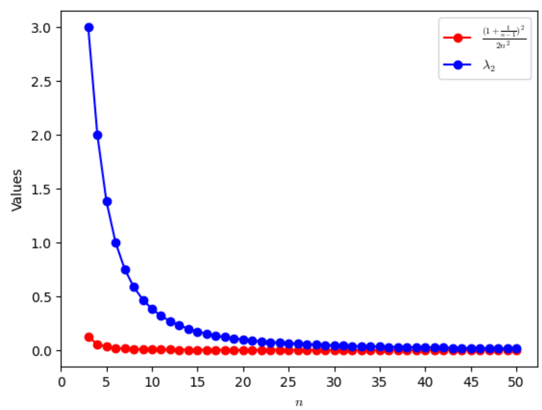



**Figure 3.10**: *Lower bound on* $\lambda_2(C_n)$.

# Chapter 4

# Applications of neural networks to some problems

## 4.1 Overview and Motivation

Neural networks are machine learning models that consist of artificial neurons which are functions—usually non-linear—of one-dimensional inputs. A neural network stacks these neurons to form layers of neurons, and when there are many layers of these neurons, it is called a multilayer perceptron [57]. It is known that neural networks are universal approximators [58, 59, 60, 61]. In other words, any continuous function on compact domains of Euclidean spaces can be approximated to arbitrary precision under specific conditions on the hyperparameters: depth, width, activation function, etc.

In this chapter, we focus on two problems in which we apply graph neural networks in the supervised learning framework to learn these problems and find intricate patterns and connections in the mathematics behind these problems. Apart from this goal, we also have the inherent benefit of using these models to make predictions on unseen data without having to go through the computational route of producing an output specific to a problem, which, sometimes, can be computationally hard (NP-hard problems).

## 4.2 Message Passing Neural Networks (Graph Neural Networks)

In this section, we formally define message passing neural networks which will be our main tool for learning to solve the relevant problems in this chapter.

**Definition 73** (Graph)**.** A *graph* $G = (V, E)$ consists of a set of nodes $V$ and a set of edges $E$. For each node $v \in V$ we denote the neighbors of $v$ by $\mathcal{N}(v)$.

**Definition 74** (Attributed graph)**.** An *attributed graph* $G = (V, E, X, Y)$ is a graph $G$, where the set of vertices is denoted by $V$, the set of edges is denoted by $E$, $X \in \mathbb{R}^{|V| \times d_X}$ contains the node attributes for each node, and $Y \in \mathbb{R}^{|E| \times d_Y}$ contains the edge attributes for each edge. Here $d_X$ and $d_Y$ are the embedding dimensions of the vertices and edges, respectively. We let $x_v \in \mathbb{R}^d$ denote the row of $X$ corresponding to node $v$. ($d$ is called the embedding dimension)

**Example 75.** A semi-supervised graph learning problem consists of an attributed graph where the attributes on the nodes are real numbers (also called signals) and the attributes on the edges are also real numbers, which are called the weights of the graph [62].

**Definition 76** (MPNN)**.** A *message passing neural network (MPNN)* (see Figure 4.1) is a neural network that acts on the node attributes $X$ of an attributed graph $G$ by maintaining a node representation $h_v^{(t)} \in \mathbb{R}^{d_t}$ ($d_t$ is embedding dimension for $t^{\text{th}}$ layer)for each layer and applying layerwise updates according to the following iterative scheme:

1. $h_v^{(0)} = x_v$ for all $v \in V$ (Initialization),
2. $m_v^{(t)} = \psi^{(t)}(h_v^{(t)}, \bigoplus_{u \in \mathcal{N}(v)} h_u^{(t)})$ (Message Aggregation),
3. $h_v^{(t+1)} = \phi^{(t)}(h_v^{(t)}, m_v^{(t)})$ (Update).

We explain the notations below:

- $\psi^{(t)}$ is a learnable function of the concatenation of the two vectors: $(h_v^{(t)}, \bigoplus_{u \in \mathcal{N}(v)} h_u^{(t)})$. It is usually taken to be a composition of an activation function and an affine transformation (also called a linear layer). Note that the weights of $\psi^{(t)}$ are shared among all nodes and so $\psi^{(t)} : \mathbb{R}^{|V| \times d_t} \to \mathbb{R}^{|V| \times d_t}$.

- $\bigoplus$ is any permutation invariant function, and it is commonly taken to be sum, max, or mean.

- For each node $v$, $m_v^{(t)}$ is called the message aggregated from its neighbors.

- $\phi^{(t)}$ is a learnable function of the concatenation of the two vectors: $(h_v^{(t)}, m_v^{(t)})$. It is usually a linear layer, and so $\phi^{(t)} : \mathbb{R}^{|V| \times d_t} \times \mathbb{R}^{|V| \times d_t} \to \mathbb{R}^{|V| \times d_{t+1}}$, where $d_t$ and $d_{t+1}$ are the embedding dimensions for the $t^{th}$ and $t + 1^{st}$ layers respectively. Similar to $\psi^{(t)}$, the weights are shared among the nodes. Observe that the embedding dimensions for different layers can be different, but the input layer and output layer have fixed dimensions depending on the learning task.

- Intuitively, for each node $v$ the $t^{th}$ MPNN layer first applies a linear layer composed with activation using $\psi$ to all the node features to get their "outputs", aggregates the messages from the embeddings of the neighbors of $v$ using $\oplus$, and finally mixes that messages with the output of $v$ using $\phi$ to get the updated feature vector $h_v^{(t+1)}$.

## 4.3 Saliency analysis for minimum feasible partition

### 4.3.1 Saliency analysis of neural networks

While traditional statistical models such as linear regression and logistic regression are interpretable, that is, these models give a clear insight into the relationship between the input features and the target variable, modern deep neural networks are still considered black box systems owing to their complex architecture and training process, and the inherent non-linearity present in their computational modules. Explainable AI (XAI) is a field that aims

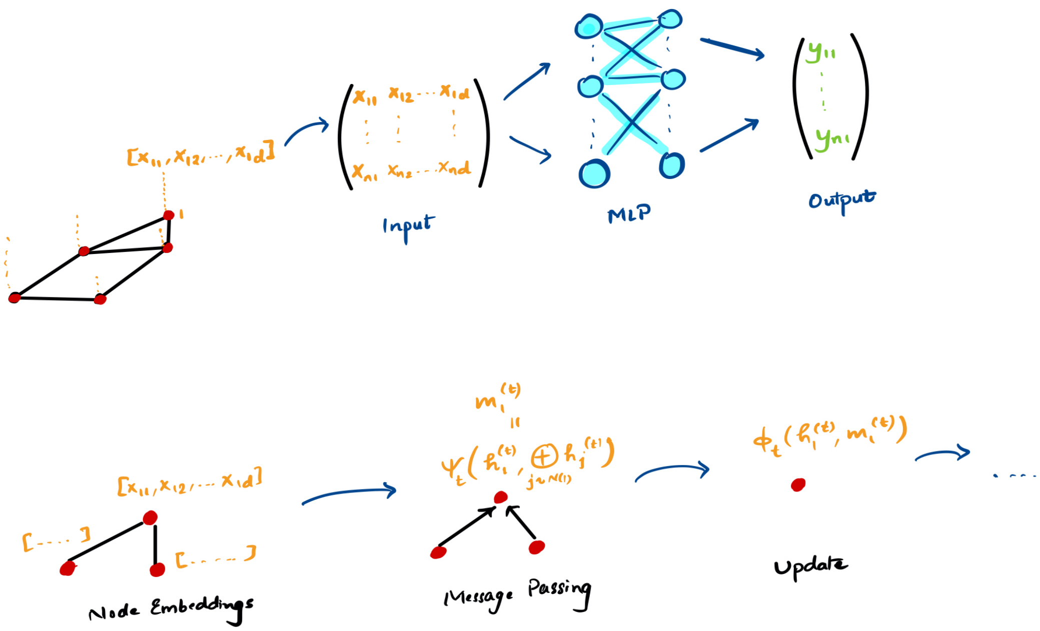



**Figure 4.1**: *MPNN illustration.*

to provide a coherent explanation of the computational and decision-making process of a machine learning model [63].

Analyzing the gradients of target variables with respect to input features is a method for measuring and ranking feature importance [64, 65]. For the following problem of predicting the edges between the minimum feasible partition of a graph, we used the mean of the absolute value of the gradients with respect to each input feature in the test dataset.

### 4.3.2 Problem

Let $G = (V, E)$ be a finite graph. The problem we want the neural network model to solve is the problem of predicting the edges between the minimum feasible partition solution (see Section (3.2)):

$$\begin{aligned} \text{minimize} \quad & \frac{|E(P)|}{k_P - 1} \\ \text{subject to} \quad & P \text{ is a feasible partition.} \end{aligned} \tag{4.1}$$

As mentioned in Section 3.1, finding the minimum feasible partition has a polynomial time algorithm [51, 52]. The motivation behind training this machine learning model is to find, using machine learning, the important topological features of the graph that are related to the minimum feasible partition problem, and hence, affect the strength of graphs.

### 4.3.3 Synthetic Dataset

We generated 2755 random graphs using the following models:

1. **Random Geometric Graphs** [66]: The random graph for this model was generated with parameters $\min(\text{multiplier} \cdot p, 0.99)$, where multiplier was chosen uniformly randomly from $[0.8, 1, 1.2, 1.5, 2, 2.5, 3]$.

2. **Erdos-Renyi Graphs** [67]: The random graph for this model was generated with the parameter:
   With threshold $= \frac{\ln(n)}{n}$ ($n$ is the size of the graph)

$$p = \text{Uniform}(\text{threshold}, \min(3 \cdot \text{threshold}, 1))$$

In each iteration of the random graphs generation process, we select uniformly at random from one of the above models and create a random graph from the corresponding model. We select the order of the graph from $[a, b]$ in steps of size $s$ and generate an equal number of graphs for each order. For this dataset we selected the interval $[a, b] = \{[10, 100], [11, 121]\}$ and $s = 10$.

Once we have the list of random graphs, we iterate over this list, and for each graph we calculate the node features: degree, betweeness, closeness, clustering coefficient, node resistance curvature, and 10 smallest non-zero Laplacian eigenvalues. See Appendix A for the definitions of these node features.

In order to get the required edges for each graph, we use the theory of spanning tree modulus [31]. In particular, we use the following theorem:

**Theorem 77.** *[68] Let $\mu^*$ be optimal for the MEO problem and let $\eta^* = \mathbf{N}^T\mu^*$. Define*

$$E^* = \{e \in E : \eta^*(e) = K\}, \qquad \textit{where} \quad K := \max_{e\in E} \eta^*(e).$$

*Then there exists a feasible partition $P^*$ such that*

1. $E^* = E_{P^*}$,
2. $|\gamma^* \cap E_{P^*}| = k_{P^*} - 1 \qquad \forall \gamma^* \in supp\mu^*$,
3. $\eta^*(e) = K = w(P^*)^{-1} \qquad \forall e \in E_{P^*}$,
4. $w(P^*) = \min\{w(P) : P$ *is a feasible partition of* $G\}$.

To calculate $\eta^*$ for the spanning tree modulus we used the basic algorithm explained in "The Modulus Book" [69] and the code therein. Note that this algorithm will only calculate $\eta^*$ up to a specified threshold, which in our case was chosen to be $10^{-3}$. Finally, we rounded the $\eta^*$ values to 3 decimal places and found the edges that have the maximum $\eta^*$ value to get the required edges in the minimum feasible partition.

### 4.3.4 Methodology

The model (Figure 4.2) consists of three hidden layers with dimensions equal to 64. The input features are of dimensions equal to 15 as there are 5 topological node features and 10 Laplacian eigenvalues for each node. There is also a dropout of 0.5 at the end of each GCN layer. The final edge prediction is evaluated by applying a sigmoid to the concatenation of the final hidden layer output of the two nodes.

### 4.3.5 Training

We use the binary cross-entropy loss for training this model. For the training loop, the training data is loaded in batches of size 16. We use the Adam optimizer with an initial learning rate of 0.001 and weight decay of $10^{-5}$. We use 151 epochs in total to train the model.

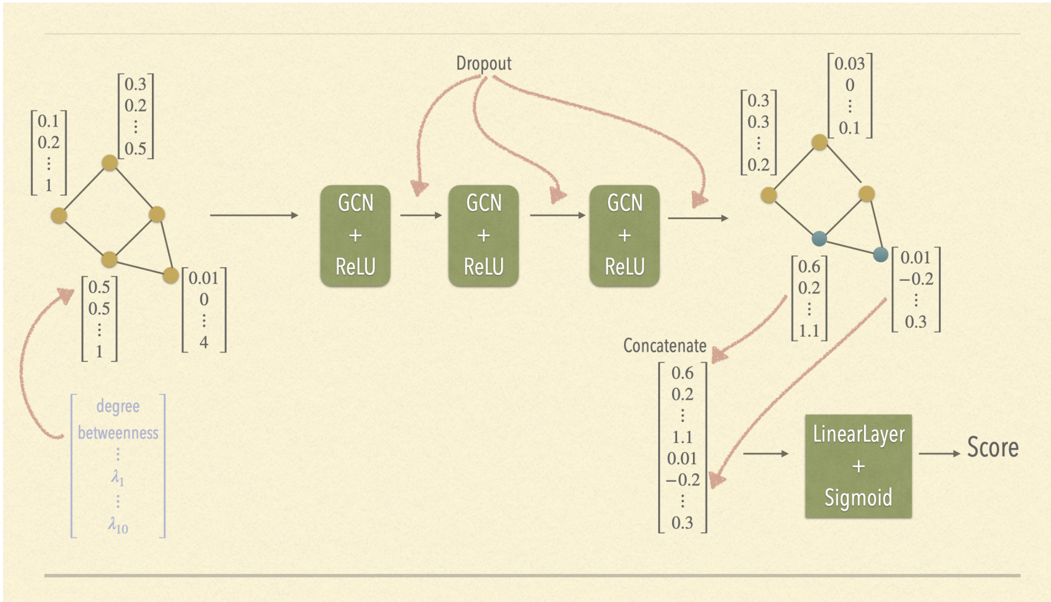



**Figure 4.2**: *Model architecture for minimum feasible partition GNN model.*

### 4.3.6 Testing

To test the model, we use the threshold of 0.5 to make the binary prediction. We use accuracy, precision, recall, and F1 score to test the model.

Accuracy measures the overall correctness of the model:

$$\text{Accuracy} = \frac{\text{True positives} + \text{True negatives}}{\text{Total predictions}}$$

Precision measures the accuracy of the positive class predictions:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

In other words, precision measures how often the model is correct, when the prediction is positive.

Recall measures the ability of the positive class predictions:

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

In other words, recall measures how often the model correctly identifies the positive instances.

F1 score is the harmonic mean of precision and recall, and it gives a measure of the balance between precision and recall:

$$\text{F1 score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

### 4.3.7 Results and Analysis

The results are as follows: The confusion matrix is shown in Figure 4.3. Aggregating the

**Table 4.1**: *Metric Results.*

| **Metric** | **Value** |
|---|---|
| Accuracy | 0.8756 |
| Precision | 0.8380 |
| Recall | 0.9369 |
| F1 | 0.8847 |

mean of the absolute value of the gradients with respect to each input feature in the test dataset and normalizing we get the result shown in Figure 4.4. In this figure, $\text{Spectral}_i$ denotes the $i^{\text{th}}$ smallest combinatorial Laplacian eigenvalue.

**Model with only curvature as the node feature:** We trained a graph neural network using the same hyper-parameters as above, but only node resistance curvature as the input feature. The metric results are in Table 4.2 and the confusion matrix is in Figure 4.5.

**Table 4.2**: *Metric results for model with only curvature as the node feature.*

| **Metric** | **Value** |
|---|---|
| Accuracy | 0.8869 |
| Precision | 0.9479 |
| Recall | 0.8232 |
| F1 | 0.8812 |

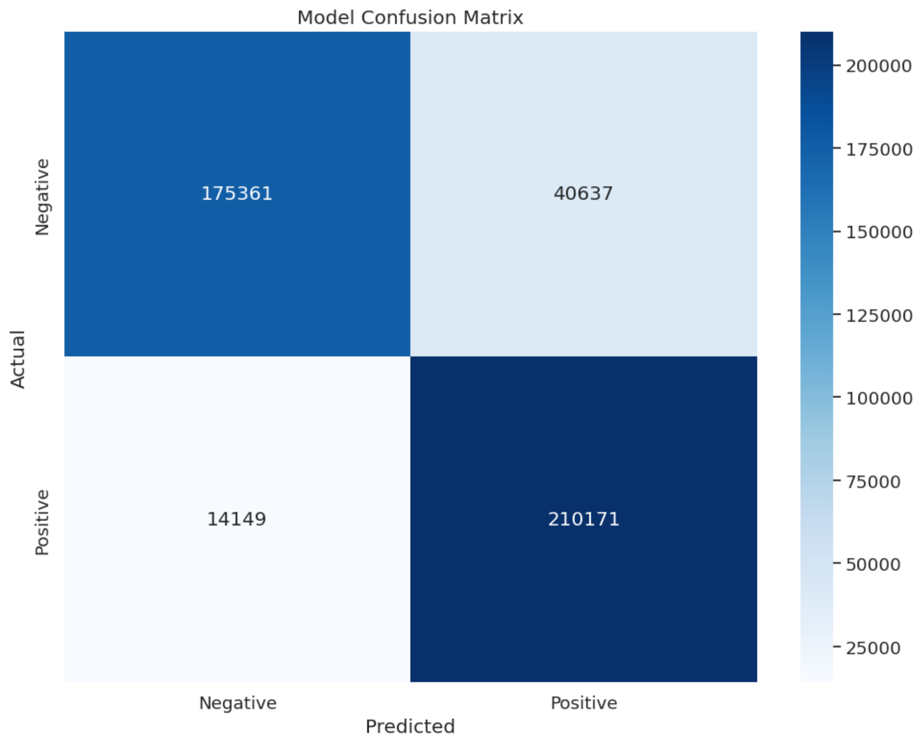



**Figure 4.3**: *Confusion matrix.*

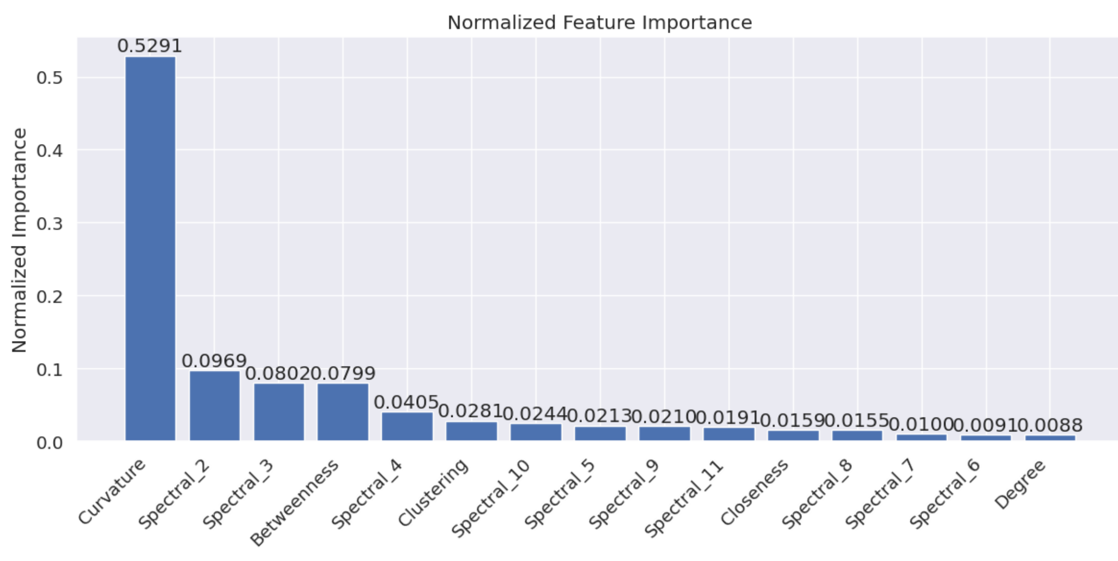



**Figure 4.4**: *Saliency Analysis.*

Note that resistance curvature plays a significant role in the predictions of the above graph neural network model. This suggests that the minimum feasible partition problem and the strength of graphs in particular should have a relationship to the curvature of graphs.

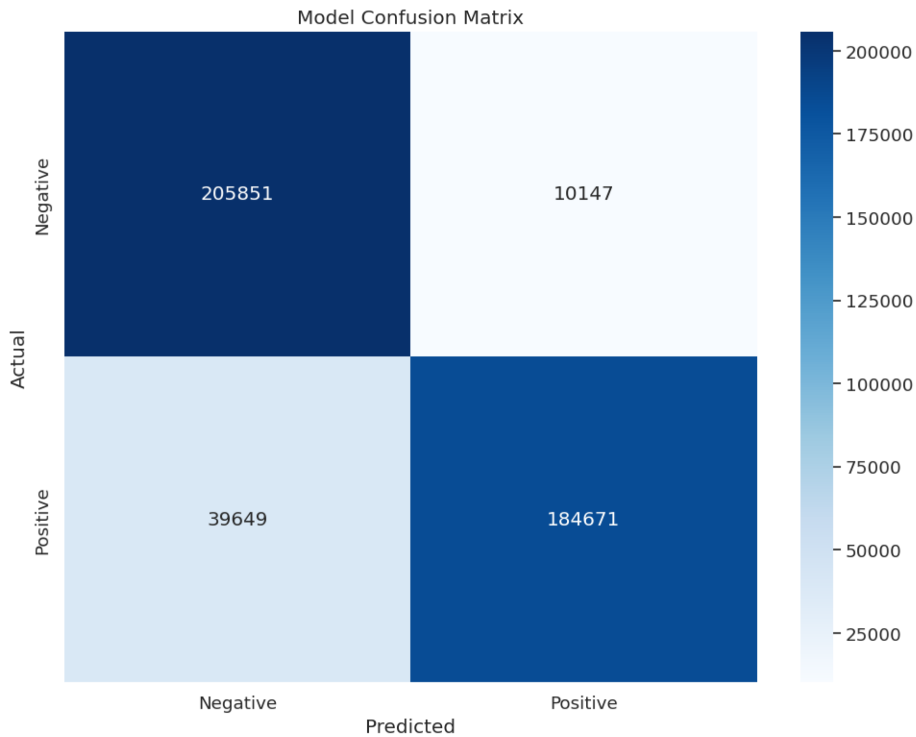



**Figure 4.5**: *Confusion matrix for model with only curvature as the node feature.*

Intuitively node resistance curvature $p_i$ and other notions of discrete curvature capture the connectedness of the neighbors of a node. We quote Devriendt and Lambiotte [70]: "If the neighbourhood of a node is 'tree-like' with few short cycles, then the local relative resistances will be large and thus $p_i$ will be small, as expected for a notion of curvature [71, Thm. 1]. If the neighbourhood is 'clique-like' with many short-cycles, then the local relative resistances will be small with a large $p_i$ as a result."

We can intuitively see why discrete curvature seems to be an important feature in the solution of the minimum feasible partition problem. Consider the barbell graph in Figure 4.6. This graph has a clear community structure, and the minimum feasible partition is the partition into two communities of cliques. The endpoints of the edges which connects the two cliques has negative curvature, in fact, they are equal, while the other nodes have positive curvature.

Although the minimum feasible partition problem is essentially different from community detection, in graphs with strong community structures such as barbell graphs, the two

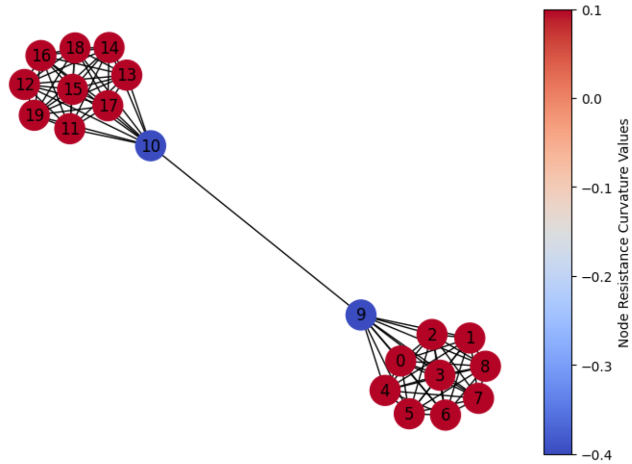



**Figure 4.6**: *Barbell graph node resistance curvature visualization.*

problems coincide. Ollivier Ricci curvature [72] has been used extensively for community detection [73, 74, 75]. While resistance curvature has not been applied as extensively as Ollivier-Ricci curvature, our graph neural network model shows that resistance curvature has a relation to the minimum feasible partition problem and strengths of graphs.

## 4.4 Graph neural networks applied to a percolation problem

### 4.4.1 Background

Brooks and Porter [76] introduced a content-spreading model combining bounded confidence opinion models and percolation models on networks. Unlike other opinion dynamics model, this model relies on the state of the content and updates each node's opinion state based on its distance to the content state. The site percolation model introduced below is a special case of the content spreading model introduced by Brooks and Porter [76].

The concept of influence maximization was first formally introduced by Domingos and Richardson in 2001 [77]. A notable work on this problem is Kempe et al. [78] which formulated influence maximization as a discrete optimization problem and proposed the two fundamental diffusion models: the Independent Cascade (IC) model and the Linear Threshold (LT) model.

### 4.4.2 Site Percolation as an influence problem

For this section, we consider the special case of site percolation. Let $G = (V, E)$ be a finite graph. The percolation starts at a given seed node $v$ which is "open". Each of the other nodes is either open or closed according to a Bernoulli parameter $p$ called the occupation probability. A node $v \in V$ is said to be influenced by $a$ when $v$ is connected to $a$ via a path of open nodes. The probability that a node $v$ is influenced by $a$ is denoted as $\mathbb{P}_a(v)$. When the context is clear, we also denote this probability by simply $\mathbb{P}(v)$.

The influence of $a$ is the expectation of the number of nodes that are influenced by $a$. We have

$$\sigma(a) := \sum_{v \in V} \mathbb{P}_a(v). \tag{4.2}$$

Let $\Gamma$ denote the chordless paths in $G$ and $\Gamma_{a,v}$ the chordless paths joining $a$ and $v$. Recall that a path $\gamma \in G$ is chordless if $\gamma$ is the induced subgraph on its vertices. Note that

$$\mathbb{P}_a(v) = \mathbb{P}(\cup_{\gamma \in \Gamma_{a,v}} \gamma \text{ is open}).$$

By the inclusion-exclusion principle, this is a polynomial expression in $p$. Hence the expected influence (4.2) is also a polynomial in $p$.

We consider the following related problems:

1. **Influence computation:** For a seed $S \subset V$, find the probability that a node $u \in V$ is influenced, denoted as $\mathbb{P}_S(u)$.

2. **Influence maximization:** Find $k$ nodes to form the seed set so as to maximize the influence $\sigma(S)$.

The degree of a node is a good measure of influential seeds for the influence maximization problem, especially, when the occupation probability $p$ is small. In fact, we have:

$$\left.\frac{d\sigma(a)}{dp}\right|_{p=0} = \deg(a). \tag{4.3}$$

Since $\sigma(a)|_{p=0} = 1$ for all $a \in V$, this shows that for a graph $G$, when $p$ is close to zero, the highest degree vertices maximize the influence.

Now we consider the other edge case, i.e., when $p$ is close to 1 for a tree. We have the following:

**Theorem 78.** *Let $G = (V, E)$ be a tree graph. When $p$ is close to 1, the seed node/nodes that maximizes the influence is the centroid set of $G$.*

*Proof.* For $a \in V$ we have $\mathbb{E}[a]|_{p=1} = n$ where $n$ is the number of vertices of $G$. Hence the node(s) that maximize the influence for $p$ close to 1 is one for which the derivative of the expected influence is minimum. Let $d$ denote the shortest path metric on $G$. Note that for the $a \in V$,

$$\begin{aligned} \mathbb{E}[a] &= \sum_{v \in V} \mathbb{P}_a(v) \\ &= 1 + \sum_{v \neq a} p^{d(a,v)}. \end{aligned} \tag{4.4}$$

Hence,

$$\begin{aligned} \left.\frac{d\mathbb{E}[a]}{dp}\right|_{p=1} &= \left.\sum_{v \neq a} d(a,v) p^{d(a,v)-1}\right|_{p=1} \\ &= \sum_{v \neq a} d(a,v). \end{aligned} \tag{4.5}$$

Hence, for $p$ close to 1, the influence is maximized for node/nodes that minimize the distance to all other nodes, i.e., the centroid of the tree. □

The following theorem was proved by C. Jordan. Also see Santhakumaran [79].

**Theorem 79.** *[80] Every tree has a centroid consisting of either one vertex or two adjacent vertices.*

**Example 80.** Consider two star graphs of degree $d \geq 3$ connected by a path graph of order $m$ (See Figure 4.7). The centroid is in the middle of the path, while the highest degree vertices are the centers of the two star graphs.

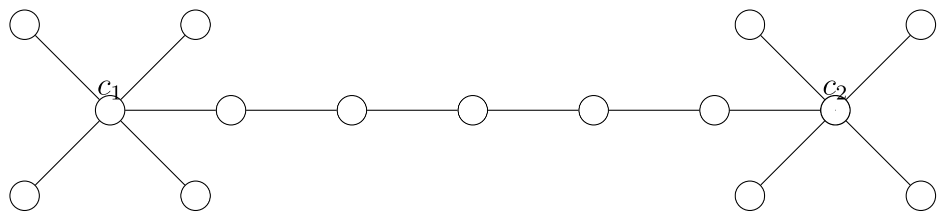



**Figure 4.7**: *Two star graphs $S_d$ connected by a path of length $m$.*

**Example 81.** In Figure 4.8 we show a tree where there exist nodes other than the centroids and the highest degree that attain the maximum influence for some values of $p$ between 0 and 1. For the tree shown in the figure, the centroid nodes are 0 and 2, and the highest degree node is 9.

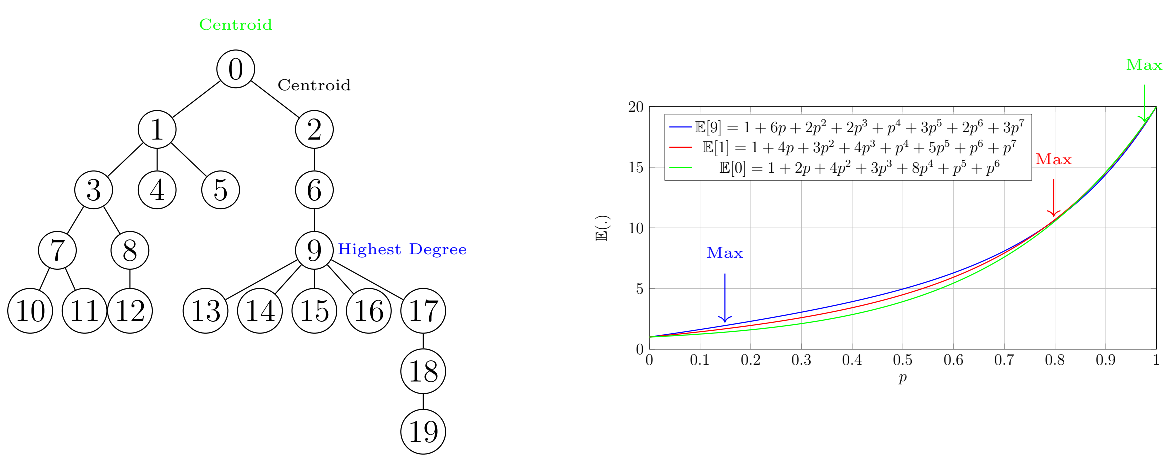



**Figure 4.8**: *Tree graph and influence of the upper-envelope nodes.*

Both influence computation and influence maximization can be solved in polynomial times for trees by first computing the distance matrix from which the percolation matrix (Definition 84) can be derived. Below we will highlight a different way of solving the influence maximization problem without reference to the influence computation problem. For this, we need the following result:

**Theorem 82.** *[81] Let $T$ be a tree on $n$ vertices and $P$ be the corresponding exponential distance matrix. If $q \neq \pm 1$ then*

$$P^{-1} = I + \frac{q^2}{1-q^2}D - \frac{q}{1-q^2}A,$$

*where $D$ is the degree matrix and $A$ is the adjacency matrix of $T$.*

**Remark 83.** [81] consider the $q$ distance matrix and the exponential distance matrix of a tree. The exponential distance matrix is equal to the percolation matrix for trees.

Since $P$ is a diagonally dominant symmetric matrix, we can use the results of [82] to calculate the row sums of $P$ approximately and hence find the node with the maximum influence.

### 4.4.3 Machine Learning model

### 4.4.4 Task

Firstly, we define the percolation matrix of a graph, which we set out to predict.

**Definition 84** (Percolation matrix)**.** Let $G = (V, E)$ be a connected graph. The percolation matrix $P$ of $G$ is defined as

$$P^p_{ij} = \mathbb{P}_i(j; p).$$

In other words, the $ij$-th entry of $P$ is the probability that the node $j$ is influenced when the seed node is $i$ and the occupation probability is $p$.

### 4.4.5 Dataset

We generated 2000 random graphs of sizes 8 to 50 and performed Monte Carlo simulations for percolation on these graphs. For each random graph in the dataset, 9 occupations probabilities were chosen in the intervals $[0.1 + k(\frac{8}{10}), 0.1 + (k+1)\frac{8}{10}]$ where $k \in \{0, 1, ..., 9\}$. Then for each of the random occupation probabilities and each node of a graph, 10,000 Monte Carlo simulations were performed for the corresponding probabilities. Then using this tally a close approximation to the percolation matrix is calculated. The percolation matrices corresponding to occupation probabilities of 0 and 1, which are the identity matrix and the matrix with all ones, are also appended in the dataset. The random graph models we used for these purposes are the following:

1. **Random Geometric Graphs**: The random graph for this model was generated with parameters $\min(\text{multiplier} \cdot p, 0.99)$, where multiplier was chosen uniformly randomly from $[0.8, 1, 1.2, 1.5, 2, 2.5, 3]$.

2. **Erdos-Renyi Graphs**: The random graph for this model was generated with the parameters:

   (a) threshold $= \frac{\ln(n)}{n}$ ($n$ is the size of the graph).

   (b) $p = \text{Uniform}(\text{threshold}, \min(3 \cdot \text{threshold}, 1))$.

3. **Barabasi-Albert Graphs**: The random graph for this model was generated with the parameter $m$ which was a random integer in $[1, \min\{5, \text{size} - 1\})$.

4. **Watts Strogatz Graphs**: The random graph for this model was generated with the following parameters:

   (a) $k = 2 \cdot \text{UniformInt}(2, \min(5, \lfloor \frac{n}{2} \rfloor))$, where UniformInt generates a random integer uniformly in the given interval.

   (b) $p = \text{Uniform}[0.01, 0.3)$.

For each iteration of random graph generation, we uniformly select one of the above random graph models to generate a random graph. We ensure that our graphs are connected by adding edges randomly to connect the different components. Finally, we do an 80-10-10 train-validation-test split.

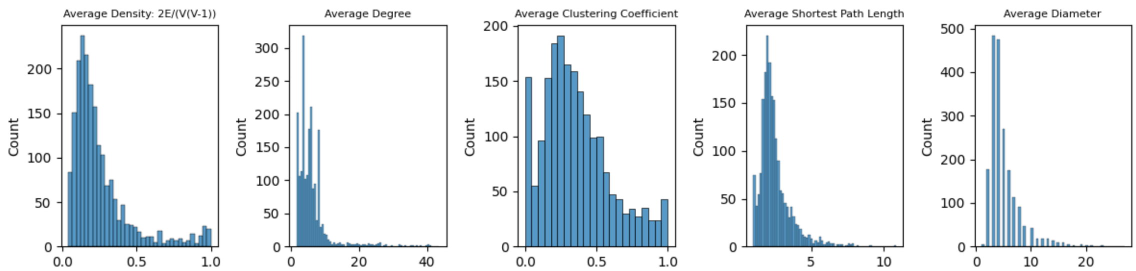



**Figure 4.9**: *Some statistics for the graphs in the dataset.*

### 4.4.6 Model Architecture

Since this task is inherently long-range, we opted to use the Node2Vec embeddings as inputs for the nodes of a graph. The Node2Vec algorithm performs a specified number of random walks on a graph and then uses the Word2Vec algorithm [83] to generate node embeddings [84, 85]. We used the parameters $p = 1, q = 0.25$ for the biased random walk parameters in Node2Vec.

**Table 4.3**: *Hyperparameter Values.*

| Hyperparameter | Value |
|---|---|
| Learning Rate | 0.001 |
| Weight Decay | 0.0001 |
| Batch Size | 32 |
| Epochs | 100 |
| Patience | 10 |

Once we have the node embeddings, we append the corresponding occupation probability $p$ for each data point. Then we use a three-layer message passing neural network with batch normalization and dropout to process the node inputs. Then another three-layer multilayer perceptron with RELU activation is used to further process the embeddings. Finally, for each pairwise node prediction, the processed embeddings are concatenated and a three-layer multilayer perceptron with a sigmoid output layer is used to make the pairwise predictions.

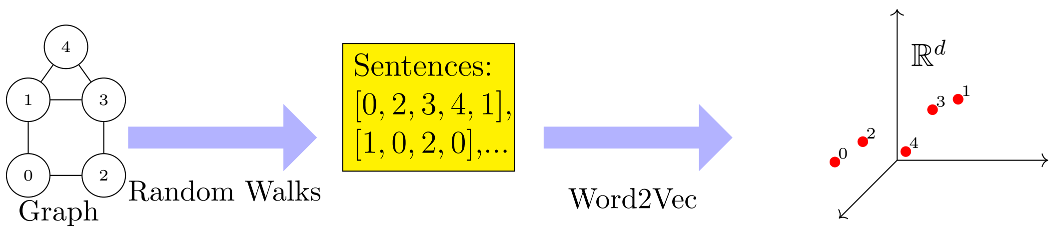



**Figure 4.10**: *A schematic illustration of the Node2Vec algorithm.*

For the training loop, we used Adam optimizer with a scheduler (factor=0.5) and

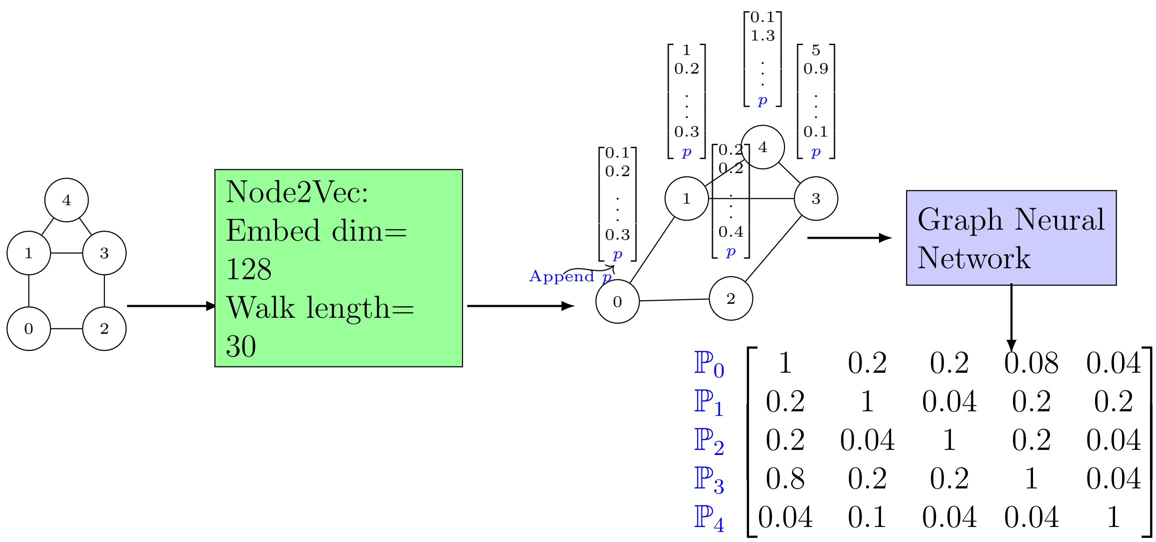



**Figure 4.11**: *A schematic illustration of the model architecture.*

patience equal to 5. We allowed early stopping with patience equal to 10 on validation loss improvement. The training loop configurations used for the model are listed in Table 4.3.

### 4.4.7 Results

The model was stopped after 60 epochs with train mean squared error (MSE) of 0.0012 and validation loss of 0.0021. The test MSE was 0.0017. Some sample predictions on six randomly chosen samples are shown in Figure 4.13. The loss curves as shown in Figure 4.12 show that the model is not overfitting.

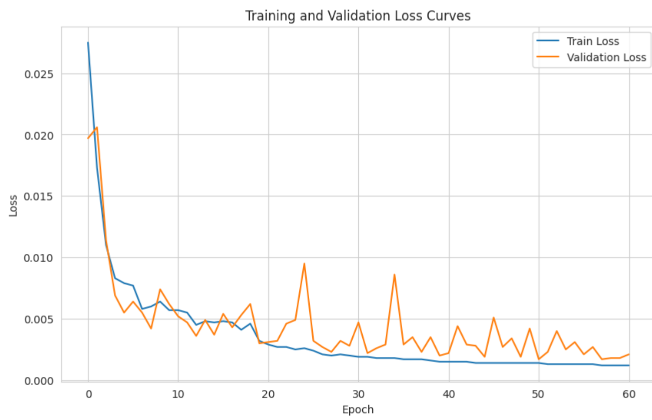



**Figure 4.12**: *Training and validation loss curves for 60 epochs.*

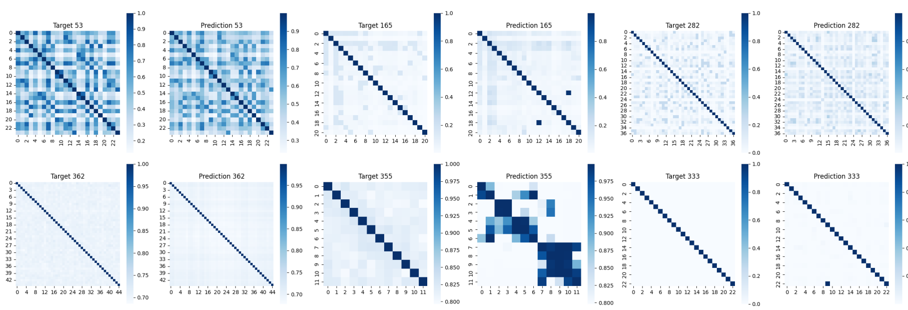



**Figure 4.13**: *Target and Predication heat map of percolation matrix for six randomly chosen samples.*

# Chapter 5

# Conclusion and Future Directions

## 5.1 Conclusion

In this thesis, we introduced a transformation rule called the switch identity for reducing graphs when computing the 2-modulus of paths or the effective resistance. We believe that this identity could be particularly useful for answering questions on the behavior of effective resistance in structured graphs [86]. We also studied the bound on the algebraic connectivity of graphs using the effective resistance diameter and the conditions under which equality holds.

We used the work of Lee et al. [30] to obtain the higher order Cheeger inequalities for the combinatorial Laplacian eigenvalues and the isoperimetric ratio. Using this, we obtained bounds on the eigenvalues of the combinatorial Laplacian using the strength of graphs.

We trained graph neural networks to solve two problems: the minimum feasible partition problem (P) and the site percolation influence computation (#P-hard). Note that by solving the minimum feasible partition we also derive the strength. We believe that both models can be improved using different node features and GNN architectures. Moreover, these results will hopefully inspire the application of GNNs to other problems in mathematics and combinatorial optimization.

## 5.2 Future Work

### 5.2.1 Improving Bounds on Strength

We showed in Section 3.4 that equality holds in the lower bound inequality (3.17) for $P_2$ (path graph on 2 nodes). Hence, the lower bound is tight. But this inequality can be improved in at least two ways: show sharper inequalities for certain families of graphs, and consider properties other than edge connectedness to include in the lower bound for $m_k(G)$. For the upper bound, we do not have the exact conditions under which equality holds. Moreover, we do not know if this upper bound is tight, that is, if equality holds for some graphs.

For graphs with a strong community structure, for instance, cliques connected by a few edges, the upper bound in (3.22) and (3.23) is small as the eigenvalues are small in this case, but for general graphs, it would be reasonable to expect the upper bound on the strength to be a function of some subset of the Laplacian eigenvalues (not just individual eigenvalues). Finding the characteristics of such a function is not currently clear, but we can use neural network models such as in Chapter 4 to learn such a function from synthetic datasets.

### 5.2.2 Question on strength spectrum

Recall (3.21) :

$$\min_{k=2,\ldots,N} \frac{\lambda_k(G)}{4(k-1)} \leq m(G) - \frac{r}{2}. \tag{5.1}$$

A natural question is how to characterize the graphs using $\frac{\lambda_k(G)}{k-1}$. We refer to this set of numbers as the strength spectrum. Can we characterize graphs in which the strength spectrum is increasing or decreasing? In addition, what can this ratio tell us about the structure of the graph? We provide some graphs generated using the random geometric graph model and their corresponding strength spectrum.

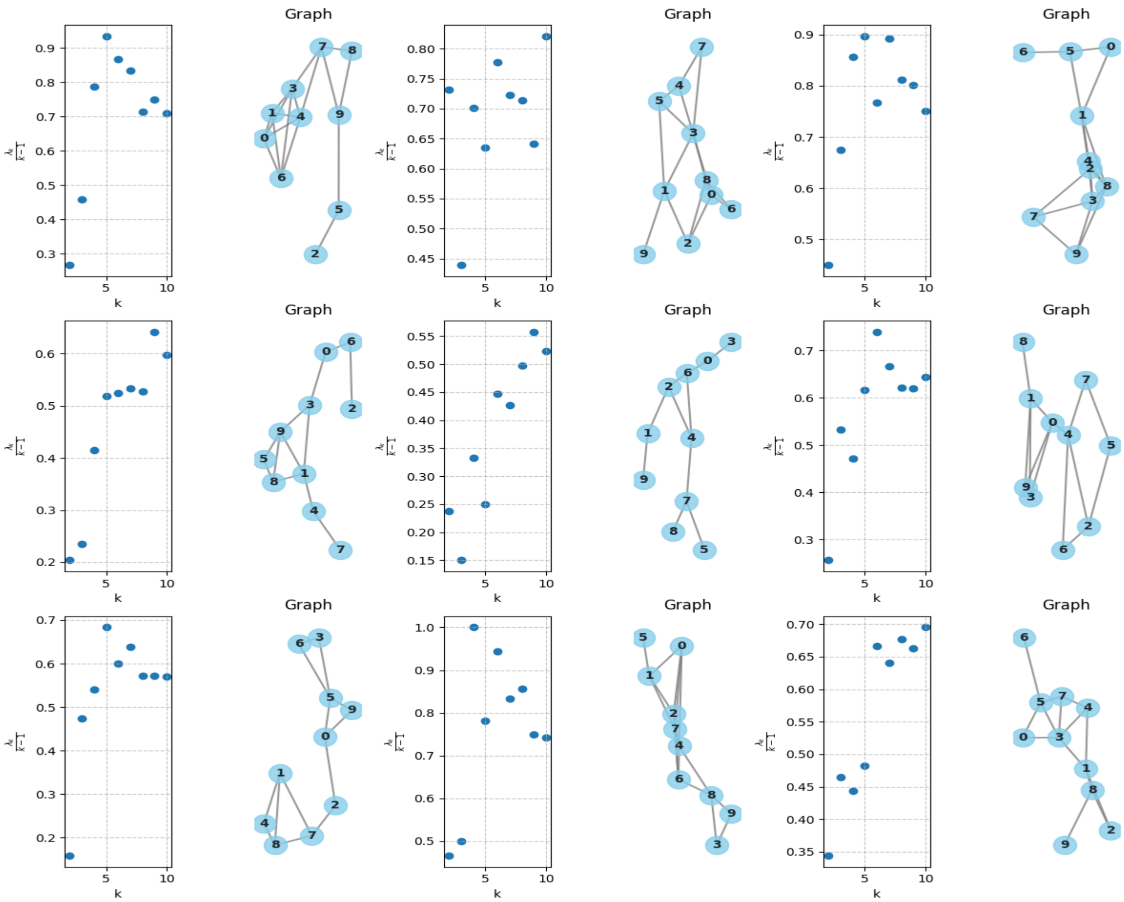



**Figure 5.1**: *Strength spectrum of some random geometric graphs.*

### 5.2.3 Improving graph neural network model for strength prediction

In Section 4.3 we trained a graph neural network model to identify the edges between the minimum feasible partition via the spanning tree modulus. There is an exact algorithm to predict these edges discovered by Albin et.al. [52] which can be used instead of the $\eta_{\max}$ values which is only an approximation.

Furthermore, we used graph convolutional networks which are less expressive than more general message passing neural networks or graph attention networks [87]. These newer architectures improve the prediction of the model and provide new insights into the saliency aliency presented in Section 4.3.

Finally, the results from the saliency analysis provide a strong relationship between the node resistance curvature and the minimum feasible partition problem. Finding an exact relationship between various notions of curvature and the minimum feasible partition problem is an open problem.

### 5.2.4 Embedding Question on the site percolation model

In Section 4.4 we used graph neural networks to predict the percolation matrix $\mathcal{P}$ for a graph. For this model, we used node2Vec to obtain the node input vectors and used graph neural networks to refine these embeddings for the downstream task. We propose a theoretical embedding question that can guide this line of research.

**Problem 85.** Consider the percolation matrix $\mathcal{P}(p)$ as a function from $[0, 1]$ to $\mathbb{M}_n(\mathbb{R})$. Note that $\mathcal{P}(0)$ is the identity matrix and $\mathcal{P}(1)$ is the rank one matrix consisting of all ones. Is it possible to write $\mathcal{P}(p)$ (considered as an element of $C([0, 1], \mathbb{M}_n(\mathbb{R}))$) as a Gram matrix? In other words, do there exist vectors $v_1(p), v_2(p), \ldots, v_n(p)$ such that $A(p) := \langle v_1(p), v_2(p), \ldots, v_n(p)\rangle \in \mathbb{M}_n(\mathbb{R})$ such that $\mathcal{P}(p) = A(p)^T A(p)$?

# Bibliography

[1] Prakash Gorroochurn. *Classic Problems of Probability*. John Wiley and Sons, Inc., 7 May 2012. doi: 10.1002/9781118314340.

[2] Antonio Ortega. *Introduction to Graph Signal Processing*. Cambridge University Press, 2022. ISBN 9781108428132.

[3] Fernando Gama, Elvin Isufi, Geert Leus, and Alejandro Ribeiro. Graphs, convolutions, and neural networks: From graph filters to graph neural networks. *IEEE Signal Processing Magazine*, 37(6):128–138, November 2020. ISSN 1558-0792. doi: 10.1109/msp.2020.3016143. URL http://dx.doi.org/10.1109/MSP.2020.3016143.

[4] Mark Kac. Can one hear the shape of a drum? *American Mathematical Monthly*, 73 (1):1–23, 1966.

[5] Carolyn Gordon and David Webb. You Can't Hear the Shape of a Drum. *American Scientist*, 84(1):46–55, January 1996.

[6] Boris Gutkin and Uzy Smilansky. Can one hear the shape of a graph? *Journal of Physics A: Mathematical and General*, 34(31):6061, jul 2001. doi: 10.1088/0305-4470/34/31/301. URL https://dx.doi.org/10.1088/0305-4470/34/31/301.

[7] L. Collatz and U. Sinogowitz. Spektren endlicher grafen. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 21:63–77, 1957. doi: 10.1007/BF02941924.

[8] S. Skiena. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Addison-Wesley, Reading, MA, 1990.

[9] Alexander Lubotzky. Expander graphs in pure and applied mathematics. *American Mathematical Society*, 49(1):113–162, 2012. doi: 10.1090/psp/04901. Article electronically published on November 2, 2011.

[10] M. S. Pinsker. On the complexity of a concentrator. In *Proceedings of the 7th International Teletraffic Congress*, pages 318/1–318/4. Academy of Sciences, Moscow, USSR, 1973.

[11] H. D. Simon. Partitioning of unstructured problems for parallel processing. *Computing Systems in Engineering*, 2(2-3):135–148, 1991.

[12] C. J. Alpert and A. B. Kahng. Recent directions in netlist partitioning: A survey. *Integration: The VLSI Journal*, 19(1-2):1–81, 1995.

[13] W. Donath and A. B. Hoffman. Lower bounds for the partitioning of graphs. *IBM Journal of Research and Development*, 17(5):420–425, 1973.

[14] L. Hagen and A. B. Kahng. New spectral methods for ratio cut partitioning and clustering. In *Proceedings of the IEEE International Conference on Computer-Aided Design (ICCAD)*, pages 10–13, 1992.

[15] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 731–737, 2000.

[16] Leo Grady and Eric L. Schwartz. Isoperimetric partitioning: A new algorithm for graph partitioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3): 469–475, 2006. doi: 10.1109/TPAMI.2006.60.

[17] Bojan Mohar. Isoperimetric numbers of graphs. *Journal of Combinatorial Theory, Series B*, 47(3):274–291, 1989.

[18] N. Alon and V. D. Milman. Isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, Series B*, 38:73–88, 1985.

[19] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(98):298–305, 1973.

[20] Bojan Mohar. Some applications of laplace eigenvalues of graphs. In *Graph Symmetry: Algebraic Methods and Applications*, NATO ASI Series, pages 225–275. Springer, Dordrecht, 1997. Notes taken by Martin Juvan.

[21] Alexander Lubotzky. Expander graphs in pure and applied mathematics. *Bulletin of the American Mathematical Society*, 49(1):113–162, 2012. doi: 10.1090/S0273-0979-2011-01359-3.

[22] Herman H. Goldstine. *The Computer from Pascal to von Neumann.* Princeton University Press, Princeton, NJ, 1972.

[23] William Aspray. *John von Neumann and the Origins of Modern Computing.* MIT Press, Cambridge, MA, 1990.

[24] Kenneth Appel and Wolfgang Haken. Every planar map is four colorable. part i: Discharging. *Illinois Journal of Mathematics*, 21(3):429–490, 1977.

[25] Guillaume Lample and François Charton. Deep learning for symbolic mathematics. In *International Conference on Learning Representations*, 2020.

[26] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Fidler, Pushmeet Kohli, Bernardino Battaglino, Andrew Fitzgibbon, and Charles Blundell. Advancing mathematics by guiding human intuition with ai. *Nature*, 600:70–74, 2021. doi: 10.1038/s41586-021-04086-x.

[27] Quentin Cappart, Didier Chételat, Elias B. Khalil, Andrea Lodi, Christopher Morris, and Petar Veličković. Combinatorial optimization and reasoning with graph neural networks. *Journal of Machine Learning Research*, 24:1–61, April 2023. Submitted 4/21; Revised 1/23; Published 4/23.

[28] Gregory F. Lawler. A self-avoiding random walk. *Duke Mathematical Journal*, 47(3): 655–693, 1980. doi: 10.1215/S0012-7094-80-04741-9.

[29] Gregory F. Lawler. Loop-erased random walk. In Maury Bramson and Rick Durrett, editors, *Perplexing Problems in Probability*, volume 44 of *Progress in Probability*, pages 197–217. Birkhäuser Boston, Boston, MA, 1999. doi: 10.1007/978-1-4612-2168-5_12.

[30] James R. Lee, Shayan Oveis Gharan, and Luca Trevisan. Multi-way spectral partitioning and higher-order cheeger inequalities, 2014. URL https://arxiv.org/abs/1111.1055.

[31] Nathan Albin and Pietro Poggi-Corradini. *Open Problems and New Directions for p-Modulus on Networks*, pages 129–141. Springer International Publishing, Cham, 2017. ISBN 978-3-319-73126-1. doi: 10.1007/978-3-319-73126-1_10. URL https://doi.org/10.1007/978-3-319-73126-1_10.

[32] C. Loewner. On the conformal capacity in space. *Journal of Mathematics and Mechanics*, 8:411–414, 1959.

[33] Lars Valerian Ahlfors. *Conformal Invariants: Topics in Geometric Function Theory*. McGraw-Hill, 1973.

[34] Nathan Albin, Megan Brunner, and Roberto Pérez. Modulus on graphs as a generalization of standard graph theoretic quantities. *Conform. Geom. Dyn.*, 19:298–317, 2015. ISSN 1088-4173. doi: 10.1090/ecgd/287. URL http://dx.doi.org/10.1090/ecgd/287.

[35] N. Albin, J. Clemens, N. Fernando, P. Pankka, and V. Suomala. Blocking duality for p-modulus on networks and applications. *Annali di Matematica Pura ed Applicata (1923 -)*, 198(3):973–999, jul 2019. doi: 10.1007/s10231-018-0806-0. URL https://doi.org/10.1007/s10231-018-0806-0.

[36] Max Goering, Nathan Albin, Pietro Poggi-Corradini, Caterina Scoglio, and Faryad Darabi Sahneh. Numerical investigation of metrics for epidemic processes on

graphs. In *2015 49th Asilomar Conference on Signals, Systems and Computers*, pages 1317–1322, 2015. doi: 10.1109/ACSSC.2015.7421356.

[37] Josh Ericson, Pietro Poggi-Corradini, and Hainan Zhang. Effective resistance on graphs and the epidemic quasimetric. *Involve, a Journal of Mathematics*, 7(1):97–124, 2014. doi: 10.2140/involve.2014.7.97.

[38] Heman Shakeri, Pietro Poggi-Corradini, Nathan Albin, and Caterina Scoglio. Network clustering and community detection using modulus of families of loops. *Phys. Rev. E*, 95:012316, Jan 2017. doi: 10.1103/PhysRevE.95.012316. URL https://link.aps.org/doi/10.1103/PhysRevE.95.012316.

[39] Behnaz Moradi, Heman Shakeri, Pietro Poggi-Corradini, and Michael J. Higgins. New methods for incorporating network cyclic structures to improve community detection. *CoRR*, abs/1805.07484, 2018.

[40] Nathan Albin, Kapila Kottegoda, and Pietro Poggi-Corradini. Spanning tree modulus for secure broadcast games. *Networks*, apr 2020. doi: 10.1002/net.21945. URL https://doi.org/10.1002/net.21945.

[41] Asaf Nachmias. Planar maps, random walks and circle packing, 2018.

[42] Nathan Albin and Pietro Poggi-Corradini. *Mathematics of Networks: Modulus Theory and Convex Optimization*. Routledge, 1st edition, 2025. ISBN 9780367457075.

[43] Nathan Albin, Megan Brunner, Roberto Perez, Pietro Poggi-Corradini, and Natalie Wiens. Modulus on graphs as a generalization of standard graph theoretic quantities. *Conformal Geometry and Dynamics*, 19:298–317, 2015. doi: 10.1090/ecgd/287.

[44] Daniel A. Spielman. *Spectral and Algebraic Graph Theory*. Yale University, 2025. URL http://cs-www.cs.yale.edu/homes/spielman/sagt. Incomplete Draft, dated February 16, 2025.

[45] Russell Lyons and Shayan Oveis Gharan. Sharp bounds on random walk eigenvalues via spectral embedding. *International Mathematics Research Notices*, 2018(24):7555–7605, 2018. doi: 10.1093/imrn/rnx082.

[46] Nathan Albin, Brian Benson, Abhinav Chand, and Pietro Poggi-Corradini. The spanning tree modulus and the laplace spectrum of a graph. Working Paper, 2025.

[47] Nathan Albin, Jason Clemens, Derek Hoare, Pietro Poggi-Corradini, Brandon Sit, and Sarah Tymochko. Fairest edge usage and minimum expected overlap for random spanning trees. *Discrete Mathematics*, 344(5):112282, 2021. ISSN 0012-365X. doi: https://doi.org/10.1016/j.disc.2020.112282. URL https://www.sciencedirect.com/science/article/pii/S0012365X20304684.

[48] Dan Gusfield. Connectivity and edge-disjoint spanning trees. *Information Processing Letters*, 16(2):87–89, 1983. ISSN 0020-0190. doi: https://doi.org/10.1016/0020-0190(83)90031-5. URL https://www.sciencedirect.com/science/article/pii/0020019083900315.

[49] C. St. J. A. Nash-Williams. Edge-disjoint spanning trees of finite graphs. *Journal of the London Mathematical Society*, s1-36(1):445–450, 1961. doi: 10.1112/jlms/s1-36.1.445.

[50] W. T. Tutte. On the problem of decomposing a graph into n connected factors. *Journal of the London Mathematical Society*, s1-36(1):221–230, 1961. doi: 10.1112/jlms/s1-36.1.221.

[51] William H. Cunningham. Optimal attack and reinforcement of a network. *J. ACM*, 32:549–561, 1985. doi: 10.1145/3828.38.

[52] Nathan Albin, Kapila Kottegoda, and Pietro Poggi-Corradini. An exact-arithmetic algorithm for spanning tree modulus. *Networks*, feb 2025. ISSN 0028-3045, 1097-0037. doi: 10.1002/net.22270. URL https://doi.org/10.1002/net.22270.

[53] Fan R. K. Chung. *Spectral Graph Theory*, volume 92 of *CBMS Regional Conference*

*Series in Mathematics*. American Mathematical Society, 1997. ISBN 978-0-8218-0315-8. URL https://doi.org/10.1090/cbms/092.

[54] Anupam Gupta, Robert Krauthgamer, and James R. Lee. Bounded geometries, fractals, and low-distortion embeddings. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '03, page 534, USA, 2003. IEEE Computer Society. ISBN 0769520405.

[55] James R. Lee and Assaf Naor. Extending lipschitz functions via random metric partitions. *Inventiones mathematicae*, 160(1):59–95, 2005. doi: 10.1007/s00222-004-0400-5. URL https://doi.org/10.1007/s00222-004-0400-5.

[56] James R. Lee, Shayan Oveis Gharan, and Luca Trevisan. Multiway spectral partitioning and higher-order Cheeger inequalities. *J. ACM*, 61(6):Art. 37, 30, 2014. ISSN 0004-5411. URL https://doi.org/10.1145/2665063.

[57] Frank Rosenblatt. The perceptron: A perceiving and recognizing automaton. Report, Project PARA, Cornell Aeronautical Laboratory, January 1957.

[58] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:303–314, 1989.

[59] Kurt Hornik, Maxwell B. Stinchcombe, and Halbert L. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[60] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4:251–257, 1991.

[61] Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a non-polynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1993.

[62] Mitchell Black, Zhengchao Wan, Amir Nayyeri, and Yusu Wang. Understanding

oversquashing in gnns through the lens of effective resistance, 2023. URL https://arxiv.org/abs/2302.06835.

[63] Luca Longo, Mario Brcic, Federico Cabitza, Jaesik Choi, Roberto Confalonieri, Javier Del Ser, Riccardo Guidotti, Yoichi Hayashi, Francisco Herrera, Andreas Holzinger, Richard Jiang, Hassan Khosravi, Freddy Lecue, Gianclaudio Malgieri, Andrés Páez, Wojciech Samek, Johannes Schneider, Timo Speith, and Simone Stumpf. Explainable artificial intelligence (xai) 2.0: A manifesto of open challenges and interdisciplinary research directions. *Information Fusion*, 106:102301, 2024. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2024.102301.

[64] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11:1803–1831, Jun 2010.

[65] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013. URL https://arxiv.org/abs/1312.6034.

[66] Mathew Penrose. *Random Geometric Graphs.* Oxford University Press, Oxford, 2003. ISBN 0198506260.

[67] E. N. Gilbert. Random graphs. *Annals of Mathematical Statistics*, 30(4):1141–1144, 1959. doi: 10.1214/aoms/1177706098.

[68] Nathan Albin, Jason Clemens, Derek Hoare, Pietro Poggi-Corradini, Brandon Sit, and Sarah Tymochko. Fairest edge usage and minimum expected overlap for random spanning trees. *preprint*, 2019.

[69] Nathan Albin and Pietro Poggi-Corradini. The modulus book. https://github.com/nathan-albin/modulus_book, 2025.

[70] Karel Devriendt, Andrea Ottolini, and Stefan Steinerberger. Graph curvature via resistance distance, 2023. URL https://arxiv.org/abs/2302.06021.

[71] Jürgen Jost and Shiping Liu. Ollivier's ricci curvature, local clustering and curvature dimension inequalities on graphs, 2013. URL https://arxiv.org/abs/1103.4037.

[72] Yann Ollivier. Ricci curvature of markov chains on metric spaces. *Journal of Functional Analysis*, 256:810–864, 2009.

[73] Adam Gosztolai and Alexis Arnaudon. Unfolding the multiscale structure of networks with dynamical ollivier–ricci curvature. *Nature Communications*, 12:4561, 2021. doi: 10.1038/s41467-021-24941-4.

[74] Chien-Chun Ni, Yu-Yao Lin, Feng Luo, and Jie Gao. Community detection on networks with ricci flow. *Scientific Reports*, 9:1–12, 2019. doi: 10.1038/s41598-019-45341-w.

[75] Jayson Sia, Edmond A. Jonckheere, and Paul Bogdan. Ollivier-ricci curvature-based method to community detection in complex networks. *Scientific Reports*, 9:9800, 2019. doi: 10.1038/s41598-019-46079-x.

[76] Heather Z. Brooks and Mason A. Porter. An opinion reproduction number for infodemics in a bounded-confidence content-spreading process on networks, 2024. URL https://arxiv.org/abs/2403.01066.

[77] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 57–66, 2001.

[78] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, page 137–146, New York, NY, USA, 2003. Association for Computing Machinery. ISBN 1581137370. doi: 10.1145/956750.956769. URL https://doi.org/10.1145/956750.956769.

[79] A.P. Santhakumaran. Median of a graph with respect to edges. *Discussiones Mathematicae Graph Theory*, 32(1):19–29, 2012. ISSN 2083-5892.

[80] C. Jordan. Sur les assemblages des lignes. *Journal für die reine und angewandte Mathematik*, 70:185–190, 1869. doi: 10.1515/crll.1869.70.185.

[81] R.B. Bapat, A.K. Lal, and Sukanta Pati. A q-analogue of the distance matrix of a tree. *Linear Algebra and its Applications*, 416(2):799–814, 2006. ISSN 0024-3795. doi: https://doi.org/10.1016/j.laa.2005.12.023. URL https://www.sciencedirect.com/science/article/pii/S0024379506000115.

[82] Daniel A. Spielman and Shang-Hua Teng. Solving sparse, symmetric, diagonally-dominant linear systems in time $o(m^{1.31})$, 2004. URL https://arxiv.org/abs/cs/0310036.

[83] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[84] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM, 2016.

[85] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 701–710, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2956-9. doi: 10.1145/2623330.2623732. URL http://doi.acm.org/10.1145/2623330.2623732.

[86] E. J. Evans and A. E. Francis. Algorithmic techniques for finding resistance distances on structured graphs, 2021. URL https://arxiv.org/abs/2108.07942.

[87] Bingxu Zhang, Changjun Fan, Shixuan Liu, Kuihua Huang, Xiang Zhao, Jincai Huang, and Zhong Liu. The expressive power of graph neural networks: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 37(3):1455–1474, March 2025. ISSN 2326-3865. doi: 10.1109/tkde.2024.3523700. URL http://dx.doi.org/10.1109/TKDE.2024.3523700.

[88] Karel Devriendt and Renaud Lambiotte. Discrete curvature on graphs from the effective resistance. *Journal of Physics: Complexity*, 3(2):025008, June 2022. ISSN 2632-072X. doi: 10.1088/2632-072x/ac730d. URL http://dx.doi.org/10.1088/2632-072x/ac730d.

[89] Linton C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977. doi: 10.2307/3033543.

[90] Alex Bavelas. Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, 22(6):725–730, 1950. doi: 10.1121/1.1906679.

[91] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998. doi: 10.1038/30918.

# Appendix A

# Some Topological Node Features

Let $G = (V, E)$ be a finite graph. We give definitions for some node attributes .

**Definition 86** (Node resistance curvature)**.** [88] The node resistance curvature is defined as

$$p_u := 1 - \frac{1}{2} \sum_{v: v \sim u} \mathcal{R}_{\text{eff}}(u, v).$$

**Definition 87** (Betweenness centrality)**.** [89] The betweenness centrality of a node $v$ is defined as follows:

$$g(v) := \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

where $\sigma_{st}$ is the total number of shortest paths from node $s$ to node $t$, and $\sigma_{st}(v)$ is the number of those shortest paths that pass through node $v$.

**Definition 88** (Closeness centrality)**.** [90] The closeness centrality of a node $v$ is defined as the reciprocal of the sum of the shortest path distances from $v$ to all other nodes:

$$C(u) := \frac{N - 1}{\sum_{v \in V} d(u, v)},$$

where $N$ is the total number of nodes in the graph and $d(u, v)$ is the shortest path distance from node $u$ to node $v$.

**Definition 89** (Clustering coefficient)**.** [91] The clustering coefficient of a node $u$ is given by the formula:

$$c_u = \frac{2T(u)}{\deg(u)(\deg(u) - 1)},$$

where $T(u)$ is the number of triangles through node $u$, and $\deg(u)$ is the degree of node $u$.