

## Introduction

*Daniel A. Spielman*

August 29, 2018

## 1.1 First Things

1. Please call me “Dan”. If such informality makes you uncomfortable, you can try “Professor Dan”. If that fails, I will also answer to “Prof. Spielman”.
2. If you are going to take this course, please sign up for it on Canvas. This is the only way you will get emails like “Problem 3 was false, so you don’t have to solve it”.
3. This class meets this coming Friday, September 31, but not on Labor Day, which is Monday, September 3.

## 1.2 Introduction

I have three objectives in this lecture: to give you an overview of the material we will cover this semester, to help you decide if this course is right for you, and to tell you how the course will work.

As the title suggests, this course is about the eigenvalues and eigenvectors of matrices associated with graphs, and their applications. I will never forget my amazement at learning that combinatorial properties of graphs could be revealed by an examination of the eigenvalues and eigenvectors of their associated matrices. I hope to both convey my amazement to you and to make it feel like common sense. I’m now shocked when any important property of a graph is not revealed by its eigenvalues and eigenvectors.

This class will fundamentally be a math class, and my emphasis is on material that I find beautiful and/or useful. I’ll present a lot of theorems, a few algorithms, and a bunch of open problems.

## 1.3 Mechanics

There is no book for this course, but I will produce notes for every lecture. **You should read the lecture notes.** They will often contain material that I did not have time to cover in class. They will sometimes contain extra expositions of elementary topics. I will try to make the notes available before lecture. Some students will want to print them out for reference during lecture.

Given that I am providing lecture notes, you might not need to take notes during lecture. I, however, take notes during every lecture that I attend. It helps me pay attention and remember what is going on. But, there are many different learning styles. You may prefer to just listen.

If you would like a book that covers some of the material from the course, I suggest one of

“Algebraic Graph Theory” by Chris Godsil and Gordon Royle,

“Spectral Graph Theory” by Fan Chung, or

“Algebraic Combinatorics” by Chris Godsil.

I expect to produce around 5 or 6 problem sets during the semester. Some of the problems I assign in these will be very hard. You will be allowed to work on them in small groups.

For some lectures, such as today’s, I have assigned a number of “exercises” at the end of the lecture notes. You should solve these on your own, as soon after lecture as possible. You should not hand them in. They are just to help you practice the material. Today’s exercises are a review of fundamental linear algebra. I will put the solutions to some of them on Canvas.

There will be no tests or exams.

### 1.3.1 This is a graduate course

As some undergrads are thinking about taking this course, I thought I should explain the main differences between an undergraduate course and a graduate course, and the differences in outlook between undergrads and graduate students.

Graduate school is essentially pass/fail. Graduate students either write a thesis and graduate, or they do not. Their grades in courses do not matter very much. Most are here because they think they might learn something in this course that they will find useful in their careers. This means that some of them will work very hard.

Graduate students are also occasionally occupied with other responsibilities, like teaching and research. For this reason, I will give students at least two weeks to complete the problems I assign. However, I recommend that you solve the easier problems immediately.

Graduate students routinely take courses for which they do not have all the prerequisite knowledge. I assume that they can learn anything elementary as needed. Wikipedia makes this much easier than it used to be.

Finally, graduate courses are not as “user friendly” as undergraduate courses. I make no guarantees about what will happen in this course. I may assign more or fewer problem sets than I have announced. I may completely change the topics that I decide to cover. You have been warned.

### 1.3.2 Other courses

I have adjusted my selection of material for this course to decrease overlap with others. For example, I am omitting some material that will be included in S&DS 684a: Statistical Inference on Graphs and S&DS 615b: Introduction to Random Matrix Theory.

## 1.4 Background: Graphs

First, we recall that a graph  $G = (V, E)$  is specified by its vertex<sup>1</sup> set,  $V$ , and edge set  $E$ . In an undirected graph, the edge set is a set of unordered pairs of vertices. Unless otherwise specified, all graphs will be undirected, simple (having no loops or multiple edges) and finite. We will sometimes assign weights to edges. These will usually be real numbers. If no weights have been specified, we view all edges as having weight 1. This is an arbitrary choice, and we should remember that it has an impact.

Graphs (also called “networks”) are typically used to model connections or relations between things, where “things” are vertices. However, I often prefer to think of the edges in a graph as being more important than the vertices. In this case, I may just specify an edge set  $E$ , and ignore the ambient vertex set.

Common “natural” examples of graphs are:

- Friendship graphs: people are vertices, edges exist between pairs of people who are friends (assuming the relation is symmetric).
- Network graphs: devices, routers and computers are vertices, edges exist between pairs that are connected.
- Circuit graphs: electronic components, such as transistors, are vertices: edges exist between pairs connected by wires.
- Protein-Protein Interaction graphs: proteins are vertices. Edges exist between pairs that interact. These should really have weights indicating the strength and nature of interaction. Most other graphs should to.

It is much easier to study abstract, mathematically defined graphs. For example,

- The path on  $n$  vertices. The vertices are  $\{1, \dots, n\}$ . The edges are  $(i, i+1)$  for  $1 \leq i < n$ .
- The ring on  $n$  vertices. The vertices are  $\{1, \dots, n\}$ . The edges are all those in the path, plus the edge  $(1, n)$ .
- The hypercube on  $2^k$  vertices. The vertices are elements of  $\{0, 1\}^k$ . Edges exist between vertices that differ in only one coordinate.

## 1.5 Matrices for Graphs

The naive view of a matrix is that it is essentially a spreadsheet—a table we use to organize numbers. This is like saying that a car is an enclosed metal chair with wheels. It says nothing about what it does!

---

<sup>1</sup>I will use the words “vertex” and “node” interchangeably. Sorry about that.

I will use matrices to do two things. First, I will view a matrix  $\mathbf{M}$  as providing a function that maps a vector  $\mathbf{x}$  to the vector  $\mathbf{M}\mathbf{x}$ . That is, I view  $\mathbf{M}$  as an operator. Second, I view a matrix  $\mathbf{M}$  as providing a function that maps a vector  $\mathbf{x}$  to a number  $\mathbf{x}^T \mathbf{M} \mathbf{x}$ . That is, I use  $\mathbf{M}$  to define a quadratic form.

### 1.5.1 A spreadsheet

We will usually write  $V$  for the set of vertices of a graph, and let  $n$  denote the number of vertices. There are times that we will need to order the vertices and assign numbers to them. In this case, they will usually be  $\{1, \dots, n\}$ . For example, if we wish to draw a matrix as a table, then we need to decide which vertex corresponds to which row and column.

The most natural matrix to associate with a graph  $G$  is its adjacency matrix<sup>2</sup>,  $\mathbf{M}_G$ , whose entries  $\mathbf{M}_G(a, b)$  are given by

$$\mathbf{M}_G(a, b) = \begin{cases} 1 & \text{if } (a, b) \in E \\ 0 & \text{otherwise.} \end{cases}$$

It is important to observe that I index the rows and columns of the matrix by vertices, rather than by number. Almost every statement that we make in this class will remain true under renaming of vertices. The first row of a matrix has no special importance. To understand this better see the exercises at the end of the lecture.

While the adjacency matrix is the most natural matrix to associate with a graph, I also find it the least useful. Eigenvalues and eigenvectors are most meaningful when used to understand a natural operator or a natural quadratic form. The adjacency matrix provides neither.

### 1.5.2 An operator

The most natural operator associated with a graph  $G$  is probably its diffusion operator. This operator describes the diffusion of stuff among the vertices of a graph and how random walks behave. We will save further discussion of this perspective for a later lecture.

### 1.5.3 A quadratic form

The most natural quadratic form associated with a graph is defined in terms of its Laplacian matrix,

$$\mathbf{L}_G \stackrel{\text{def}}{=} \mathbf{D}_G - \mathbf{M}_G,$$

where  $\mathbf{D}_G$  is the diagonal matrix in which  $\mathbf{D}_G(a, a)$  is the degree of vertex  $a$ . We will usually write  $d(a)$  for the degree of vertex  $a$ . In an unweighted graph, the degree of a vertex is the number of edges attached to it. In the case of a weighted graph, we use the *weighted degree*: the sum of the weights of the edges attached to the vertex  $a$ .

---

<sup>2</sup>I am going to try to always use the letter  $\mathbf{M}$  for the adjacency matrix, in contrast with my past practice which was to use  $\mathbf{A}$ . I will use letters like  $a$  and  $b$  to denote vertices.

Given a function on the vertices,  $\mathbf{x} \in \mathbb{R}^V$ , the Laplacian quadratic form is

$$\mathbf{x}^T \mathbf{L}_G \mathbf{x} = \sum_{(a,b) \in E} (\mathbf{x}(a) - \mathbf{x}(b))^2. \quad (1.1)$$

This form measures the smoothness of the function  $\mathbf{x}$ . It will be small if the function  $\mathbf{x}$  does not jump too much over any edge.

I use the notation  $\mathbf{x}(a)$  to denote the coordinate of vector  $\mathbf{x}$  corresponding to vertex  $a$ . Other people often use subscripts for this, like  $\mathbf{x}_a$ . I reserve subscripts for other purposes.

## 1.6 Background: Spectral Theory

I now review the highlights of the spectral theory for symmetric matrices. Almost all of the matrices we consider in this course will be symmetric or will be similar<sup>3</sup> to symmetric matrices.

We recall that a vector  $\psi$  is an eigenvector of a matrix  $\mathbf{M}$  with eigenvalue  $\lambda$  if

$$\mathbf{M}\psi = \lambda\psi. \quad (1.2)$$

That is,  $\lambda$  is an eigenvalue if and only if  $\lambda\mathbf{I} - \mathbf{M}$  is a singular matrix. Thus, the eigenvalues are the roots of the characteristic polynomial of  $\mathbf{M}$ :

$$\det(x\mathbf{I} - \mathbf{M}).$$

**Theorem 1.6.1.** [The Spectral Theorem] If  $\mathbf{M}$  is an  $n$ -by- $n$ , real, symmetric matrix, then there exist real numbers  $\lambda_1, \dots, \lambda_n$  and  $n$  mutually orthogonal unit vectors  $\psi_1, \dots, \psi_n$  and such that  $\psi_i$  is an eigenvector of  $\mathbf{M}$  of eigenvalue  $\lambda_i$ , for each  $i$ .

This is the great fact about symmetric matrices. If the matrix is not symmetric, it might not have  $n$  eigenvalues. And, even if it has  $n$  eigenvalues, their eigenvectors will not be orthogonal<sup>4</sup>. In fact, if  $\mathbf{M}$  is not symmetric, then its eigenvalues and eigenvectors might be the wrong thing to look at.

I remind you that the eigenvectors are not uniquely determined, although the eigenvalues are. If  $\psi$  is an eigenvector, then  $-\psi$  is as well. Some eigenvalues can be repeated. If  $\lambda_i = \lambda_{i+1}$ , then  $\psi_i + \psi_{i+1}$  will also be an eigenvector of eigenvalue  $\lambda_i$ . Generally, the eigenvectors of a given eigenvalue are only determined up to an orthogonal transformation.

**Fact 1.6.2.** The Laplacian matrix of a graph is positive semidefinite. That is, all its eigenvalues are nonnegative.

*Proof.* Let  $\psi$  be a unit eigenvector of  $\mathbf{L}$  of eigenvalue  $\lambda$ . Then,

$$\psi^T \mathbf{L} \psi = \psi^T \lambda \psi = \lambda = \sum_{(a,b) \in E} (\psi(a) - \psi(b))^2 > 0.$$

□

---

<sup>3</sup>A matrix  $\mathbf{M}$  is similar to a matrix  $\mathbf{B}$  if there is a non-singular matrix  $\mathbf{X}$  such that  $\mathbf{X}^{-1} \mathbf{M} \mathbf{X} = \mathbf{B}$ . In this case,  $\mathbf{M}$  and  $\mathbf{B}$  have the same eigenvalues. See the exercises at the end of this lecture.

<sup>4</sup>You can prove that if the eigenvectors are orthogonal, then the matrix is symmetric.

We always number the eigenvalues of the Laplacian from smallest to largest. Thus,  $\lambda_1 = 0$ . We will refer to  $\lambda_2$ , and in general  $\lambda_k$  for small  $k$ , as *low-frequency* eigenvalues.  $\lambda_n$  is a *high-frequency* eigenvalue. We will see why in a moment.

## 1.7 Overview of the course

We will begin the course by learning about the eigenvalues and eigenvectors of many special graphs. These will include simple graphs like paths, rings, stars, trees and hypercubes, and we will eventually get to Cayley graphs and Strongly Regular Graphs.

Before we get to any theorems, I would like to convince you that the eigenvalues and eigenvectors of graphs are meaningful by showing you some examples. I will do these examples in Julia using a Jupyter notebook. I include snippets of the code and the images they generate in this text, and have provided the notebook on the course webpage.

### 1.7.1 Paths

A path graph has vertices  $\{1, \dots, n\}$  and edges  $(i, i+1)$  for  $1 \leq i < n$ . Here is the adjacency matrix of a path graph on 4 vertices.

```
M = path_graph(4)
Matrix(M)
0.0  1.0  0.0  0.0
1.0  0.0  1.0  0.0
0.0  1.0  0.0  1.0
0.0  0.0  1.0  0.0
```

And, here is its Laplacian matrix

```
Matrix(lap(M))
1.0  -1.0  0.0  0.0
-1.0   2.0  -1.0  0.0
0.0  -1.0   2.0  -1.0
0.0   0.0  -1.0   1.0
```

Here are the eigenvalues of a longer path.

```
L = lap(path_graph(10))
E = eigen(Matrix(L))
println(E.values)

[0.0, 0.097887, 0.381966, 0.824429, 1.38197, 2.0, 2.61803, 3.17557, 3.61803, 3.90211]
```

The eigenvector of the zero-eigenvalue is a constant vector (up to numerical issues):

```
E.vectors[:,1]
```

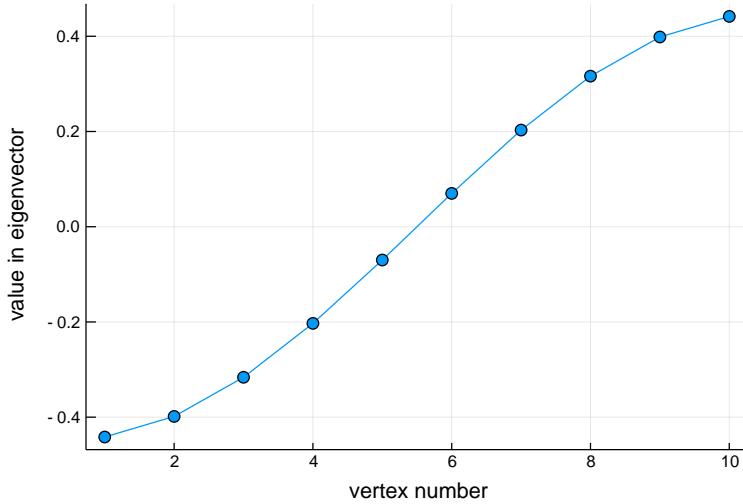
```
0.31622776601683755  
0.31622776601683716  
0.31622776601683766  
0.3162277660168381  
0.31622776601683855  
0.3162277660168381  
0.3162277660168385  
0.31622776601683805  
0.3162277660168378  
0.3162277660168378
```

The eigenvector of  $\lambda_2$  is the lowest frequency eigenvector, as we can see that it increases monotonically along the path:

```
v2 = E.vectors[:,2]
```

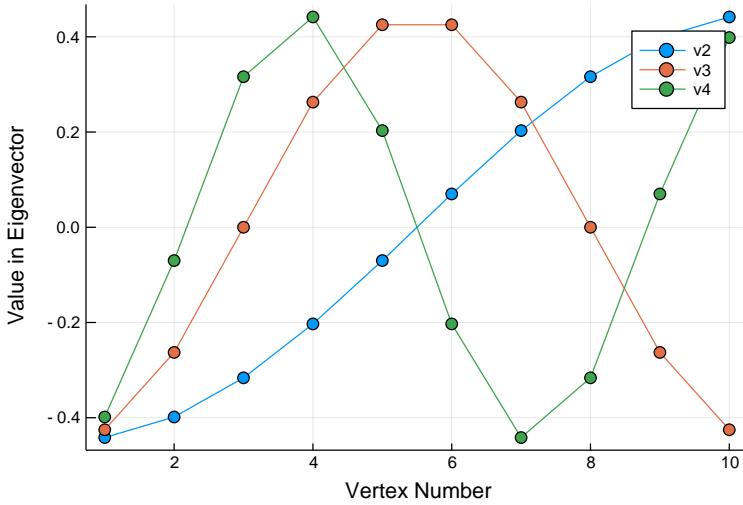
```
-0.44170765403093937  
-0.39847023129620024  
-0.316227766016838  
-0.20303072371134553  
-0.06995961957075425  
0.06995961957075386  
0.2030307237113457  
0.31622776601683766  
0.3984702312961997  
0.4417076540309382
```

Let's plot that.



```
plot(v2,marker=5,legend=false)
xlabel!("vertex number")
ylabel!("value in eigenvector")
```

The x-axis is the name/number of the vertex, and the y-axis is the value of the eigenvector at that vertex. Now, let's look at the next few eigenvectors.

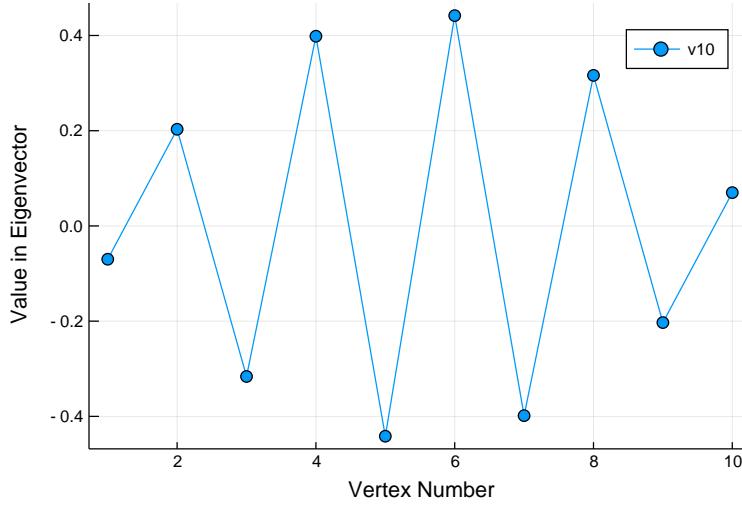


```
Plots.plot(E.vectors[:,2],label="v2",marker = 5)
Plots.plot!(E.vectors[:,3],label="v3",marker = 5)
Plots.plot!(E.vectors[:,4],label="v4",marker = 5)
xlabel!("Vertex Number")
ylabel!("Value in Eigenvector")
```

You may now understand why I refer to these as the low-frequency eigenvectors. The curves they trace out resemble the low-frequency modes of vibration of a string. The reason for this is

that the path graph can be viewed as a discretization of the string, and its Laplacian matrix is a discretization of the Laplace operator. We will relate the low-frequency eigenvalues to connectivity.

In contrast, the highest frequency eigenvalue alternates positive and negative with every vertex. We will show that these may be related to problems of graph coloring and finding independent sets.



```
Plots.plot(E.vectors[:,10],label="v10",marker=5)
xlabel!("Vertex Number")
ylabel!("Value in Eigenvector")
```

### 1.7.2 Spectral Graph Drawing

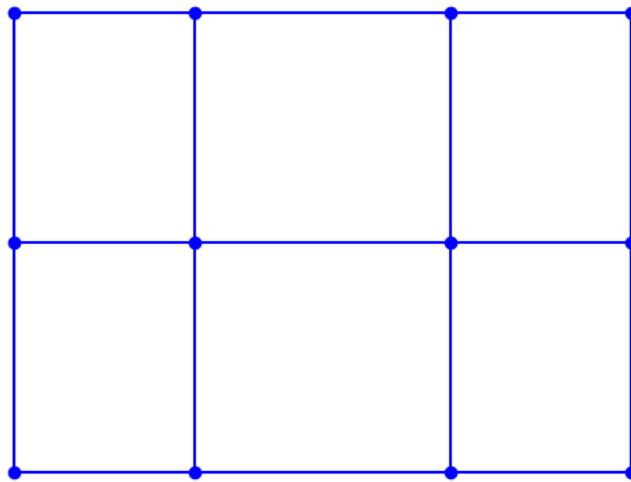
We can often use the low-frequency eigenvalues to obtain a nice drawing of a graph. For example, here is 3-by-4 grid graph, and its first two non-trivial eigenvectors. Looking at them suggests that they might provide nice coordinates for the vertices.

```
M = grid2(3,4)
L = lap(M)
E = eigen(Matrix(L))
V = E.vectors[:,2:3]

-0.377172  0.353553
-0.15623   0.353553
 0.15623   0.353553
 0.377172  0.353553
-0.377172  -1.66533e-16
-0.15623   -4.16334e-16
 0.15623   -5.82867e-16
 0.377172  2.77556e-16
```

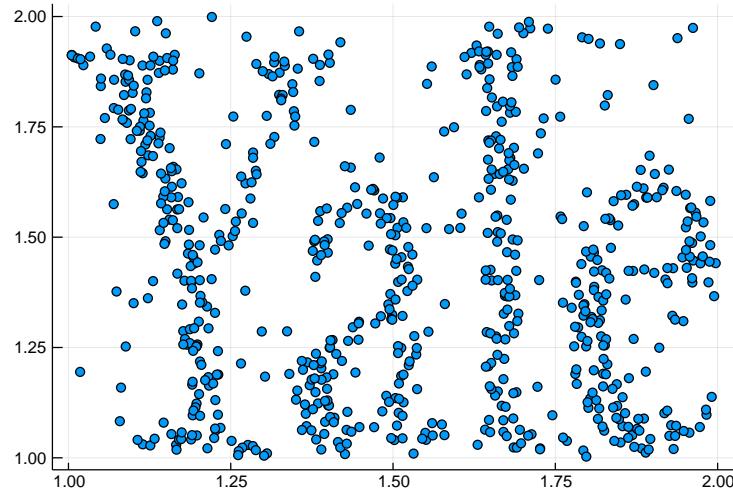
```
-0.377172 -0.353553
-0.15623 -0.353553
0.15623 -0.353553
0.377172 -0.353553
```

In the figure below, we use these eigenvectors to draw the graph. Vertex  $a$  be been plotted at coordinates  $\psi_2(a), \psi_3(a)$ . That is, we use  $\psi_2$  to provide a horizontal coordinate for every vertex, and  $\psi_3$  to obtain a vertical coordinate. We then draw the edges as straight lines.



```
plot_graph(M,V[:,1],V[:,2])
```

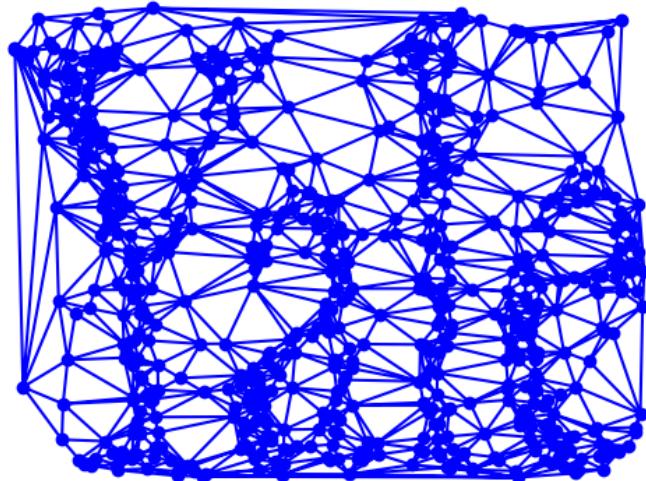
Let's do a fancier example that should convince you something interesting is going on. I begin by generating points by sampling them from the Yale logo.



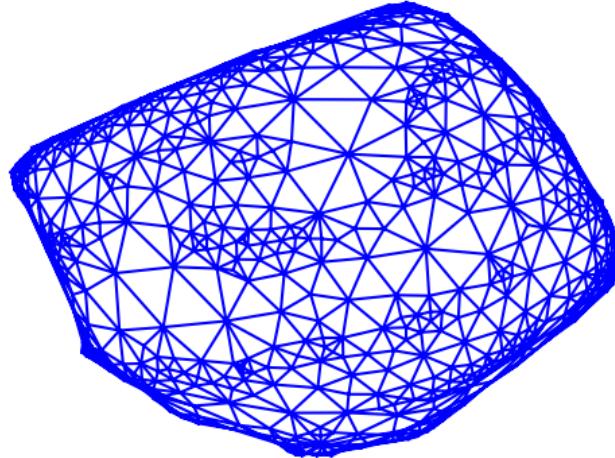
```
@load "yale.jld2"
scatter(xy[:,1],xy[:,2],legend=false)
```

I then construct a graph on them by forming their Delaunay triangulation. I won't get to teach about Delaunay triangulations during this course. But, they are terrific and I recommend that you look them up.

Since the vertices came with coordinates, it was easy to draw a nice picture of the graph. But, what if we just knew the graph, and not the coordinates? We could generate coordinates by computing two eigenvectors, and using each as a coordinate. Below, I plot vertex  $a$  at position  $\psi_2(a), \psi_3(a)$ , and again draw the edges as straight lines.



```
plot_graph(a,xy[:,1],xy[:,2])
```



```
plot_graph(a, v2,v3, dots=false)
```

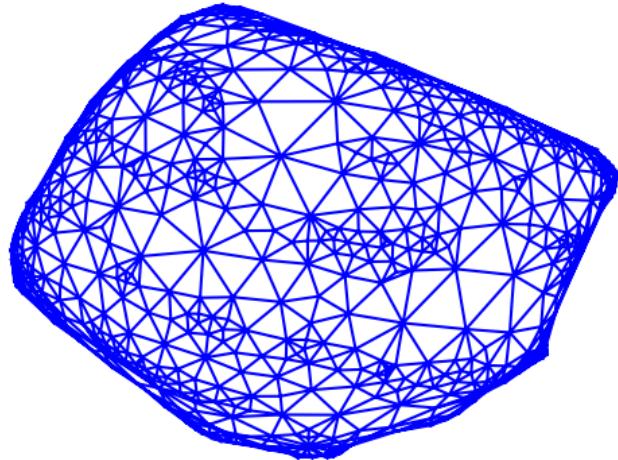
That's a great way to draw a graph if you start out knowing nothing about it. It's the first thing I do whenever I meet a strange graph. Note that the middle of the picture is almost planar, although edges do cross near the boundaries.

### 1.7.3 Graph Isomorphism

It is important to note that the eigenvalues do not change if we relabel the vertices. Moreover, if we permute the vertices then the eigenvectors are similarly permuted. That is, if  $\mathbf{P}$  is a permutation matrix, then

$$\mathbf{L}\boldsymbol{\psi} = \lambda\boldsymbol{\psi} \quad \text{if and only if} \quad (\mathbf{P}\mathbf{L}\mathbf{P}^T)(\mathbf{P}\boldsymbol{\psi}) = \lambda(\mathbf{P}\boldsymbol{\psi}),$$

because  $\mathbf{P}^T\mathbf{P} = \mathbf{I}$ . To prove it by experiment, let's randomly permute the vertices, and plot the permuted graph.



```

Random.seed!(1)
p = randperm(size(a,1))
M = a[p,p]
E = eigen(Matrix(lap(M)))
V = E.vectors[:,2:3]
plot_graph(M,V[:,1],V[:,2], dots=false)

```

Note that this picture is slightly different from the previous one: it has flipped vertically. That's because eigenvectors are only determined up to signs, and that's only if they have multiplicity 1. This gives us a very powerful heuristic for testing if one graph is a permutation of another (this is the famous “Graph Isomorphism Testing Problem”). First, check if the two graphs have the same sets of eigenvalues. If they don't, then they are not isomorphic. If they do, and the eigenvalues have multiplicity one, then draw the pictures above. If the pictures are the same, up to horizontal or vertical flips, and no vertex is mapped to the same location as another, then by lining up the pictures we can recover the permutation.

As some vertices can map to the same location, this heuristic doesn't always work. We will learn about it to the extent to which it does. In particular, we will see that if every eigenvalue of two graphs  $G$  and  $H$  have multiplicity 1, then we can efficiently test whether or not they are isomorphic.

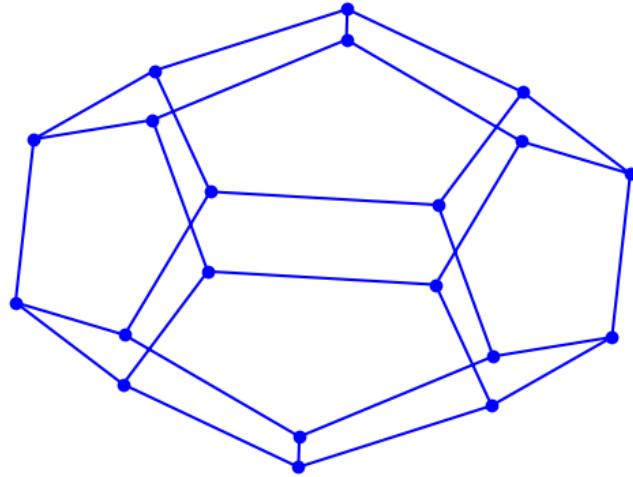
These algorithms have been extended to handle graph in which the multiplicity of every eigenvalue is bounded by a constant. But, there are graphs in which every non-trivial eigenvalue has large multiplicity. We will learn how to construct and analyze these, as they constitute fundamental examples and counter-examples to many natural conjectures. For example, here are the eigenvalues of a Latin Square Graph on 25 vertices. These are a type of Strongly Regular Graph.

```
M = latin_square_graph(5);
println(eigvals(Matrix(lap(M))))
```

All Latin Square Graphs of the same size have the same eigenvalues, whether or not they are isomorphic. We will learn some surprisingly fast (but still not polynomial time) algorithms for checking whether or not Strongly Regular Graphs are isomorphic.

#### 1.7.4 Platonic Solids

Of course, somme graphs are not meant to be drawn in 3 dimensions. For example let's try this with the dodecahedron.

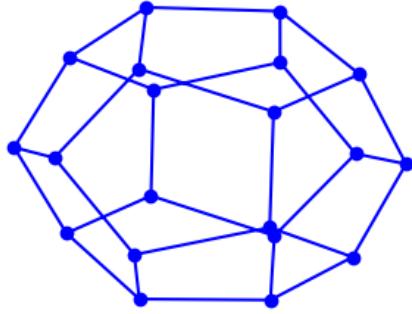


```
M = readIJV("dodec.txt")
spectral_drawing(M)
```

You will notice that this looks like what you would get if you squashed the dodecahedron down to the plane. The reason is that we really shouldn't be drawing this picture in two dimensions: the smallest non-zero eigenvalue of the Laplacian has multiplicity three.

```
E = eigen(Matrix(lap(M)))
println(E.values)
```

So, we can't reasonably choose just two eigenvectors. We should be choosing three that span the eigenspace. If we do, we would get the canonical representation of the dodecahedron in three dimensions.



```

x = E.vectors[:,2]
y = E.vectors[:,3]
z = E.vectors[:,4]
pygui(true)
plot_graph(M, x, y, z; setaxis=false)

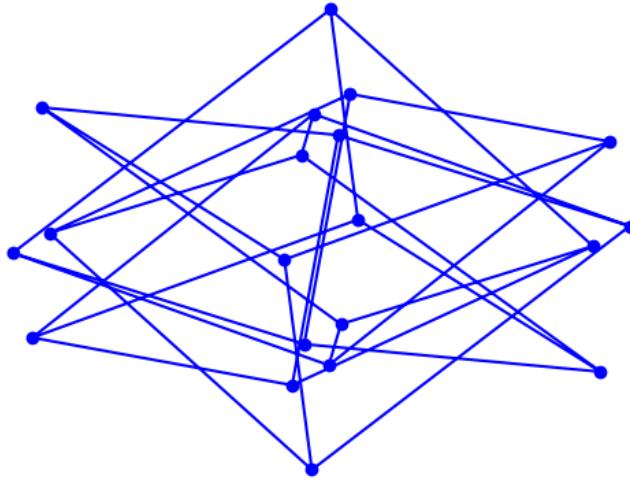
```

As you would guess, this happens for all Platonic solids. In fact, if you properly re-weight the edges, it happens for every graph that is the one-skeleton of a convex polytope. Let me state that more concretely. A weighted graph is a graph along with a weight function,  $w$ , mapping every edge to a positive number. The adjacency matrix of a weighted graph has the weights of edges as its entries, instead of 1s. The diagonal degree matrix of a weighted graph,  $D_G$ , has the weighted degrees on its diagonal. That is,

$$D_G(i, i) = \sum_{j:(i,j) \in E} w(i, j).$$

The Laplacian then becomes  $L_G = D_G - A_G$ . Given a convex polytope in  $\mathbb{R}^d$ , we can treat its 1-skeleton as a graph on its vertices. We will prove that there is a way of assigning weights to edges so that the second-smallest Laplacian eigenvalue has multiplicity  $d$ , and so that the corresponding eigenspace is spanned by the coordinate vectors of the vertices of the polytope.

Before we turn off the computer, let's take a look at the high-frequency eigenvectors of the dodecahedron.



```

x = E.vectors[:,11]
y = E.vectors[:,12]
z = E.vectors[:,10]
pygui(true)
plot_graph(M, x, y, z; setaxis=false)

```

### 1.7.5 The Fiedler Value

The second-smallest eigenvalue of the Laplacian matrix of a graph is zero if and only if the graph is disconnected. If  $G$  is disconnected, then we can partition it into two graphs  $G_1$  and  $G_2$  with no edges between them, and then write

$$\mathbf{L}_G = \begin{pmatrix} \mathbf{L}_{G_1} & 0 \\ 0 & \mathbf{L}_{G_2} \end{pmatrix}.$$

As the eigenvalues of  $\mathbf{L}_G$  are the union, with multiplicity, of the eigenvalues of  $\mathbf{L}_{G_1}$  and  $\mathbf{L}_{G_2}$  we see that  $\mathbf{L}_G$  inherits a zero eigenvalue from each. Conversely, if  $G$  is connected then we can show that the only vectors  $\mathbf{x}$  for which  $\mathbf{x}^T \mathbf{L}_G \mathbf{x} = 0$  are the constant vectors. If  $\mathbf{x}$  is not constant and  $G$  is connected then there must be an edge  $(a, b)$  for which  $\mathbf{x}(a) \neq \mathbf{x}(b)$ . And, this edge will contribute a positive term to the sum (1.1).

Fiedler suggested that we make this qualitative observation quantitative and think of  $\lambda_2$  as a measure of how well connected the graph is. For this reason, he called it the “Algebraic Connectivity” of a graph, and we call it the “Fiedler value”.

Fiedler proved that the further  $\lambda_2$  is from 0, the better connected the graph is. We will cover the ultimate extension of this result: Cheeger’s inequality.

In short, we say that a graph is poorly connected if one can cut off many vertices by removing only a few edges. We measure how poorly connected it is by the ratio of these quantities (almost).

Cheeger's inequality gives a tight connection between this ratio and  $\lambda_2$ . If  $\lambda_2$  is small, then for some  $t$ , the set of vertices

$$S_i \stackrel{\text{def}}{=} \{i : v_2(i) < t\}$$

may be removed by cutting much less than  $|S_i|$  edges. This spectral graph partitioning heuristic has proved very successful in practice.

In general, it will be interesting to turn qualitative statements like this into quantitative ones. For example, we will see that the smallest eigenvalue of the diffusion matrix is zero if and only if the graph is bipartite. One can relate the magnitude of this eigenvalue to how far a graph is from being bipartite.

### 1.7.6 Bounding Eigenvalues

We will often be interested in the magnitudes of certain eigenvalues. For this reason, we will learn multiple techniques for proving bounds on eigenvalues. The most prominent of these will be proofs by test vectors and proofs by comparison with simpler graphs.

### 1.7.7 Planar Graphs

We will prove that graphs that can be drawn nicely must have small Fiedler value, and we will prove very tight results for planar graphs.

We will also see how to use the graph Laplacian to draw planar graphs: Tutte proved that if one reasonably fixes the locations of the vertices on a face of a planar graph and then lets the others settle into the positions obtained by treating the edges as springs, then one obtains a planar drawing of the graph!

### 1.7.8 Random Walks on Graphs

Spectral graph theory is one of the main tools we use for analyzing random walks on graphs. We will spend a few lecture on this theory, connect it to Cheeger's inequality, and use tools developed to study random walks to derive a fascinating proof of Cheeger's inequality.

### 1.7.9 Expanders

We will be particularly interested in graphs that are very well connected. These are called *expanders*. Roughly speaking, expanders are sparse graphs (say a number of edges linear in the number of vertices), in which  $\lambda_2$  is bounded away from zero by a constant. They are among the most important examples of graphs, and play a prominent role in Theoretical Computer Science.

Expander graphs have numerous applications. We will see how to use random walks on expander graphs to construct pseudo-random generators *about which one can actually prove something*. We will also use them to construct good error-correcting codes.

Error-correcting codes and expander graphs are both fundamental objects of study in the field of Extremal Combinatorics and are extremely useful. If students in the class have not learned about these, I will teach about them. We will also use error-correcting codes to construct crude expander graphs.

We will learn at least one construction of good expanders. The best expanders are the Ramanujan graphs. These were first constructed by Margulis and Lubotzky, Phillips and Sarnak. We might finish the class by proving the existence of Ramanujan graphs.

### 1.7.10 Approximations of Graphs

We will ask what it means for one graph to approximate another. Given graphs  $G$  and  $H$ , we will measure how well  $G$  approximates  $H$  by the closeness of their Laplacian quadratic forms. We will see that expanders are precisely the sparse graphs that provide good approximations of the complete graph, and we will use this perspective for most of our analysis of expanders. We will show that every graph can be well-approximated by a sparse graph through a process called *sparsification*.

### 1.7.11 Solving equations in and computing eigenvalues of Laplacians

We will also ask how well a graph can be approximated by a tree, and see that low-stretch spanning-trees provide good approximations under this measure.

My motivation for this material is not purely graph-theoretic. Rather, it is inspired by the need to design fast algorithms for computing eigenvectors of Laplacian matrices and for solving linear equations in Laplacian matrices. This latter problem arises in numerous contexts, including the solution of elliptic PDEs by the finite element method, the solution of network flow problems by interior point algorithms, and in classification problems in Machine Learning.

In fact, our definition of graph approximation is designed to suit the needs of the Preconditioned Conjugate Gradient algorithm. We may finish the semester by learning how these algorithms work.

## 1.8 Eigenvalues and Optimization

One of the reasons that the eigenvalues of matrices have meaning is that they arise as the solution to natural optimization problems. We will spend a lot of time on this connection next lecture. For now, we start with one result in this direction. Observe that its proof does not require the spectral theorem.

**Theorem 1.8.1.** *Let  $\mathbf{M}$  be a symmetric matrix and let  $\mathbf{x}$  be a non-zero vector that maximizes the Rayleigh quotient with respect to  $\mathbf{M}$ :*

$$\frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

*Then,  $\mathbf{x}$  is an eigenvector of  $\mathbf{M}$  with eigenvalue equal to the Rayleigh quotient. Moreover, this eigenvalue is the largest eigenvalue of  $\mathbf{M}$ .*

*Proof.* We first observe that the maximum is achieved: As the Rayleigh quotient is homogeneous, it suffices to consider unit vectors  $\mathbf{x}$ . As the set of unit vectors is a closed and compact set, the maximum is achieved on this set.

Now, let  $\mathbf{x}$  be a non-zero vector that maximizes the Rayleigh quotient. We recall that the gradient of a function at its maximum must be the zero vector. Let's compute that gradient.

We have

$$\nabla \mathbf{x}^T \mathbf{x} = 2\mathbf{x},$$

and

$$\nabla \mathbf{x}^T \mathbf{M} \mathbf{x} = 2\mathbf{M} \mathbf{x}.$$

So,

$$\nabla \frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \frac{(\mathbf{x}^T \mathbf{x})(2\mathbf{M} \mathbf{x}) - (\mathbf{x}^T \mathbf{M} \mathbf{x})(2\mathbf{x})}{(\mathbf{x}^T \mathbf{x})^2}.$$

In order for this to be zero, we must have

$$\mathbf{M} \mathbf{x} = \frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \mathbf{x}.$$

That is, if and only if  $\mathbf{x}$  is an eigenvector of  $\mathbf{M}$  with eigenvalue equal to its Rayleigh quotient.  $\square$

## 1.9 Exercises

The following exercises are for your own practice. They are intended as a review of fundamental linear algebra. I have put the solutions in a separate file that you can find on Classes V2. I recommend that you try to solve all of these before you look at the solutions, so that you can get back in practice at doing linear algebra.

**1. Orthogonal eigenvectors.** Let  $\mathbf{M}$  be a symmetric matrix, and let  $\psi$  and  $\phi$  be vectors so that

$$\mathbf{M}\psi = \mu\psi \quad \text{and} \quad \mathbf{M}\phi = \nu\phi.$$

Prove that if  $\mu \neq \nu$  then  $\psi$  must be orthogonal to  $\phi$ . Note that your proof should exploit the symmetry of  $\mathbf{M}$ , as this statement is false otherwise.

**2. Invariance under permutations.**

Let  $\Pi$  be a permutation matrix. That is, there is a permutation  $\pi : V \rightarrow V$  so that

$$\Pi(u, v) = \begin{cases} 1 & \text{if } u = \pi(v), \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

Prove that if

$$\mathbf{M}\psi = \lambda\psi,$$

then

$$(\Pi \mathbf{M} \Pi^T)(\Pi\psi) = \lambda(\Pi\psi).$$

That is, permuting the coordinates of the matrix merely permutes the coordinates of the eigenvectors, and does not change the eigenvalues.

### 3. Invariance under rotations.

Let  $\mathbf{Q}$  be an orthonormal matrix. That is, a matrix such that  $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$ . Prove that if

$$\mathbf{M}\psi = \lambda\psi,$$

then

$$(\mathbf{Q}\mathbf{M}\mathbf{Q}^T)(\mathbf{Q}\psi) = \lambda(\mathbf{Q}\psi).$$

### 4. Similar Matrices.

A matrix  $\mathbf{M}$  is similar to a matrix  $\mathbf{B}$  if there is a non-singular matrix  $\mathbf{X}$  such that  $\mathbf{X}^{-1}\mathbf{M}\mathbf{X} = \mathbf{B}$ . Prove that similar matrices have the same eigenvalues.

### 5. Spectral decomposition.

Let  $\mathbf{M}$  be a symmetric matrix with eigenvalues  $\lambda_1, \dots, \lambda_n$  and let  $\psi_1, \dots, \psi_n$  be a corresponding set of orthonormal column eigenvectors. Let  $\Psi$  be the orthonormal matrix whose  $i$ th column is  $\psi_i$ . Prove that

$$\Psi^T \mathbf{M} \Psi = \Lambda,$$

where  $\Lambda$  is the diagonal matrix with  $\lambda_1, \dots, \lambda_n$  on its diagonal. Conclude that

$$\mathbf{M} = \Psi \Lambda \Psi^T = \sum_{i \in V} \lambda_i \psi_i \psi_i^T.$$

## 2.1 Eigenvalues and Optimization

The Rayleigh quotient of a vector  $\mathbf{x}$  with respect to a matrix  $\mathbf{M}$  is defined to be

$$\frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

At the end of the last class, I gave the following characterization of the largest eigenvalue of a symmetric matrix in terms of the Rayleigh quotient.

**Theorem 2.1.1.** *Let  $\mathbf{M}$  be a symmetric matrix and let  $\mathbf{x}$  be a non-zero vector that maximizes the Rayleigh quotient with respect to  $\mathbf{M}$ :*

$$\frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

*Then,  $\mathbf{x}$  is an eigenvector of  $\mathbf{M}$  with eigenvalue equal to the Rayleigh quotient. Moreover, this eigenvalue is the largest eigenvalue of  $\mathbf{M}$ .*

*Proof.* We first observe that the maximum is achieved: As the Rayleigh quotient is homogeneous, it suffices to consider unit vectors  $\mathbf{x}$ . As the set of unit vectors is a closed and compact set, the maximum is achieved on this set.

Now, let  $\mathbf{x}$  be a non-zero vector that maximizes the Rayleigh quotient. We recall that the gradient of a function at its maximum must be the zero vector. Let's compute that gradient.

We have

$$\nabla \mathbf{x}^T \mathbf{x} = 2\mathbf{x},$$

and

$$\nabla \mathbf{x}^T \mathbf{M} \mathbf{x} = 2\mathbf{M} \mathbf{x}.$$

So,

$$\nabla \frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \frac{(\mathbf{x}^T \mathbf{x})(2\mathbf{M} \mathbf{x}) - (\mathbf{x}^T \mathbf{M} \mathbf{x})(2\mathbf{x})}{(\mathbf{x}^T \mathbf{x})^2}.$$

In order for this to be zero, we must have

$$\mathbf{M} \mathbf{x} = \frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \mathbf{x}.$$

That is, if and only if  $\mathbf{x}$  is an eigenvector of  $\mathbf{M}$  with eigenvalue equal to its Rayleigh quotient.  $\square$

Of course, the analogous characterization holds for the smallest eigenvalue. A substantial generalization of these characterizations is given by the Courant-Fischer Theorem. We will state it for the Laplacian, as that is the case we will consider for the rest of the lecture.

**Theorem 2.1.2** (Courant-Fischer Theorem). *Let  $\mathbf{L}$  be a symmetric matrix with eigenvalues  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . Then,*

$$\lambda_k = \min_{\substack{S \subseteq \mathbb{R}^n \\ \dim(S)=k}} \max_{\mathbf{x} \in S} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \max_{\substack{T \subseteq \mathbb{R}^n \\ \dim(T)=n-k+1}} \min_{\mathbf{x} \in T} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

For example, consider the case  $k = 1$ . In this case,  $S$  is just the span of  $\psi_1$  and  $T$  is all of  $\mathbb{R}^n$ . For general  $k$ , the proof reveals that the optimum is achieved when  $S$  is the span of  $\psi_1, \dots, \psi_k$  and when  $T$  is the span of  $\psi_{k+1}, \dots, \psi_n$ .

As many proofs in Spectral Graph Theory begin by expanding a vector in the eigenbasis of a matrix, we begin by carefully stating a key property of these expansions.

**Lemma 2.1.3.** *Let  $\mathbf{M}$  be a symmetric matrix with eigenvalues  $\mu_1, \dots, \mu_n$  and a corresponding orthonormal basis of eigenvectors  $\psi_1, \dots, \psi_n$ . Let  $\mathbf{x}$  be a vector and expand  $\mathbf{x}$  in the eigenbasis as*

$$\mathbf{x} = \sum_{i=1}^n c_i \psi_i.$$

*Then,*

$$\mathbf{x}^T \mathbf{M} \mathbf{x} = \sum_{i=1}^n c_i^2 \lambda_i.$$

You should check for yourself (or recall) that  $c_i = \mathbf{x}^T \psi_i$  (this is obvious if you consider the standard coordinate basis).

*Proof.* Compute:

$$\begin{aligned} \mathbf{x}^T \mathbf{M} \mathbf{x} &= \left( \sum_i c_i \psi_i \right)^T \mathbf{M} \left( \sum_j c_j \psi_j \right) \\ &= \left( \sum_i c_i \psi_i \right)^T \left( \sum_j c_j \lambda_j \psi_j \right) \\ &= \sum_{i,j} c_i c_j \lambda_j \psi_i^T \psi_j \\ &= \sum_i c_i^2 \lambda_i, \end{aligned}$$

as  $\psi_i^T \psi_j = 0$  for  $i \neq j$ . □

*Proof of 2.1.2.* Let  $\psi_1, \dots, \psi_n$  be an orthonormal set of eigenvectors of  $\mathbf{L}$  corresponding to  $\lambda_1, \dots, \lambda_n$ . We will just verify the first characterization of  $\lambda_k$ . The other is similar.

First, let's verify that  $\lambda_k$  is achievable. Let  $S_k$  be the span of  $\psi_1, \dots, \psi_k$ . We can expand every  $\mathbf{x} \in S_k$  as

$$\mathbf{x} = \sum_{i=1}^k c_i \psi_i.$$

Applying Lemma 2.1.3 we obtain

$$\frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \frac{\sum_{i=1}^k \lambda_i c_i^2}{\sum_{i=1}^k c_i^2} \leq \frac{\sum_{i=1}^k \lambda_k c_i^2}{\sum_{i=1}^k c_i^2} = \lambda_k.$$

To show that this is in fact the maximum, we will prove that for all subspaces  $S$  of dimension  $k$ ,

$$\max_{\mathbf{x} \in S} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \geq \lambda_k.$$

Let  $T_k$  be the span of  $\psi_k, \dots, \psi_n$ . As  $T_k$  has dimension  $n - k + 1$ , every  $S$  of dimension  $k$  has an intersection with  $T_k$  of dimension at least 1. So,

$$\max_{\mathbf{x} \in S} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \geq \max_{\mathbf{x} \in S \cap T_k} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

Any such  $\mathbf{x}$  may be expressed as

$$\mathbf{x} = \sum_{i=k}^n c_i \psi_i,$$

and so

$$\frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \frac{\sum_{i=k}^n \lambda_i c_i^2}{\sum_{i=k}^n c_i^2} \geq \frac{\sum_{i=k}^n \lambda_k c_i^2}{\sum_{i=k}^n c_i^2} = \lambda_k.$$

□

We give one last characterization of the eigenvalues and eigenvectors of a symmetric matrix. Its proof is similar, so we will save it for an exercise.

**Theorem 2.1.4.** *Let  $\mathbf{L}$  be an  $n \times n$  symmetric matrix with eigenvalues  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$  with corresponding eigenvectors  $\psi_1, \dots, \psi_n$ . Then,*

$$\lambda_i = \min_{\mathbf{x} \perp \psi_1, \dots, \psi_{i-1}} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}},$$

and the eigenvectors satisfy

$$\psi_i = \arg \min_{\mathbf{x} \perp \psi_1, \dots, \psi_{i-1}} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

## 2.2 Drawing with Laplacian Eigenvalues

I will now explain the motivation for the pictures of graphs that I drew last lecture using the Laplacian eigenvalues. Well, the real motivation was just to convince you that eigenvectors are cool. The following is the technical motivation. It should come with the caveat that it does not produce nice pictures of all graphs. In fact, it produces bad pictures of most graphs. But, it is still the first thing I always try when I encounter a new graph that I want to understand.

This approach to using eigenvectors to draw graphs was suggested by Hall [Hal70] in 1970.

To explain Hall's approach, I'll begin by describing the problem of drawing a graph on a line. That is, mapping each vertex to a real number. It isn't easy to see what a graph looks like when you do this, as all of the edges sit on top of one another. One can fix this either by drawing the edges of the graph as curves, or by wrapping the line around a circle.

Let  $\mathbf{x} \in \mathbb{R}^V$  be the vector that describes the assignment of a real number to each vertex. We would like most pairs of vertices that are neighbors to be close to one another. So, Hall suggested that we choose an  $\mathbf{x}$  minimizing

$$\mathbf{x}^T \mathbf{L} \mathbf{x}. \quad (2.1)$$

Unless we place restrictions on  $\mathbf{x}$ , the solution will be degenerate. For example, all of the vertices could map to 0. To avoid this, and to fix the scale of the embedding overall, we require

$$\sum_{a \in V} \mathbf{x}(a)^2 = \|\mathbf{x}\|^2 = 1. \quad (2.2)$$

Even with this restriction, another degenerate solution is possible: it could be that every vertex maps to  $1/\sqrt{n}$ . To prevent this from happening, we add the additional restriction that

$$\sum_a \mathbf{x}(a) = \mathbf{1}^T \mathbf{x} = 0. \quad (2.3)$$

On its own, this restriction fixes the shift of the embedding along the line. When combined with (2.2), it guarantees that we get something interesting.

As  $\mathbf{1}$  is the eigenvector of the 0 eigenvalue of the Laplacian, the nonzero vectors that minimize (2.1) subject to (2.2) and (2.3) are the unit eigenvectors of the Laplacian of eigenvalue  $\lambda_2$ .

Of course, we really want to draw a graph in two dimensions. So, we will assign two coordinates to each vertex given by  $\mathbf{x}$  and  $\mathbf{y}$ . As opposed to minimizing (2.1), we will minimize

$$\sum_{(a,b) \in E} \left\| \begin{pmatrix} \mathbf{x}(a) \\ \mathbf{y}(a) \end{pmatrix} - \begin{pmatrix} \mathbf{x}(b) \\ \mathbf{y}(b) \end{pmatrix} \right\|^2.$$

This turns out not to be so different from minimizing (2.1), as it equals

$$\sum_{(a,b) \in E} (\mathbf{x}(a) - \mathbf{x}(b))^2 + (\mathbf{y}(a) - \mathbf{y}(b))^2 = \mathbf{x}^T \mathbf{L} \mathbf{x} + \mathbf{y}^T \mathbf{L} \mathbf{y}.$$

As before, we impose the scale conditions

$$\|\mathbf{x}\|^2 = 1 \quad \text{and} \quad \|\mathbf{y}\|^2 = 1,$$

and the centering constraints

$$\mathbf{1}^T \mathbf{x} = 0 \quad \text{and} \quad \mathbf{1}^T \mathbf{y} = 0.$$

However, this still leaves us with the degenerate solution  $\mathbf{x} = \mathbf{y} = \psi_2$ . To ensure that the two coordinates are different, Hall introduced the restriction that  $\mathbf{x}$  be orthogonal to  $\mathbf{y}$ . One can use the spectral theorem to prove that the solution is then given by setting  $\mathbf{x} = \psi_2$  and  $\mathbf{y} = \psi_3$ , or by taking a rotation of this solution (this is a problem on the first problem set).

## 2.3 Isoperimetry and $\lambda_2$

Computer Scientists are often interested in cutting, partitioning, and clustering graphs. Their motivations range from algorithm design to data analysis. We will see that the second-smallest eigenvalue of the Laplacian is intimately related to the problem of dividing a graph into two pieces without cutting too many edges.

Let  $S$  be a subset of the vertices of a graph. One way of measuring how well  $S$  can be separated from the graph is to count the number of edges connecting  $S$  to the rest of the graph. These edges are called the *boundary* of  $S$ , which we formally define by

$$\partial(S) \stackrel{\text{def}}{=} \{(a, b) \in E : a \in S, b \notin S\}.$$

We are less interested in the total number of edges on the boundary than in the ratio of this number to the size of  $S$  itself. For now, we will measure this in the most natural way—by the number of vertices in  $S$ . We will call this ratio the *isoperimetric ratio* of  $S$ , and define it by

$$\theta(S) \stackrel{\text{def}}{=} \frac{|\partial(S)|}{|S|}.$$

The *isoperimetric number* of a graph is the minimum isoperimetric number over all sets of at most half the vertices:

$$\theta_G \stackrel{\text{def}}{=} \min_{|S| \leq n/2} \theta(S).$$

We will now derive a lower bound on  $\theta_G$  in terms of  $\lambda_2$ . We will present an upper bound, known as Cheeger's Inequality, in a later lecture.

**Theorem 2.3.1.** *For every  $S \subset V$*

$$\theta(S) \geq \lambda_2(1 - s),$$

where  $s = |S| / |V|$ . In particular,

$$\theta_G \geq \lambda_2/2.$$

*Proof.* As

$$\lambda_2 = \min_{\mathbf{x}: \mathbf{x}^T \mathbf{1} = 0} \frac{\mathbf{x}^T \mathbf{L}_G \mathbf{x}}{\mathbf{x}^T \mathbf{x}},$$

for every non-zero  $\mathbf{x}$  orthogonal to  $\mathbf{1}$  we know that

$$\mathbf{x}^T \mathbf{L}_G \mathbf{x} \geq \lambda_2 \mathbf{x}^T \mathbf{x}.$$

To exploit this inequality, we need a vector related to the set  $S$ . A natural choice is  $\chi_S$ , the characteristic vector of  $S$ ,

$$\chi_S(a) = \begin{cases} 1 & \text{if } a \in S \\ 0 & \text{otherwise.} \end{cases}$$

We find

$$\chi_S^T \mathbf{L}_G \chi_S = \sum_{(a,b) \in E} (\chi_S(a) - \chi_S(b))^2 = |\partial(S)|.$$

However,  $\chi_S$  is not orthogonal to  $\mathbf{1}$ . To fix this, use

$$\mathbf{x} = \chi_S - s\mathbf{1},$$

so

$$\mathbf{x}(a) = \begin{cases} 1-s & \text{for } a \in S, \text{ and} \\ -s & \text{otherwise.} \end{cases}$$

We have  $\mathbf{x}^T \mathbf{1} = 0$ , and

$$\mathbf{x}^T \mathbf{L}_G \mathbf{x} = \sum_{(a,b) \in E} ((\chi_S(a) - s) - (\chi_S(b) - s))^2 = |\partial(S)|.$$

To finish the proof, we compute

$$\mathbf{x}^T \mathbf{x} = |S|(1-s)^2 + (|V| - |S|)s^2 = |S|(1-2s+s^2) + |S|s - |S|s^2 = |S|(1-s).$$

This gives

$$\lambda_2 \leq \frac{\chi_S^T \mathbf{L}_G \chi_S}{\chi_S^T \chi_S} = \frac{|\partial(S)|}{|S|(1-s)}.$$

□

This theorem says that if  $\lambda_2$  is big, then  $G$  is very well connected: the boundary of every small set of vertices is at least  $\lambda_2$  times something just slightly smaller than the number of vertices in the set.

We will use the computation in the last line of that proof often, so we will make it a claim.

**Claim 2.3.2.** *Let  $S \subseteq V$  have size  $s|V|$ . Then*

$$\|\chi_S - s\mathbf{1}\|^2 = s(1-s)|V|.$$

## 2.4 Exercises

The following exercises are for your own practice. They are intended as a review of fundamental linear algebra. I will put the solutions in a separate file that you can find on Canvas. I recommend that you try to solve all of these before you look at the solutions, so that you can get back in practice at doing linear algebra.

### 1. Characterizing Eigenvalues.

Prove Theorem 2.1.4.

### 2. Traces.

Recall that the trace of a matrix  $\mathbf{A}$ , written  $\text{Tr}(\mathbf{A})$ , is the sum of the diagonal entries of  $\mathbf{A}$ . Prove that for two matrices  $\mathbf{A}$  and  $\mathbf{B}$ ,

$$\text{Tr}(\mathbf{AB}) = \text{Tr}(\mathbf{BA}).$$

Note that the matrices **do not** need to be square for this to be true. They can be rectangular matrices of dimensions  $n \times m$  and  $m \times n$ .

Use this fact and the previous exercise to prove that

$$\text{Tr}(\mathbf{A}) = \sum_{i=1}^n \lambda_i,$$

where  $\lambda_1, \dots, \lambda_n$  are the eigenvalues of  $\mathbf{A}$ . You are probably familiar with this fact about the trace, or it may have been the definition you were given. This is why I want you to remember how to prove it.

### 3. The Characteristic Polynomial

Let  $\mathbf{M}$  be a symmetric matrix. Recall that the eigenvalues of  $\mathbf{M}$  are the roots of the characteristic polynomial of  $\mathbf{M}$ :

$$p(x) \stackrel{\text{def}}{=} \det(x\mathbf{I} - \mathbf{M}) = \prod_{i=1}^n (x - \mu_i).$$

Write

$$p(x) = \sum_{k=0}^n x^{n-k} c_k (-1)^k.$$

Prove that

$$c_k = \sum_{S \subseteq [n], |S|=k} \det(\mathbf{M}(S, S)).$$

Here, we write  $[n]$  to denote the set  $\{1, \dots, n\}$ , and  $\mathbf{M}(S, S)$  to denote the submatrix of  $\mathbf{M}$  with rows and columns indexed by  $S$ .

### 4. Reversing products.

Let  $\mathbf{M}$  be a  $d$ -by- $n$  matrix. Prove that the multiset of nonzero eigenvalues of  $\mathbf{MM}^T$  is the same as the multiset of nonzero eigenvalues of  $\mathbf{M}^T\mathbf{M}$ .

## References

- [Hal70] K. M. Hall. An r-dimensional quadratic placement algorithm. *Management Science*, 17:219–229, 1970.

## Fundamental Graphs

*Daniel A. Spielman*

September 5, 2018

### 3.1 Overview

We will bound and derive the eigenvalues of the Laplacian matrices of some fundamental graphs, including complete graphs, star graphs, ring graphs, path graphs, and products of these that yield grids and hypercubes. As all these graphs are connected, they all have eigenvalue zero with multiplicity one. We will have to do some work to compute the other eigenvalues.

We derive some meaning from the eigenvalues by using them to bound isoperimetric numbers of graphs, which I recall are defined by

$$\theta(S) \stackrel{\text{def}}{=} \frac{|\partial(S)|}{|S|}.$$

We bound this using the following theorem from last lecture.

**Theorem 3.1.1.** *For every  $S \subset V$*

$$\theta(S) \geq \lambda_2(1 - s),$$

where  $s = |S| / |V|$ . In particular,

$$\theta_G \geq \lambda_2/2.$$

### 3.2 The Laplacian Matrix

We begin this lecture by establishing the equivalence of multiple expressions for the Laplacian. These will be necessary to derive its eigenvalues.

The Laplacian Matrix of a weighted graph  $G = (V, E, w)$ ,  $w : E \rightarrow \mathbb{R}^+$ , is designed to capture the Laplacian quadratic form:

$$\mathbf{x}^T \mathbf{L}_G \mathbf{x} = \sum_{(a,b) \in E} w_{a,b} (\mathbf{x}(a) - \mathbf{x}(b))^2. \quad (3.1)$$

We will now use this quadratic form to derive the structure of the matrix. To begin, consider a graph with just two vertices and one edge. Let's call it  $G_{1,2}$ . We have

$$\mathbf{x}^T \mathbf{L}_{G_{1,2}} \mathbf{x} = (\mathbf{x}(1) - \mathbf{x}(2))^2. \quad (3.2)$$

Consider the vector  $\delta_1 - \delta_2$ , where by  $\delta_i$  I mean the elementary unit vector with a 1 in coordinate  $i$ . We have

$$\mathbf{x}(1) - \mathbf{x}(2) = \delta_1^T \mathbf{x} - \delta_2^T \mathbf{x} = (\delta_1 - \delta_2)^T \mathbf{x},$$

so

$$(\mathbf{x}(1) - \mathbf{x}(2))^2 = ((\delta_1 - \delta_2)^T \mathbf{x})^2 = \mathbf{x}^T (\delta_1 - \delta_2) (\delta_1 - \delta_2)^T \mathbf{x} = \mathbf{x}^T \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \mathbf{x}.$$

Thus,

$$\mathbf{L}_{G_{1,2}} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Now, let  $G_{a,b}$  be the graph with just one edge between  $u$  and  $v$ . It can have as many other vertices as you like. The Laplacian of  $G_{a,b}$  can be written in the same way:  $\mathbf{L}_{G_{a,b}} = (\delta_a - \delta_b)(\delta_a - \delta_b)^T$ . This is the matrix that is zero except at the intersection of rows and columns indexed by  $u$  and  $v$ , where it looks like

$$\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Summing the matrices for every edge, we obtain

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

You should check that this agrees with the definition of the Laplacian from the first class:

$$\mathbf{L}_G = \mathbf{D}_G - \mathbf{A}_G,$$

where

$$\mathbf{D}_G(a, a) = \sum_b w_{a,b}.$$

This formula turns out to be useful when we view the Laplacian as an operator. For every vector  $\mathbf{x}$  we have

$$(\mathbf{L}_G \mathbf{x})(a) = d(a) \mathbf{x}(a) - \sum_{(a,b) \in E} w_{a,b} \mathbf{x}(b) = \sum_{(a,b) \in E} w_{a,b} (\mathbf{x}(a) - \mathbf{x}(b)). \quad (3.3)$$

### 3.3 The complete graph

The complete graph on  $n$  vertices,  $K_n$ , has edge set  $\{(a, b) : a \neq b\}$ .

**Lemma 3.3.1.** *The Laplacian of  $K_n$  has eigenvalue 0 with multiplicity 1 and  $n$  with multiplicity  $n - 1$ .*

*Proof.* To compute the non-zero eigenvalues, let  $\psi$  be any non-zero vector orthogonal to the all-1s vector, so

$$\sum_a \psi(a) = 0. \quad (3.4)$$

We now compute the first coordinate of  $\mathbf{L}_{K_n} \psi$ . Using (3.3), we find

$$(\mathbf{L}_{K_n} \psi)(1) = \sum_{v \geq 2} (\psi(1) - \psi(v)) = (n-1)\psi(1) - \sum_{v=2}^n \psi(v) = n\psi(1), \quad \text{by (3.4).}$$

As the choice of coordinate was arbitrary, we have  $\mathbf{L}\psi = n\psi$ . So, every vector orthogonal to the all-1s vector is an eigenvector of eigenvalue  $n$ .  $\square$

*Alternative approach.* Observe that  $\mathbf{L}_{K_n} = n\mathbf{I} - \mathbf{1}\mathbf{1}^T$ .  $\square$

We often think of the Laplacian of the complete graph as being a scaling of the identity. For every  $\mathbf{x}$  orthogonal to the all-1s vector,  $\mathbf{L}\mathbf{x} = n\mathbf{x}$ .

Now, let's see how our bound on the isoperimetric number works out. Let  $S \subset [n]$ . Every vertex in  $S$  has  $n - |S|$  edges connecting it to vertices not in  $S$ . So,

$$\theta(S) = \frac{|S|(n - |S|)}{|S|} = n - |S| = \lambda_2(\mathbf{L}_{K_n})(1 - s),$$

where  $s = |S|/n$ . Thus, Theorem 3.1.1 is sharp for the complete graph.

### 3.4 The star graphs

The star graph on  $n$  vertices  $S_n$  has edge set  $\{(1, a) : 2 \leq a \leq n\}$ .

To determine the eigenvalues of  $S_n$ , we first observe that each vertex  $a \geq 2$  has degree 1, and that each of these degree-one vertices has the same neighbor. Whenever two degree-one vertices share the same neighbor, they provide an eigenvector of eigenvalue 1.

**Lemma 3.4.1.** *Let  $G = (V, E)$  be a graph, and let  $a$  and  $b$  be vertices of degree one that are both connected to another vertex  $c$ . Then, the vector  $\psi = \delta_a - \delta_b$  is an eigenvector of  $\mathbf{L}_G$  of eigenvalue 1.*

*Proof.* Just multiply  $\mathbf{L}_G$  by  $\psi$ , and check (using (3.3)) vertex-by-vertex that it equals  $\psi$ .  $\square$

As eigenvectors of different eigenvalues are orthogonal, this implies that  $\psi(a) = \psi(b)$  for every eigenvector with eigenvalue different from 1.

**Lemma 3.4.2.** *The graph  $S_n$  has eigenvalue 0 with multiplicity 1, eigenvalue 1 with multiplicity  $n - 2$ , and eigenvalue  $n$  with multiplicity 1.*

*Proof.* Applying Lemma 3.4.1 to vertices  $i$  and  $i+1$  for  $2 \leq i < n$ , we find  $n-2$  linearly independent eigenvectors of the form  $\delta_i - \delta_{i+1}$ , all with eigenvalue 1. As 0 is also an eigenvalue, only one eigenvalue remains to be determined.

Recall that the trace of a matrix equals both the sum of its diagonal entries and the sum of its eigenvalues. We know that the trace of  $\mathbf{L}_{S_n}$  is  $2n - 2$ , and we have identified  $n - 1$  eigenvalues that sum to  $n - 2$ . So, the remaining eigenvalue must be  $n$ .

To determine the corresponding eigenvector, recall that it must be orthogonal to the other eigenvectors we have identified. This tells us that it must have the same value at each of the points of the star. Let this value be 1, and let  $x$  be the value at vertex 1. As the eigenvector is orthogonal to the constant vectors, it must be that

$$(n - 1) + x = 0,$$

so  $x = -(n - 1)$ . □

### 3.5 Products of graphs

We now define a product on graphs. If we apply this product to two paths, we obtain a grid. If we apply it repeatedly to one edge, we obtain a hypercube.

**Definition 3.5.1.** *Let  $G = (V, E)$  and  $H = (W, F)$  be graphs. Then  $G \times H$  is the graph with vertex set  $V \times W$  and edge set*

$$\begin{aligned} & \left( (a, b), (\hat{a}, b) \right) \text{ where } (a, \hat{a}) \in E \text{ and} \\ & \left( (a, b), (a, \hat{b}) \right) \text{ where } (b, \hat{b}) \in F. \end{aligned}$$

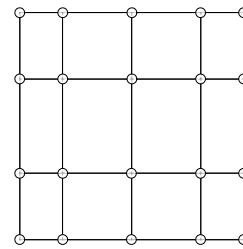


Figure 3.1: An  $m$ -by- $n$  grid graph is the product of a path on  $m$  vertices with a path on  $n$  vertices. This is a drawing of a 5-by-4 grid made using Hall's algorithm.

**Theorem 3.5.2.** *Let  $G = (V, E)$  and  $H = (W, F)$  be graphs with Laplacian eigenvalues  $\lambda_1, \dots, \lambda_n$  and  $\mu_1, \dots, \mu_m$ , and eigenvectors  $\alpha_1, \dots, \alpha_n$  and  $\beta_1, \dots, \beta_m$ , respectively. Then, for each  $1 \leq i \leq n$  and  $1 \leq j \leq m$ ,  $G \times H$  has an eigenvector  $\gamma_{i,j}$  of eigenvalue  $\lambda_i + \mu_j$  such that*

$$\gamma_{i,j}(a, b) = \alpha_i(a)\beta_j(b).$$

*Proof.* Let  $\boldsymbol{\alpha}$  be an eigenvector of  $L_G$  of eigenvalue  $\lambda$ , let  $\boldsymbol{\beta}$  be an eigenvector of  $L_H$  of eigenvalue  $\mu$ , and let  $\boldsymbol{\gamma}$  be defined as above.

To see that  $\boldsymbol{\gamma}$  is an eigenvector of eigenvalue  $\lambda + \mu$ , we compute

$$\begin{aligned}
(L\boldsymbol{\gamma})(a, b) &= \sum_{(a, \hat{a}) \in E} (\boldsymbol{\gamma}(a, b) - \boldsymbol{\gamma}(\hat{a}, b)) + \sum_{(b, \hat{b}) \in F} (\boldsymbol{\gamma}(a, b) - \boldsymbol{\gamma}(a, \hat{b})) \\
&= \sum_{(a, \hat{a}) \in E} (\boldsymbol{\alpha}(a)\boldsymbol{\beta}(b) - \boldsymbol{\alpha}(\hat{a})\boldsymbol{\beta}(b)) + \sum_{(b, \hat{b}) \in F} (\boldsymbol{\alpha}(a)\boldsymbol{\beta}(b) - \boldsymbol{\alpha}(a)\boldsymbol{\beta}(\hat{b})) \\
&= \sum_{(a, \hat{a}) \in E} \boldsymbol{\beta}(b)(\boldsymbol{\alpha}(a) - \boldsymbol{\alpha}(\hat{a})) + \sum_{(b, \hat{b}) \in F} \boldsymbol{\alpha}(a)(\boldsymbol{\beta}(b) - \boldsymbol{\beta}(\hat{b})) \\
&= \sum_{(a, \hat{a}) \in E} \boldsymbol{\beta}(b)\lambda\boldsymbol{\alpha}(a) + \sum_{(b, \hat{b}) \in F} \boldsymbol{\alpha}(a)\mu\boldsymbol{\beta}(b) \\
&= (\lambda + \mu)(\boldsymbol{\alpha}(a)\boldsymbol{\beta}(b)).
\end{aligned}$$

□

### 3.5.1 The Hypercube

The  $d$ -dimensional hypercube graph,  $H_d$ , is the graph with vertex set  $\{0, 1\}^d$ , with edges between vertices whose names differ in exactly one bit. The hypercube may also be expressed as the product of the one-edge graph with itself  $d - 1$  times, with the proper definition of graph product.

Let  $H_1$  be the graph with vertex set  $\{0, 1\}$  and one edge between those vertices. Its Laplacian matrix has eigenvalues 0 and 2. As  $H_d = H_{d-1} \times H_1$ , we may use this to calculate the eigenvalues and eigenvectors of  $H_d$  for every  $d$ .

Using Theorem 3.1.1 and the fact that  $\lambda_2(H_d) = 2$ , we can immediately prove the following isoperimetric theorem for the hypercube.

**Corollary 3.5.3.**

$$\theta_{H_d} \geq 1.$$

In particular, for every set of at most half the vertices of the hypercube, the number of edges on the boundary of that set is at least the number of vertices in that set.

This result is tight, as you can see by considering one face of the hypercube, such as all the vertices whose labels begin with 0. It is possible to prove this by more concrete combinatorial means. In fact, very precise analyses of the isoperimetry of sets of vertices in the hypercube can be obtained. See [Har76] or [Bol86].

## 3.6 Bounds on $\lambda_2$ by test vectors

We can reverse our thinking and use Theorem 3.1.1 to prove an upper bound on  $\lambda_2$ . If you recall the proof of that theorem, you will see a special case of proving an upper bound by a test vector.

By Theorem 2.1.3 we know that every vector  $\mathbf{v}$  orthogonal to  $\mathbf{1}$  provides an upper bound on  $\lambda_2$ :

$$\lambda_2 \leq \frac{\mathbf{v}^T \mathbf{L} \mathbf{v}}{\mathbf{v}^T \mathbf{v}}.$$

When we use a vector  $\mathbf{v}$  in this way, we call it a *test vector*.

Let's see what a test vector can tell us about  $\lambda_2$  of a path graph on  $n$  vertices. I would like to use the vector that assigns  $i$  to vertex  $a$  as a test vector, but it is not orthogonal to  $\mathbf{1}$ . So, we will use the next best thing. Let  $\mathbf{x}$  be the vector such that  $\mathbf{x}(a) = (n+1) - 2a$ , for  $1 \leq a \leq n$ . This vector satisfies  $\mathbf{x} \perp \mathbf{1}$ , so

$$\begin{aligned} \lambda_2(P_n) &\leq \frac{\sum_{1 \leq a < n} (x(a) - x(a+1))^2}{\sum_a x(a)^2} \\ &= \frac{\sum_{1 \leq a < n} 2^2}{\sum_a (n+1-2a)^2} \\ &= \frac{4(n-1)}{(n+1)n(n-1)/3} \quad (\text{clearly, the denominator is } n^3/c \text{ for some } c) \\ &= \frac{12}{n(n+1)}. \end{aligned} \tag{3.5}$$

We will soon see that this bound is of the right order of magnitude. Thus, Theorem 3.1.1 does not provide a good bound on the isoperimetric number of the path graph. The isoperimetric number is minimized by the set  $S = \{1, \dots, n/2\}$ , which has  $\theta(S) = 2/n$ . However, the upper bound provided by Theorem 3.1.1 is of the form  $c/n^2$ . Cheeger's inequality, which we will prove later in the semester, will tell us that the error of this approximation can not be worse than quadratic.

The Courant-Fischer theorem is not as helpful when we want to prove lower bounds on  $\lambda_2$ . To prove lower bounds, we need the form with a maximum on the outside, which gives

$$\lambda_2 \geq \max_{S: \dim(S)=n-1} \min_{\mathbf{v} \in S} \frac{\mathbf{v}^T \mathbf{L} \mathbf{v}}{\mathbf{v}^T \mathbf{v}}.$$

This is not too helpful, as it is difficult to prove lower bounds on

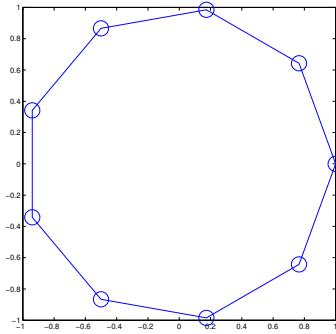
$$\min_{\mathbf{v} \in S} \frac{\mathbf{v}^T \mathbf{L} \mathbf{v}}{\mathbf{v}^T \mathbf{v}}$$

over a space  $S$  of large dimension. We will see a technique that lets us prove such lower bounds next lecture.

But, first we compute the eigenvalues and eigenvectors of the path graph exactly.

### 3.7 The Ring Graph

The ring graph on  $n$  vertices,  $R_n$ , may be viewed as having a vertex set corresponding to the integers modulo  $n$ . In this case, we view the vertices as the numbers 0 through  $n-1$ , with edges  $(a, a+1)$ , computed modulo  $n$ .



(a) The ring graph on 9 vertices.

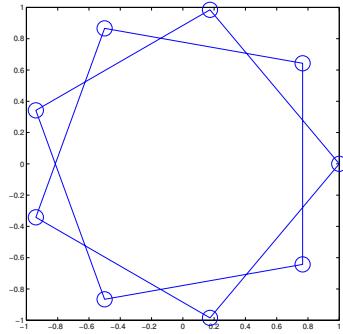
(b) The eigenvectors for  $k = 2$ .

Figure 3.2:

**Lemma 3.7.1.** *The Laplacian of  $R_n$  has eigenvectors*

$$\begin{aligned}\mathbf{x}_k(a) &= \cos(2\pi ka/n), \text{ and} \\ \mathbf{y}_k(a) &= \sin(2\pi ka/n),\end{aligned}$$

for  $0 \leq k \leq n/2$ , ignoring  $\mathbf{y}_0$  which is the all-zero vector, and for even  $n$  ignoring  $\mathbf{y}_{n/2}$  for the same reason. Eigenvectors  $\mathbf{x}_k$  and  $\mathbf{y}_k$  have eigenvalue  $2 - 2\cos(2\pi k/n)$ .

Note that  $\mathbf{x}_0$  is the all-ones vector. When  $n$  is even, we only have  $\mathbf{x}_{n/2}$ , which alternates  $\pm 1$ .

*Proof.* We will first see that  $\mathbf{x}_1$  and  $\mathbf{y}_1$  are eigenvectors by drawing the ring graph on the unit circle in the natural way: plot vertex  $u$  at point  $(\cos(2\pi a/n), \sin(2\pi a/n))$ .

You can see that the average of the neighbors of a vertex is a vector pointing in the same direction as the vector associated with that vertex. This should make it obvious that both the  $x$  and  $y$  coordinates in this figure are eigenvectors of the same eigenvalue. The same holds for all  $k$ .

Alternatively, we can verify that these are eigenvectors by a simple computation.

$$\begin{aligned}(L_{R_n} \mathbf{x}_k)(a) &= 2\mathbf{x}_k(a) - \mathbf{x}_k(a+1) - \mathbf{x}_k(a-1) \\ &= 2\cos(2\pi ka/n) - \cos(2\pi k(a+1)/n) - \cos(2\pi k(a-1)/n) \\ &= 2\cos(2\pi ka/n) - \cos(2\pi ka/n)\cos(2\pi k/n) + \sin(2\pi ka/n)\sin(2\pi k/n) \\ &\quad - \cos(2\pi ka/n)\cos(2\pi k/n) - \sin(2\pi ka/n)\sin(2\pi k/n) \\ &= 2\cos(2\pi ka/n) - \cos(2\pi ka/n)\cos(2\pi k/n) - \cos(2\pi ka/n)\cos(2\pi k/n) \\ &= (2 - 2\cos(2\pi k/n))\cos(2\pi ka/n) \\ &= (2 - \cos(2\pi k/n))\mathbf{x}_k(a).\end{aligned}$$

The computation for  $\mathbf{y}_k$  follows similarly. □

### 3.8 The Path Graph

We will derive the eigenvalues and eigenvectors of the path graph from those of the ring graph. To begin, I will number the vertices of the ring a little differently, as in Figure 3.3.

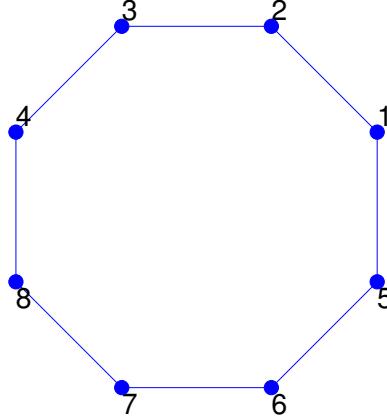


Figure 3.3: The ring on 8 vertices, numbered differently

**Lemma 3.8.1.** *Let  $P_n = (V, E)$  where  $V = \{1, \dots, n\}$  and  $E = \{(a, a+1) : 1 \leq a < n\}$ . The Laplacian of  $P_n$  has the same eigenvalues as  $R_{2n}$ , excluding 2. That is,  $P_n$  has eigenvalues namely  $2(1 - \cos(\pi k/n))$ , and eigenvectors*

$$v_k(a) = \cos(\pi k u / n - \pi k / 2n).$$

for  $0 \leq k < n$

*Proof.* We derive the eigenvectors and eigenvalues by treating  $P_n$  as a quotient of  $R_{2n}$ : we will identify vertex  $u$  of  $P_n$  with vertices  $u$  and  $u+n$  of  $R_{2n}$  (under the new numbering of  $R_{2n}$ ). These are pairs of vertices that are above each other in the figure that I drew.

Let  $\mathbf{I}_n$  be the  $n$ -dimensional identity matrix. You should check that

$$(\mathbf{I}_n \quad \mathbf{I}_n) \mathbf{L}_{R_{2n}} \begin{pmatrix} \mathbf{I}_n \\ \mathbf{I}_n \end{pmatrix} = 2\mathbf{L}_{P_n}.$$

If there is an eigenvector  $\psi$  of  $R_{2n}$  with eigenvalue  $\lambda$  for which  $\psi(a) = \psi(a+n)$  for  $1 \leq a \leq n$ , then the above equation gives us a way to turn this into an eigenvector of  $P_n$ : Let  $\phi \in \mathbb{R}^n$  be the vector for which

$$\phi(a) = \psi(a), \text{ for } 1 \leq a \leq n.$$

Then,

$$\begin{pmatrix} \mathbf{I}_n \\ \mathbf{I}_n \end{pmatrix} \phi = \psi, \quad \mathbf{L}_{R_{2n}} \begin{pmatrix} \mathbf{I}_n \\ \mathbf{I}_n \end{pmatrix} \phi = \lambda \psi, \quad \text{and} \quad (\mathbf{I}_n \quad \mathbf{I}_n) \mathbf{L}_{R_{2n}} \begin{pmatrix} \mathbf{I}_n \\ \mathbf{I}_n \end{pmatrix} \psi = 2\lambda \phi.$$

So, if we can find such a vector  $\psi$ , then the corresponding  $\phi$  is an eigenvector of  $P_n$  of eigenvalue  $\lambda$ .

As you've probably guessed, we can find such vectors  $\psi$ . I've drawn one in Figure 3.3. For each of the two-dimensional eigenspaces of  $R_{2n}$ , we get one such a vector. These provide eigenvectors of eigenvalue

$$2(1 - \cos(\pi k/n)),$$

for  $1 \leq k < n$ . Thus, we now know  $n - 1$  distinct eigenvalues. The last, of course, is zero.  $\square$

The type of quotient used in the above argument is known as an *equitable partition*. You can find a extensive exposition of these in Godsil's book [God93].

## References

- [Bol86] Béla Bollobás. *Combinatorics: set systems, hypergraphs, families of vectors, and combinatorial probability*. Cambridge University Press, 1986.
- [God93] Chris Godsil. *Algebraic Combinatorics*. Chapman & Hall, 1993.
- [Har76] Sergiu Hart. A note on the edges of the n-cube. *Discrete Mathematics*, 14(2):157–163, 1976.

**Spectral Graph Theory**

Lecture 3.5

**Dan's Favorite Inequality***Daniel A. Spielman*

September 8, 2018

In find the following inequality very useful. We will use it often during the semester, and it might help you on the problem set.

**Theorem 3.0.1.** *Let  $a_1, \dots, a_n$  and  $b_1, \dots, b_n$  be positive numbers. Then*

$$\min_i \frac{a_i}{b_i} \leq \frac{\sum_i a_i}{\sum_i b_i} \leq \max_i \frac{a_i}{b_i}.$$

*Proof.* We have

$$\sum_i a_i = \sum_i b_i \left( \frac{a_i}{b_i} \right) \leq \sum_i b_i \left( \max_j \frac{a_j}{b_j} \right) = \left( \max_j \frac{a_j}{b_j} \right) \sum_i b_i.$$

So,

$$\frac{\sum a_i}{\sum b_i} \leq \max_j \frac{a_j}{b_j}. \quad (3.1)$$

One can similarly prove

$$\frac{\sum a_i}{\sum b_i} \geq \min_j \frac{a_j}{b_j}. \quad (3.2)$$

□

## Bounding Eigenvalues

*Daniel A. Spielman*

September 10, 2018

## 4.1 Overview

It is unusual when one can actually explicitly determine the eigenvalues of a graph. Usually one is only able to prove loose bounds on some eigenvalues.

In this lecture we will see a powerful technique that allows one to compare one graph with another, and prove things like lower bounds on the smallest eigenvalue of a Laplacians. It often goes by the name “Poincaré Inequalities” (see [DS91, SJ89, GLM99]), although I often use the name “Graphic inequalities”, as I see them as providing inequalities between graphs.

## 4.2 Graphic Inequalities

I begin by recalling an extremely useful piece of notation that is used in the Optimization community. For a symmetric matrix  $\mathbf{A}$ , we write

$$\mathbf{A} \succcurlyeq 0$$

if  $\mathbf{A}$  is *positive semidefinite*. That is, if all of the eigenvalues of  $\mathbf{A}$  are nonnegative, which is equivalent to

$$\mathbf{v}^T \mathbf{A} \mathbf{v} \geq 0,$$

for all  $\mathbf{v}$ . We similarly write

$$\mathbf{A} \succ \mathbf{B}$$

if

$$\mathbf{A} - \mathbf{B} \succcurlyeq 0$$

which is equivalent to

$$\mathbf{v}^T \mathbf{A} \mathbf{v} \geq \mathbf{v}^T \mathbf{B} \mathbf{v}$$

for all  $\mathbf{v}$ .

The relation  $\preccurlyeq$  is called the Loewner *partial order*. It applies to some pairs of symmetric matrices, while others are incomparable. But, for all pairs to which it does apply, it acts like an order. For example, we have

$$\mathbf{A} \succcurlyeq \mathbf{B} \text{ and } \mathbf{B} \succcurlyeq \mathbf{C} \text{ implies } \mathbf{A} \succcurlyeq \mathbf{C},$$

and

$$\mathbf{A} \succcurlyeq \mathbf{B} \text{ implies } \mathbf{A} + \mathbf{C} \succcurlyeq \mathbf{B} + \mathbf{C},$$

for symmetric matrices  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$ .

I find it convenient to overload this notation by defining it for graphs as well. Thus, I'll write

$$G \succcurlyeq H$$

if  $\mathbf{L}_G \succcurlyeq \mathbf{L}_H$ . For example, if  $G = (V, E)$  is a graph and  $H = (V, F)$  is a subgraph of  $G$ , then

$$\mathbf{L}_G \succcurlyeq \mathbf{L}_H.$$

To see this, recall the Laplacian quadratic form:

$$\mathbf{x}^T \mathbf{L}_G \mathbf{x} = \sum_{(u,v) \in E} w_{u,v} (\mathbf{x}(u) - \mathbf{x}(v))^2.$$

It is clear that dropping edges can only decrease the value of the quadratic form. The same holds for decreasing the weights of edges.

This notation is most powerful when we consider some multiple of a graph. Thus, I could write

$$G \succcurlyeq c \cdot H,$$

for some  $c > 0$ . What is  $c \cdot H$ ? It is the same graph as  $H$ , but the weight of every edge is multiplied by  $c$ .

Using the Courant-Fischer Theorem, we can prove

**Lemma 4.2.1.** *If  $G$  and  $H$  are graphs such that*

$$G \succcurlyeq c \cdot H,$$

*then*

$$\lambda_k(G) \geq c \lambda_k(H),$$

*for all  $k$ .*

*Proof.* The Courant-Fischer Theorem tells us that

$$\lambda_k(G) = \min_{\substack{S \subseteq \mathbb{R}^n \\ \dim(S)=k}} \max_{\mathbf{x} \in S} \frac{\mathbf{x}^T \mathbf{L}_G \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \geq c \min_{\substack{S \subseteq \mathbb{R}^n \\ \dim(S)=k}} \max_{\mathbf{x} \in S} \frac{\mathbf{x}^T \mathbf{L}_H \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = c \lambda_k(H).$$

□

**Corollary 4.2.2.** *Let  $G$  be a graph and let  $H$  be obtained by either adding an edge to  $G$  or increasing the weight of an edge in  $G$ . Then, for all  $i$*

$$\lambda_i(G) \leq \lambda_i(H).$$

### 4.3 Approximations of Graphs

An idea that we will use in later lectures is that one graph approximates another if their Laplacian quadratic forms are similar. For example, we will say that  $H$  is a  $c$ -approximation of  $G$  if

$$cH \succsim G \succsim H/c.$$

Surprising approximations exist. For example, expander graphs are very sparse approximations of the complete graph. For example, the following is known.

**Theorem 4.3.1.** *For every  $\epsilon > 0$ , there exists a  $d > 0$  such that for all sufficiently large  $n$  there is a  $d$ -regular graph  $G_n$  that is a  $(1 + \epsilon)$ -approximation of  $K_n$ .*

These graphs have many fewer edges than the complete graphs!

In a later lecture we will also prove that every graph can be well-approximated by a sparse graph.

### 4.4 The Path Inequality

By now you should be wondering, “how do we prove that  $G \succsim c \cdot H$  for some graph  $G$  and  $H$ ?” Not too many ways are known. We’ll do it by proving some inequalities of this form for some of the simplest graphs, and then extending them to more general graphs. For example, we will prove

$$(n - 1) \cdot P_n \succsim G_{1,n}, \quad (4.1)$$

where  $P_n$  is the path from vertex 1 to vertex  $n$ , and  $G_{1,n}$  is the graph with just the edge  $(1, n)$ . All of these edges are unweighted.

The following very simple proof of this inequality was discovered by Sam Daitch.

**Lemma 4.4.1.**

$$(n - 1) \cdot P_n \succsim G_{1,n}.$$

*Proof.* We need to show that for every  $\mathbf{x} \in \mathbb{R}^n$ ,

$$(n - 1) \sum_{i=1}^{n-1} (\mathbf{x}(i+1) - \mathbf{x}(i))^2 \geq (\mathbf{x}(n) - \mathbf{x}(1))^2.$$

For  $1 \leq i \leq n - 1$ , set

$$\Delta(i) = \mathbf{x}(i+1) - \mathbf{x}(i).$$

The inequality we need to prove then becomes

$$(n - 1) \sum_{i=1}^{n-1} \Delta(i)^2 \geq \left( \sum_{i=1}^{n-1} \Delta(i) \right)^2.$$

But, this is just the Cauchy-Schwartz inequality. I'll remind you that Cauchy-Schwartz just follows from the fact that the inner product of two vectors is at most the product of their norms:

$$(n-1) \sum_{i=1}^{n-1} \Delta(i)^2 = \|\mathbf{1}_{n-1}\|^2 \|\Delta\|^2 = (\|\mathbf{1}_{n-1}\| \|\Delta\|)^2 \geq (\mathbf{1}_{n-1}^T \Delta)^2 = \left( \sum_{i=1}^{n-1} \Delta(i) \right)^2.$$

□

#### 4.4.1 Bounding $\lambda_2$ of a Path Graph

I'll now demonstrate the power of Lemma 4.4.1 by using it to prove a lower bound on  $\lambda_2(P_n)$  that will be very close to the upper bound we obtained from the test vector.

To prove a lower bound on  $\lambda_2(P_n)$ , we will prove that some multiple of the path is at least the complete graph. To this end, write

$$L_{K_n} = \sum_{i < j} L_{G_{i,j}},$$

and recall that

$$\lambda_2(K_n) = n.$$

For every edge  $(i, j)$  in the complete graph, we apply the only inequality available in the path:

$$G_{i,j} \preccurlyeq (j-i) \sum_{k=i}^{j-1} G_{k,k+1} \preccurlyeq (j-i)P_n. \quad (4.2)$$

This inequality says that  $G_{i,j}$  is at most  $(j-i)$  times the part of the path connecting  $i$  to  $j$ , and that this part of the path is less than the whole.

Summing inequality (4.2) over all edges  $(i, j) \in K_n$  gives

$$K_n = \sum_{i < j} G_{i,j} \preccurlyeq \sum_{i < j} (j-i)P_n.$$

To finish the proof, we compute

$$\sum_{1 \leq i < j \leq n} (j-i) = \sum_{k=1}^{n-1} k(n-k) = n(n+1)(n-1)/6.$$

So,

$$L_{K_n} \preccurlyeq \frac{n(n+1)(n-1)}{6} \cdot L_{P_n}.$$

Applying Lemma 4.2.1, we obtain

$$\frac{6}{(n+1)(n-1)} \leq \lambda_2(P_n).$$

This only differs from the upper bound we obtained last lecture using a test vector by a factor of 2.

## 4.5 The Complete Binary Tree

Let's do the same analysis with the complete binary tree.

One way of understanding the complete binary tree of depth  $d + 1$  is to identify the vertices of the tree with strings over  $\{0, 1\}$  of length at most  $d$ . The root of the tree is the empty string. Every other node has one ancestor, which is obtained by removing the last character of its string, and two children, which are obtained by appending one character to its label.

Alternatively, you can describe it as the graph on  $n = 2^{d+1} - 1$  nodes with edges of the form  $(i, 2i)$  and  $(i, 2i + 1)$  for  $i < n$ . We will name this graph  $T_d$ . Pictures of this graph appear below.

Pictorially, these graphs look like this:

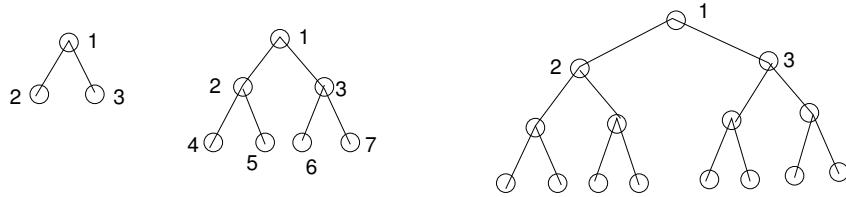


Figure 4.1:  $T_1$ ,  $T_2$  and  $T_3$ . Node 1 is at the top, 2 and 3 are its children. Some other nodes have been labeled as well.

Let's first upper bound  $\lambda_2(T_d)$  by constructing a test vector  $x$ . Set  $x(1) = 0$ ,  $x(2) = 1$ , and  $x(3) = -1$ . Then, for every vertex  $u$  that we can reach from node 2 without going through node 1, we set  $x(u) = 1$ . For all the other nodes, we set  $x(u) = -1$ .

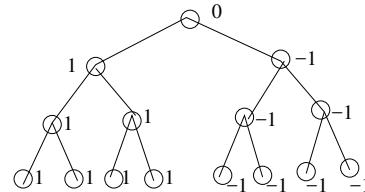


Figure 4.2: The test vector we use to upper bound  $\lambda_2(T_3)$ .

We then have

$$\lambda_2 \leq \frac{\sum_{(i,j) \in T_d} (x_i - x_j)^2}{\sum_i x_i^2} = \frac{(x_1 - x_2)^2 + (x_1 - x_3)^2}{n - 1} = 2/(n - 1).$$

We will again prove a lower bound by comparing  $T_d$  to the complete graph. For each edge  $(i, j) \in K_n$ , let  $T_d^{i,j}$  denote the unique path in  $T$  from  $i$  to  $j$ . This path will have length at most  $2d$ . So, we have

$$K_n = \sum_{i < j} G_{i,j} \asymp \sum_{i < j} (2d) T_d^{i,j} \asymp \sum_{i < j} (2 \log_2 n) T_d = \binom{n}{2} (2 \log_2 n) T_d.$$

So, we obtain the bound

$$\binom{n}{2}(2 \log_2 n) \lambda_2(T_d) \geq n,$$

which implies

$$\lambda_2(T_d) \geq \frac{1}{(n-1) \log_2 n}.$$

In the next problem set, I will ask you to improve this lower bound to  $1/cn$  for some constant  $c$ .

## 4.6 The weighted path

**Lemma 4.6.1.** *Let  $w_1, \dots, w_{n-1}$  be positive. Then*

$$G_{1,n} \preccurlyeq \left( \sum_{i=1}^{n-1} \frac{1}{w_i} \right) \sum_{i=1}^{n-1} w_i G_{i,i+1}.$$

*Proof.* Let  $\mathbf{x} \in \mathbb{R}^n$  and set  $\Delta(i)$  as in the proof of the previous lemma. Now, set

$$\gamma(i) = \Delta(i) \sqrt{w_i}.$$

Let  $\mathbf{w}^{-1/2}$  denote the vector for which

$$\mathbf{w}^{-1/2}(i) = \frac{1}{\sqrt{w_i}}.$$

Then,

$$\begin{aligned} \sum_i \Delta(i) &= \gamma^T \mathbf{w}^{-1/2}, \\ \|\mathbf{w}^{-1/2}\|^2 &= \sum_i \frac{1}{w_i}, \end{aligned}$$

and

$$\|\gamma\|^2 = \sum_i \Delta(i)^2 w_i.$$

So,

$$\begin{aligned} \mathbf{x}^T L_{G_{1,n}} \mathbf{x} &= \left( \sum_i \Delta(i) \right)^2 = \left( \gamma^T \mathbf{w}^{-1/2} \right)^2 \\ &\leq (\|\gamma\| \|\mathbf{w}^{-1/2}\|)^2 = \left( \sum_i \frac{1}{w_i} \right) \sum_i \Delta(i)^2 w_i = \left( \sum_i \frac{1}{w_i} \right) \mathbf{x}^T \left( \sum_{i=1}^{n-1} w_i L_{G_{i,i+1}} \right) \mathbf{x}. \end{aligned}$$

□

## 4.7 Exercises

1. Let  $\mathbf{v}$  be a vector so that  $\mathbf{v}^T \mathbf{1} = 0$ . Prove that

$$\|\mathbf{v}\|^2 \leq \|\mathbf{v} + t\mathbf{1}\|^2,$$

for every real number  $t$ .

## References

- [DS91] Persi Diaconis and Daniel Stroock. Geometric bounds for eigenvalues of Markov chains. *The Annals of Applied Probability*, 1(1):36–61, 1991.
- [GLM99] S. Guattery, T. Leighton, and G. L. Miller. The path resistance method for bounding the smallest nontrivial eigenvalue of a Laplacian. *Combinatorics, Probability and Computing*, 8:441–460, 1999.
- [SJ89] Alistair Sinclair and Mark Jerrum. Approximate counting, uniform generation and rapidly mixing Markov chains. *Information and Computation*, 82(1):93–133, July 1989.

## Cayley Graphs

Daniel A. Spielman

September 12, 2018

## 5.1 Cayley Graphs

Ring graphs and hypercubes are types of Cayley graph. In general, the vertices of a Cayley graph are the elements of some group  $\Gamma$ . In the case of the ring, the group is the set of integers modulo  $n$ . The edges of a Cayley graph are specified by a set  $S \subset \Gamma$ , which are called the *generators* of the Cayley graph. The set of generators must be closed under inverse. That is, if  $s \in S$ , then  $s^{-1} \in S$ . Vertices  $u, v \in \Gamma$  are connected by an edge if there is an  $s \in S$  such that

$$u \circ s = v,$$

where  $\circ$  is the group operation. In the case of Abelian groups, like the integers modulo  $n$ , this would usually be written  $u + s = v$ . The generators of the ring graph are  $\{1, -1\}$ .

The  $d$ -dimensional hypercube,  $H_d$ , is a Cayley graph over the additive group  $(\mathbf{Z}/2\mathbf{Z})^d$ : that is the set of vectors in  $\{0, 1\}^d$  under addition modulo 2. The generators are given by the vectors in  $\{0, 1\}^d$  that have a 1 in exactly one position. This set is closed under inverse, because every element of this group is its own inverse.

We require  $S$  to be closed under inverse so that the graph is undirected:

$$u + s = v \iff v + (-s) = u.$$

Cayley graphs over Abelian groups are particularly convenient because we can find an orthonormal basis of eigenvectors without knowing the set of generators. They just depend on the group<sup>1</sup>. Knowing the eigenvectors makes it much easier to compute the eigenvalues. We give the computations of the eigenvectors in sections ?? and A.

We will now examine two exciting types of Cayley graphs: Paley graphs and generalized hypercubes.

## 5.2 Paley Graphs

The Paley graph are Cayley graphs over the group of integer modulo a prime,  $p$ , where  $p$  is equivalent to 1 modulo 4. Such a group is often written  $\mathbb{Z}/p$ .

---

<sup>1</sup>More precisely, the characters always form an orthonormal set of eigenvectors, and the characters just depend upon the group. When two different characters have the same eigenvalue, we obtain an eigenspace of dimension greater than 1. These eigenspaces do depend upon the choice of generators.

I should begin by reminding you a little about the integers modulo  $p$ . The first thing to remember is that the integers modulo  $p$  are actually a field, written  $\mathbb{F}_p$ . That is, they are closed under both addition and multiplication (completely obvious), have identity elements under addition and multiplication (0 and 1), and have inverses under addition and multiplication. It is obvious that the integers have inverses under addition:  $-x$  modulo  $p$  plus  $x$  modulo  $p$  equals 0. It is a little less obvious that the integers modulo  $p$  have inverses under multiplication (except that 0 does not have a multiplicative inverse). That is, for every  $x \neq 0$ , there is a  $y$  such that  $xy = 1$  modulo  $p$ . When we write  $1/x$ , we mean this element  $y$ .

The generators of the Paley graphs are the squares modulo  $p$  (usually called the *quadratic residues*). That is, the set of numbers  $s$  such that there exists an  $x$  for which  $x^2 \equiv_p s$ . Thus, the vertex set is  $\{0, \dots, p-1\}$ , and there is an edge between vertices  $u$  and  $v$  if  $u - v$  is a square modulo  $p$ . I should now prove that  $-s$  is a quadratic residue if and only if  $s$  is. This will hold provided that  $p$  is equivalent to 1 modulo 4. To prove that, I need to tell you one more thing about the integers modulo  $p$ : their multiplicative group is cyclic.

**Fact 5.2.1.** *For every prime  $p$ , there exists a number  $g$  such that for every number  $x$  between 1 and  $p-1$ , there is a unique  $i$  between 1 and  $p-1$  such that*

$$x \equiv g^i \pmod{p}.$$

In particular,  $g^{p-1} \equiv 1$ .

**Corollary 5.2.2.** *If  $p$  is a prime equivalent to 1 modulo 4, then  $-1$  is a square modulo  $p$ .*

*Proof.* We know that 4 divides  $p-1$ . Let  $s = g^{(p-1)/4}$ . I claim that  $s^2 = -1$ . This will follow from  $s^4 = 1$ .

To see this, consider the equation

$$x^2 - 1 \equiv 0 \pmod{p}.$$

As the numbers modulo  $p$  are a field, it can have at most 2 solutions. Moreover, we already know two solutions,  $x = 1$  and  $x = -1$ . As  $s^4 = 1$ , we know that  $s^2$  must be one of 1 or  $-1$ . However, it cannot be the case that  $s^2 = 1$ , because then the powers of  $g$  would begin repeating after the  $(p-1)/2$  power, and thus could not represent every number modulo  $p$ .  $\square$

We now understand a lot about the squares modulo  $p$  (formally called *quadratic residues*). The squares are exactly the elements  $g^i$  where  $i$  is even. As  $g^i g^j = g^{i+j}$ , the fact that  $-1$  is a square implies that  $s$  is a square if and only if  $-s$  is a square. So,  $S$  is closed under negation, and the Cayley graph of  $\mathbb{Z}/p$  with generator set  $S$  is in fact a graph. As  $|S| = (p-1)/2$ , it is regular of degree

$$d = \frac{p-1}{2}.$$

### 5.3 Eigenvalues of the Paley Graphs

It will prove simpler to compute the eigenvalues of the adjacency matrix of the Paley Graphs. Since these graphs are regular, this will immediately tell us the eigenvalues of the Laplacian. Let  $\mathbf{L}$  be

the Laplacians matrix of the Paley graph on  $p$  vertices. A remarkable feature of Paley graph is that  $\mathbf{L}^2$  can be written as a linear combination of  $\mathbf{L}$ ,  $\mathbf{J}$  and  $\mathbf{I}$ , where  $\mathbf{J}$  is the all-1's matrix. We will prove that

$$\mathbf{L}^2 = p\mathbf{L} + \frac{p-1}{4}\mathbf{J} - \frac{p(p-1)}{4}\mathbf{I}. \quad (5.1)$$

The proof will be easiest if we express  $\mathbf{L}$  in terms of a matrix  $\mathbf{X}$  defined by the *quadratic character*:

$$\chi(x) = \begin{cases} 1 & \text{if } x \text{ is a quadratic residue modulo } p \\ 0 & \text{if } x = 0, \text{ and} \\ -1 & \text{otherwise.} \end{cases}$$

This is called a character because it satisfies  $\chi(xy) = \chi(x)\chi(y)$ . We will use this to define a matrix  $\mathbf{X}$  by

$$\mathbf{X}(u, v) = \chi(u - v).$$

An elementary calculation, which I skip, reveals that

$$\mathbf{X} = p\mathbf{I} - 2\mathbf{L} - \mathbf{J}. \quad (5.2)$$

**Lemma 5.3.1.**

$$\mathbf{X}^2 = p\mathbf{I} - \mathbf{J}.$$

When combined with (5.2), this lemma immediately implies (5.1).

*Proof.* The diagonal entries of  $\mathbf{X}^2$  are the squares of the norms of the columns of  $\mathbf{X}$ . As each contains  $(p-1)/2$  entries that are 1,  $(p-1)/2$  entries that are  $-1$ , and one entry that is 0, its squared norm is  $p-1$ .

To handle the off-diagonal entries, we observe that  $\mathbf{X}$  is symmetric, so the off-diagonal entries are the inner products of columns of  $\mathbf{X}$ . That is,

$$\mathbf{X}(u, v) = \sum_x \chi(u - x)\chi(v - x) = \sum_y \chi(y)\chi((v - u) + y),$$

where we have set  $y = u - x$ . For convenience, set  $w = v - u$ , so we can write this more simply. As we are considering a non-diagonal entry,  $w \neq 0$ . The term in the sum for  $y = 0$  is zero. When  $y \neq 0$ ,  $\chi(y) \in \pm 1$ , so

$$\chi(y)\chi(w + y) = \chi(w + y)/\chi(y) = \chi(w/y + 1).$$

Now, as  $y$  varies over  $\{1, \dots, p-1\}$ ,  $w/y$  varies over all of  $\{1, \dots, p-1\}$ . So,  $w/y + 1$  varies over all elements other than 1. This means that

$$\sum_y \chi(y)\chi((v - u) + y) = \left( \sum_{z=0}^{p-1} \chi(z) \right) - \chi(1) = 0 - 1 = -1.$$

So, every off-diagonal entry in  $\mathbf{X}^2$  is  $-1$ . □

This gives us a quadratic equation that every eigenvalue other than  $d$  must obey. Let  $\phi$  be an eigenvector of  $\mathbf{L}$  of eigenvalue  $\lambda \neq 0$ . As  $\phi$  is orthogonal to the all-1s vector,  $\mathbf{J}\phi = \mathbf{0}$ . So,

$$\lambda^2\phi = \mathbf{L}^2\phi = p\mathbf{L}\phi - \frac{p(p-1)}{4}\mathbf{I}\phi == (p\lambda - p(p-1)/4)\phi.$$

So, we find

$$\lambda^2 + p\lambda - \frac{p(p-1)}{4} = 0.$$

This gives

$$\lambda = \frac{1}{2}(p \pm \sqrt{p}).$$

This tells us at least two interesting things:

1. The Paley graph is (up to a very small order term) a  $1 + \sqrt{1/p}$  approximation of the complete graph.
2. Payley graphs have only two nonzero eigenvalues. This places them within the special family of Strongly Regular Graphs, that we will study later in the semester.

## 5.4 Generalizing Hypercubes

To generalize the hypercube, we will consider this same group, but with a general set of generators. We will call them  $\mathbf{g}_1, \dots, \mathbf{g}_k$ , and remember that each is a vector in  $\{0, 1\}^d$ , modulo 2.

Let  $G$  be the Cayley graph with these generators. To be concrete, I set  $V = \{0, 1\}^d$ , and note that  $G$  has edge set

$$\{(\mathbf{x}, \mathbf{x} + \mathbf{g}_j) : \mathbf{x} \in V, 1 \leq j \leq k\}.$$

Using the analysis of products of graphs, we can derive a set of eigenvectors of  $H_d$ . We will now verify that these are eigenvectors for all generalized hypercubes. Knowing these will make it easy to describe the eigenvalues.

For each  $\mathbf{b} \in \{0, 1\}^d$ , define the function  $\psi_{\mathbf{b}}$  from  $V$  to the reals given by

$$\psi_{\mathbf{b}}(\mathbf{x}) = (-1)^{\mathbf{b}^T \mathbf{x}}.$$

When I write  $\mathbf{b}^T \mathbf{x}$ , you might wonder if I mean to take the sum over the reals or modulo 2. As both  $\mathbf{b}$  and  $\mathbf{x}$  are  $\{0, 1\}$ -vectors, you get the same answer either way you do it.

While it is natural to think of  $\mathbf{b}$  as being a vertex, that is the wrong perspective. Instead, you should think of  $\mathbf{b}$  as indexing a Fourier coefficient (if you don't know what a Fourier coefficient is, just don't think of it as a vertex).

The eigenvectors and eigenvalues of the graph are determined by the following theorem. As this graph is  $k$ -regular, the eigenvectors of the adjacency and Laplacian matrices will be the same.

**Lemma 5.4.1.** For each  $\mathbf{b} \in \{0, 1\}^d$  the vector  $\psi_b$  is a Laplacian matrix eigenvector with eigenvalue

$$k - \sum_{i=1}^k (-1)^{\mathbf{b}^T \mathbf{g}_i}.$$

*Proof.* We begin by observing that

$$\psi_b(\mathbf{x} + \mathbf{y}) = (-1)^{\mathbf{b}^T (\mathbf{x} + \mathbf{y})} = (-1)^{\mathbf{b}^T \mathbf{x}} (-1)^{\mathbf{b}^T \mathbf{y}} = \psi_b(\mathbf{x}) \psi_b(\mathbf{y}).$$

Let  $\mathbf{L}$  be the Laplacian matrix of the graph. For any vector  $\psi_b$  for  $\mathbf{b} \in \{0, 1\}^d$  and any vertex  $\mathbf{x} \in V$ , we compute

$$\begin{aligned} (\mathbf{L}\psi_b)(\mathbf{x}) &= k\psi_b(\mathbf{x}) - \sum_{i=1}^k \psi_b(\mathbf{x} + \mathbf{g}_i) \\ &= k\psi_b(\mathbf{x}) - \sum_{i=1}^k \psi_b(\mathbf{x}) \psi_b(\mathbf{g}_i) \\ &= \psi_b(\mathbf{x}) \left( k - \sum_{i=1}^k \psi_b(\mathbf{g}_i) \right). \end{aligned}$$

So,  $\psi_b$  is an eigenvector of eigenvalue

$$k - \sum_{i=1}^k \psi_b(\mathbf{g}_i) = k - \sum_{i=1}^k (-1)^{\mathbf{b}^T \mathbf{g}_i}.$$

□

## 5.5 A random set of generators

We will now show that if we choose the set of generators uniformly at random, for  $k$  some constant multiple of the dimension, then we obtain a graph that is a good approximation of the complete graph. That is, all the eigenvalues of the Laplacian will be close to  $k$ . I will set  $k = cd$ , for some  $c > 1$ . Think of  $c = 2$ ,  $c = 10$ , or  $c = 1 + \epsilon$ .

For  $\mathbf{b} \in \{0, 1\}^d$  but not all zero, and for  $\mathbf{g}$  chosen uniformly at random from  $\{0, 1\}^d$ ,  $\mathbf{b}^T \mathbf{g}$  modulo 2 is uniformly distributed in  $\{0, 1\}$ , and so

$$(-1)^{\mathbf{b}^T \mathbf{g}}$$

is uniformly distributed in  $\pm 1$ . So, if we pick  $\mathbf{g}_1, \dots, \mathbf{g}_k$  independently and uniformly from  $\{0, 1\}^d$ , the eigenvalue corresponding to the eigenvector  $\psi_b$  is

$$\lambda_b \stackrel{\text{def}}{=} k - \sum_{i=1}^k (-1)^{\mathbf{b}^T \mathbf{g}_i}.$$

The right-hand part is a sum of independent, uniformly chosen  $\pm 1$  random variables. So, we know it is concentrated around 0, and thus  $\lambda_b$  will be concentrated around  $k$ . To determine how concentrated the sum actually is, we use a Chernoff bound. There are many forms of Chernoff bounds. I will not use the strongest, but settle for one which is simple and which gives results that are qualitatively correct.

**Theorem 5.5.1.** *Let  $x_1, \dots, x_k$  be independent  $\pm 1$  random variables. Then, for all  $t > 0$ ,*

$$\Pr \left[ \left| \sum_i x_i \right| \geq t \right] \leq 2e^{-t^2/2k}.$$

This becomes very small when  $t$  is a constant fraction of  $k$ . In fact, it becomes so small that it is unlikely that any eigenvalue deviates from  $k$  by more than  $t$ .

**Theorem 5.5.2.** *With high probability, all of the nonzero eigenvalues of the generalized hypercube differ from  $k$  by at most*

$$k\sqrt{\frac{2}{c}},$$

where  $k = cd$ .

*Proof.* Let  $t = k\sqrt{2/c}$ . Then, for every nonzero  $b$ ,

$$\Pr [|k - \lambda_b| \geq t] \leq 2e^{-t^2/2k} \leq 2e^{-k/c} = 2e^{-d}.$$

Now, the probability that there is some  $b$  for which  $\lambda_b$  violates these bounds is at most the sum of these terms:

$$\Pr [\exists b : |k - \lambda_b| \geq t] \leq \sum_{b \in \{0,1\}^d, b \neq 0^d} \Pr [|k - \lambda_b| \geq t] \leq (2^d - 1)2e^{-d},$$

which is always less than 1 and goes to zero exponentially quickly as  $d$  grows.  $\square$

I initially suggested thinking of  $c = 2$  or  $c = 10$ . The above bound works for  $c = 10$ . To get a useful bound for  $c = 2$ , we need to sharpen the analysis. A naive sharpening will work down to  $c = 2 \ln 2$ . To go lower than that, you need a stronger Chernoff bound.

## 5.6 Conclusion

We have now seen that a random generalized hypercube of degree  $k$  probably has all non-zero Laplacian eigenvalues between

$$k(1 - \sqrt{2/c}) \quad \text{and} \quad k(1 + \sqrt{2/c}).$$

If we let  $n$  be the number of vertices, and we now multiply the weight of every edge by  $n/k$ , we obtain a graph with all nonzero Laplacian eigenvalues between

$$n(1 - \sqrt{2/c}) \quad \text{and} \quad n(1 + \sqrt{2/c}).$$

Thus, this is essentially a  $1 + \sqrt{2/c}$  approximation of the complete graph on  $n$  vertices. But, the degree of every vertex is only  $c \log_2 n$ . Expanders are infinite families of graphs that are constant-factor approximations of complete graphs, but with constant degrees.

We know that random regular graphs are probably expanders. If we want explicit constructions, we need to go to non-Abelian groups.

Explicit constructions that achieve bounds approaching those of random generalized hypercubes come from error-correcting codes.

Explicit constructions allow us to use these graphs in applications that require us to implicitly deal with a very large graph. A few weeks from now, we will see how to use such graphs to construct pseudo-random generators.

## 5.7 Non-Abelian Groups

In the homework, you will show that it is impossible to make constant-degree expander graphs from Cayley graphs of Abelian groups. The best expanders are constructed from Cayley graphs of 2-by-2 matrix groups. In particular, the Ramanujan expanders of Margulis [Mar88] and Lubotzky, Phillips and Sarnak [LPS88] are Cayley graphs over the Projective Special Linear Groups  $\mathrm{PSL}(2, p)$ , where  $p$  is a prime. These are the 2-by-2 matrices modulo  $p$  with determinant 1, in which we identify  $A$  with  $-A$ .

They provided a very concrete set of generators. For a prime  $q$  modulo to 1 modulo 4, it is known that there are  $p+1$  solutions to the equation

$$a_1^2 + a_2^2 + a_3^2 + a_4^2 = p,$$

where  $a_1$  is odd and  $a_2, a_3$  and  $a_4$  are even. We obtain a generator for each such solution of the form:

$$\frac{1}{\sqrt{p}} \begin{bmatrix} a_0 + ia_1 & a_2 + ia_3 \\ -a_2 + ia_3 & a_0 - ia_1 \end{bmatrix},$$

where  $i$  is an integer that satisfies  $i^2 = -1$  modulo  $p$ .

Even more explicit constructions, which do not require solving equations, may be found in [ABN<sup>+</sup>92].

## A Eigenvectors of Cayley Graphs of Abelian Groups

The wonderful thing about Cayley graphs of Abelian groups is that we can construct an orthonormal basis of eigenvectors for these graphs without even knowing the set of generators  $S$ . That is, the eigenvectors only depend upon the group. Related results also hold for Cayley graphs of arbitrary groups, and are related to representations of the groups. See [Bab79] for details.

As Cayley graphs are regular, it won't matter which matrix we consider. For simplicity, we will consider adjacency matrices.

Let  $n$  be an integer and let  $G$  be a Cayley graph on  $\mathbf{Z}/n$  with generator set  $S$ . When  $S = \{\pm 1\}$ , we get the ring graphs. For general  $S$ , I think of these as generalized Ring graphs. Let's first see that they have the same eigenvectors as the Ring graphs.

Recall that we proved that the vectors  $\mathbf{x}_k$  and  $\mathbf{y}_k$  were eigenvectors of the ring graphs, where

$$\begin{aligned}\mathbf{x}_k(u) &= \sin(2\pi ku/n), \text{ and} \\ \mathbf{y}_k(u) &= \cos(2\pi ku/n),\end{aligned}$$

for  $1 \leq k \leq n/2$ .

Let's just do the computation for the  $\mathbf{x}_k$ , as the  $\mathbf{y}_k$  are similar. For every  $u$  modulo  $n$ , we have

$$\begin{aligned}(A\mathbf{x}_k)(u) &= \sum_{g \in S} \mathbf{x}_k(u + g) \\ &= \frac{1}{2} \left( \sum_{g \in S} \mathbf{x}_k(u + g) + \mathbf{x}_k(u - g) \right) \\ &= \frac{1}{2} \left( \sum_{g \in S} \sin(2\pi k(u + g)/n) + \sin(2\pi k(u - g)/n) \right) \\ &= \frac{1}{2} \left( \sum_{g \in S} 2 \sin(2\pi ku/n) \cos(2\pi kg/n) \right) \\ &= \sin(2\pi ku/n) \sum_{g \in S} \cos(2\pi kg/n) \\ &= \mathbf{x}_k(u) \sum_{g \in S} \cos(2\pi kg/n).\end{aligned}$$

So, the corresponding eigenvalue is

$$\sum_{g \in S} \cos(2\pi kg/n).$$

## References

- [ABN<sup>+</sup>92] Noga Alon, Jehoshua Bruck, Joseph Naor, Moni Naor, and Ron M. Roth. Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs. *IEEE Transactions on Information Theory*, 38(2):509–516, March 1992.
- [Bab79] László Babai. Spectra of cayley graphs. *Journal of Combinatorial Theory, Series B*, pages 180–189, 1979.
- [LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

- [Mar88] G. A. Margulis. Explicit group theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators. *Problems of Information Transmission*, 24(1):39–46, July 1988.

## High-Frequency Eigenvalues

*Daniel A. Spielman*

September 17, 2018

## 6.1 Overview

In this lecture we will see how high-frequency eigenvalues of the Laplacian matrix can be related to independent sets and graph coloring. Some of the bounds we obtained will be more easily stated in terms of the adjacency matrix,  $\mathbf{M}$ . Recall we number the Laplacian matrix eigenvalues in increasing order:

$$0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n.$$

We call the adjacency matrix eigenvalues  $\mu_1, \dots, \mu_n$ , and number them in the reverse order:

$$\mu_1 \geq \cdots \geq \mu_n.$$

The reason is that for  $d$ -regular graphs,  $\mu_i = d - \lambda_i$ .

## 6.2 Graph Coloring and Independent Sets

A coloring of a graph is an assignment of one color to every vertex in a graph so that each edge connects vertices of different colors. We are interested in coloring graphs while using as few colors as possible. Formally, a  $k$ -coloring of a graph is a function  $c : V \rightarrow \{1, \dots, k\}$  so that for all  $(u, v) \in E$ ,  $c(u) \neq c(v)$ . A graph is  $k$ -colorable if it has a  $k$ -coloring. The chromatic number of a graph, written  $\chi_G$ , is the least  $k$  for which  $G$  is  $k$ -colorable. A graph  $G$  is 2-colorable if and only if it is bipartite. Determining whether or not a graph is 3-colorable is an NP-complete problem. The famous 4-Color Theorem [AH77a, AH77b] says that every planar graph is 4-colorable.

A set of vertices  $S$  is *independent* if there are no edges between vertices in  $S$ . In particular, each color class in a coloring is an independent set. The problem of finding large independent sets in a graph is NP-Complete, and it is very difficult to even approximate the size of the largest independent set in a graph.

However, for some carefully chosen graphs one can obtain very good bounds on the sizes of independent sets by using spectral graph theory. We may later see some uses of this theory in the analysis of error-correcting codes and sphere packings.

### 6.3 Hoffman's Bound

One of the first results in spectral graph theory was Hoffman's proof the following upper bound on the size of an independent set in a graph  $G$ .

**Theorem 6.3.1.** *Let  $G = (V, E)$  be a  $d$ -regular graph, and let  $\mu_n$  be its smallest adjacency matrix eigenvalue. Then*

$$\alpha(G) \leq n \frac{-\mu_n}{d - \mu_n}.$$

Recall that  $\mu_n < 0$ . Otherwise this theorem would not make sense. We will prove a generalization of Hoffman's theorem due to Godsil and Newman [GN08]:

**Theorem 6.3.2.** *Let  $S$  be an independent set in  $G$ , and let  $d_{ave}(S)$  be the average degree of a vertex in  $S$ . Then,*

$$|S| \leq n \left( 1 - \frac{d_{ave}(S)}{\lambda_n} \right).$$

This is a generalization because in the  $d$ -regular case  $d_{ave} = d$  and  $\lambda_n = d - \mu_n$ . So, these bounds are the same for regular graphs:

$$1 - \frac{d_{ave}(S)}{\lambda_n} = \frac{\lambda_n - d}{\lambda_n} = \frac{-\mu_n}{d - \mu_n}.$$

*Proof.* Let  $S$  be an independent set of vertices and let  $d(S)$  be the sum of the degrees of vertices in  $S$ .

Recall that

$$\lambda_n = \max_{\mathbf{x}} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

We also know that the vector  $\mathbf{x}$  that maximizes this quantity is  $\psi_n$ , and that  $\psi_n$  is orthogonal to  $\psi_1$ . So, we can refine this expression to

$$\lambda_n = \max_{\mathbf{x} \perp \mathbf{1}} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

Consider the vector

$$\mathbf{x} = \chi_S - s\mathbf{1},$$

where  $s = |S|/n$ . As  $S$  is independent, we have

$$\mathbf{x}^T \mathbf{L} \mathbf{x} = |\partial(S)| = d(S) = d_{ave}(S) |S|.$$

Claim 2.3.2 tells us that the square of the norm of  $\mathbf{x}$  is

$$\mathbf{x}^T \mathbf{x} = n(s - s^2).$$

So,

$$\lambda_n \geq \frac{d_{ave}(S) |S|}{n(s - s^2)} = \frac{d_{ave}(S)sn}{n(s - s^2)} = \frac{d_{ave}(S)}{1 - s}.$$

Re-arranging terms, this gives

$$1 - \frac{d_{ave}(S)}{\lambda_n} \geq s,$$

which is equivalent to the claim of the theorem.  $\square$

## 6.4 Application to Paley graphs

Let's examine what Hoffman's bound on the size of the largest independent set tells us about Paley graphs.

If  $G$  is a Paley graph and  $S$  is an independent set, we have  $n = p$ ,  $d = (p-1)/2$ , and  $\lambda_n = (p+\sqrt{p})/2$ , so Hoffman's bound tells us that

$$\begin{aligned} |S| &\leq n \left( 1 - \frac{d_{ave}(S)}{\lambda_n} \right) \\ &= p \left( 1 - \frac{p-1}{p+\sqrt{p}} \right) \\ &= p \left( \frac{\sqrt{p}+1}{p+\sqrt{p}} \right) \\ &= \sqrt{p}. \end{aligned}$$

One can also show that every clique in a Paley graph has size at most  $\sqrt{p}$ .

A graph is called a  $k$ -Ramsey graph if it contains no clique or independent set of size  $k$ . It is a challenge to find large  $k$ -Ramsey graphs. Equivalently, it is challenging to find  $k$ -Ramsey graphs on  $n$  vertices for which  $k$  is small. In one of the first papers on the Probabilistic Method in Combinatorics, Erdős proved that a random graph on  $n$  vertices in which each edge is included with probability  $1/2$  is probably  $2 \log_2 n$  Ramsey [Erd47].

However, constructing explicit Ramsey graphs has proved much more challenging. Until recently, Paley graphs were among the best known. A recent construction of Barak, Rao, Shaltiel and Wigderson [BRSW12] constructs explicit graphs that are  $2^{(\log n)^{o(1)}}$  Ramsey.

## 6.5 Lower Bound on the chromatic number

As a  $k$ -colorable graph must have an independent set of size at least  $n/k$ , an upper bound on the sizes of independent sets gives a lower bound on its chromatic number. However, this bound is not always a good one.

For example, consider a graph on  $2n$  vertices consisting of a clique on  $n$  vertices and  $n$  vertices of degree 1, each of which is connected to a different vertex in the clique. The chromatic number of

this graph is  $n$ , because each of the vertices in the clique must have a different color. However, the graph also has an independent set of size  $n$ , which would only give a lower bound of 2 on the chromatic number.

Hoffman proved the following lower bound on the chromatic number of a graph that does not require the graph to be regular. Numerically, it is obtained by dividing  $n$  by the bound in Theorem 6.3.1. But, the proof is very different because that theorem only applies to regular graphs.

**Theorem 6.5.1.**

$$\chi(G) \geq \frac{\mu_1 - \mu_n}{-\mu_n} = 1 + \frac{\mu_1}{-\mu_n}.$$

The proof of this theorem relies on one inequality that I will not have time to cover in class. So, I will put its proof in Section B.

**Lemma 6.5.2.** *Let*

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \cdots & \mathbf{A}_{1,k} \\ \mathbf{A}_{1,2}^T & \mathbf{A}_{2,2} & \cdots & \mathbf{A}_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{1,k}^T & \mathbf{A}_{2,k}^T & \cdots & \mathbf{A}_{k,k} \end{bmatrix}$$

be a block-partitioned symmetric matrix with  $k \geq 2$ . Then

$$(k-1)\lambda_{\min}(\mathbf{A}) + \lambda_{\max}(\mathbf{A}) \leq \sum_i \lambda_{\max}(\mathbf{A}_{i,i}).$$

*Proof of Theorem 6.5.1.* Let  $G$  be a  $k$ -colorable graph. After possibly re-ordering the vertices, the adjacency matrix of  $G$  can be written

$$\begin{bmatrix} \mathbf{0} & \mathbf{A}_{1,2} & \cdots & \mathbf{A}_{1,k} \\ \mathbf{A}_{1,2}^T & \mathbf{0} & \cdots & \mathbf{A}_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{1,k}^T & \mathbf{A}_{2,k}^T & \cdots & \mathbf{0} \end{bmatrix}.$$

Each block corresponds to a color.

As each diagonal block is all-zero, Lemma 6.5.2 implies

$$(k-1)\lambda_{\min}(\mathbf{A}) + \lambda_{\max}(\mathbf{A}) \leq 0.$$

Recalling that  $\lambda_{\min}(\mathbf{A}) = \mu_n < 0$ , and  $\lambda_{\max}(\mathbf{A}) = \mu_1$ , a little algebra yields

$$1 + \frac{\mu_1}{-\mu_n} \leq k.$$

□

To return to our example of the  $n$  clique with  $n$  degree-1 vertices attached, I examined an example with  $n = 6$ . We find  $\mu_1 = 5.19$  and  $\mu_{12} = -1.62$ . This gives a lower bound on the chromatic number of 4.2, which implies a lower bound of 5. We can improve the lower bound by re-weighting the edges of the graph. For example, if we give weight 2 to all the edges in the clique and weight 1 to all the others, we obtain a bound of 5.18, which agrees with the chromatic number of this graph which is 6.

## 6.6 Coloring and The Adjacency Matrix

I would also like to show how to use spectral graph theory to color a graph. I do not know how to do this using the Laplacian matrix, so we will work with the Adjacency matrix. This will provide me with a good opportunity to cover some material about adjacency matrices that I have neglected.

## 6.7 The Largest Eigenvalue, $\mu_1$

We now examine  $\mu_1$  for graphs which are not necessarily regular. Let  $G$  be a graph, let  $d_{max}$  be the maximum degree of a vertex in  $G$ , and let  $d_{ave}$  be the average degree of a vertex in  $G$ .

**Lemma 6.7.1.**

$$d_{ave} \leq \mu_1 \leq d_{max}.$$

*Proof.* The lower bound follows by considering the Rayleigh quotient with the all-1s vector:

$$\mu_1 = \max_{\mathbf{x}} \frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \geq \frac{\mathbf{1}^T \mathbf{M} \mathbf{1}}{\mathbf{1}^T \mathbf{1}} = \frac{\sum_{i,j} \mathbf{M}(i,j)}{n} = \frac{\sum_i \mathbf{d}(i)}{n}.$$

To prove the upper bound, Let  $\psi_1$  be an eigenvector of eigenvalue  $\mu_1$ . Let  $v$  be the vertex on which it takes its maximum value, so  $\psi_1(v) \geq \psi_1(u)$  for all  $u$ , and assume without loss of generality that  $\psi_1(v) \neq 0$ . We have

$$\mu_1 = \frac{(\mathbf{M}\psi_1)(v)}{\psi_1(v)} = \frac{\sum_{u \sim v} \psi_1(u)}{\psi_1(v)} = \sum_{u \sim v} \frac{\psi_1(u)}{\psi_1(v)} \leq \sum_{u \sim v} 1 \leq d(v) \leq d_{max}. \quad (6.1)$$

□

**Lemma 6.7.2.** *If  $G$  is connected and  $\mu_1 = d_{max}$ , then  $G$  is  $d_{max}$ -regular.*

*Proof.* If we have equality in (6.1), then it must be the case that  $d(v) = d_{max}$  and  $\phi_1(u) = \phi_1(v)$  for all  $(u, v) \in E$ . Thus, we may apply the same argument to every neighbor of  $v$ . As the graph is connected, we may keep applying this argument to neighbors of vertices to which it has already been applied to show that  $\phi_1(z) = \phi_1(v)$  and  $d(z) = d_{max}$  for all  $z \in V$ . □

## 6.8 Wilf's Theorem

While we may think of  $\mu_1$  as being related to the average degree, it does behave differently. In particular, if we remove the vertex of smallest degree from a graph, the average degree can increase. On the other hand,  $\mu_1$  can only decrease when we remove a vertex. Let's prove that now.

**Lemma 6.8.1.** *Let  $A$  be a symmetric matrix with largest eigenvalue  $\alpha_1$ . Let  $B$  be the matrix obtained by removing the last row and column from  $A$ , and let  $\beta_1$  be the largest eigenvalue of  $B$ . Then,*

$$\alpha_1 \geq \beta_1.$$

*Proof.* For any vector  $\mathbf{y} \in \mathbb{R}^{n-1}$ , we have

$$\mathbf{y}^T B \mathbf{y} = \begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}^T A \begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}.$$

So, for  $\mathbf{y}$  an eigenvector of  $B$  of eigenvalue  $\beta_1$ ,

$$\beta_1 = \frac{\mathbf{y}^T B \mathbf{y}}{\mathbf{y}^T \mathbf{y}} = \frac{\begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}^T A \begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}}{\begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}^T \begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}} \leq \max_{\mathbf{x} \in \mathbb{R}^n} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

□

Of course, this holds regardless of which row and column we remove, as long as they are the same row and column.

It is easy to show that every graph is  $(d_{\max} + 1)$ -colorable. Assign colors to the vertices one-by-one. As each vertex has at most  $d_{\max}$  neighbors, there is always some color one can assign that vertex that is different than those assigned to its neighbors. The following theorem of Wilf improves upon this bound.

### Theorem 6.8.2.

$$\chi(G) \leq \lfloor \mu_1 \rfloor + 1.$$

*Proof.* We prove this by induction on the number of vertices in the graph. To ground the induction, consider the graph with one vertex and no edges. It has chromatic number 1 and largest eigenvalue zero<sup>1</sup>. Now, assume the theorem is true for all graphs on  $n - 1$  vertices, and let  $G$  be a graph on  $n$  vertices. By Lemma 6.7.1,  $G$  has a vertex of degree at most  $\lfloor \mu_1 \rfloor$ . Let  $v$  be such a vertex and let  $G - \{v\}$  be the graph obtained by removing this vertex. By Lemma 6.8.1 and our induction hypothesis,  $G - \{v\}$  has a coloring with at most  $\lfloor \mu_1 \rfloor + 1$  colors. Let  $c$  be any such coloring. We just need to show that we can extend  $c$  to  $v$ . As  $v$  has at most  $\lfloor \mu_1 \rfloor$  neighbors, there is some color in  $\{1, \dots, \lfloor \mu_1 \rfloor + 1\}$  that does not appear among its neighbors, and which it may be assigned. Thus,  $G$  has a coloring with  $\lfloor \mu_1 \rfloor + 1$  colors. □

For an example, consider a path graph with at least 3 vertices. We have  $d_{\max} = 2$ , but  $\alpha_1 < 2$ .

## 6.9 Perron-Frobenius Theory

The eigenvector corresponding to the largest eigenvalue of the adjacency matrix of a graph is usually not a constant vector. However, it is always a positive vector if the graph is connected.

This follows from the Perron-Frobenius theory. In fact, the Perron-Frobenius theory says much more, and it can be applied to adjacency matrices of strongly connected directed graphs. Note that

---

<sup>1</sup>If this makes you uncomfortable, you could use both graphs on two vertices

these need not even be diagonalizable! We will defer a discussion of the general theory until we discuss directed graphs, which will happen towards the end of the semester. If you want to see it now, look at the third lecture from my notes from 2009.

In the symmetric case, the theory is made much easier by both the spectral theory and the characterization of eigenvalues as extreme values of Rayleigh quotients.

**Theorem 6.9.1.** [Perron-Frobenius, Symmetric Case] Let  $G$  be a connected weighted graph, let  $\mathbf{M}$  be its adjacency matrix, and let  $\mu_1 \geq \mu_2 \geq \dots \geq \mu_n$  be its eigenvalues. Then

- a.  $\mu_1 \geq -\mu_n$ , and
- b.  $\mu_1 > \mu_2$ ,
- c. The eigenvalue  $\mu_1$  has a strictly positive eigenvector.

Before proving Theorem 6.9.1, we will prove a lemma that will be useful in the proof and a few other places today. It says that non-negative eigenvectors of non-negative adjacency matrices of connected graphs must be strictly positive.

**Lemma 6.9.2.** Let  $G$  be a connected weighted graph (with non-negative edge weights), let  $\mathbf{M}$  be its adjacency matrix, and assume that some non-negative vector  $\phi$  is an eigenvector of  $\mathbf{M}$ . Then,  $\phi$  is strictly positive.

*Proof.* Assume by way of contradiction that  $\phi$  is not strictly positive. So, there is some vertex  $u$  for which  $\phi(u) = 0$ . Thus, there must be some edge  $(u, v)$  for which  $\phi(u) = 0$  but  $\phi(v) > 0$ . We would then

$$(\mathbf{M}\phi)(u) = \sum_{(u,z) \in E} w(u, z)\phi(z) \geq w(u, v)\phi(v) > 0,$$

as all the terms  $w(u, z)$  and  $\phi(z)$  are non-negative. But, this must also equal  $\mu\phi(u) = 0$ , where  $\mu$  is the eigenvalue corresponding to  $\phi$ . This is a contradiction.

So, we conclude that  $\phi$  must be strictly positive. □

We probably won't have time to say any more about Perron-Frobenius theory, so I defer the proof of the theorem to the appendix.

## A Perron-Frobenius, continued

*Proof of Theorem 6.9.1.* Let  $\phi_1, \dots, \phi_n$  be the eigenvectors corresponding to  $\mu_1, \dots, \mu_n$ .

We start with part c. Recall that

$$\mu_1 = \max_{\mathbf{x}} \frac{\mathbf{x}^T \mathbf{M} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

Let  $\phi_1$  be an eigenvector of  $\mu_1$ , and construct the vector  $\mathbf{x}$  such that

$$\mathbf{x}(u) = |\phi_1(u)|, \text{ for all } u.$$

We will show that  $\mathbf{x}$  is an eigenvector of eigenvalue  $\mu_1$ .

We have  $\mathbf{x}^T \mathbf{x} = \phi_1^T \phi_1$ . Moreover,

$$\phi_1^T \mathbf{M} \phi_1 = \sum_{u,v} \mathbf{M}(u,v) \phi_1(u) \phi_1(v) \leq \sum_{u,v} \mathbf{M}(u,v) |\phi_1(u)| |\phi_1(v)| = \mathbf{x}^T \mathbf{M} \mathbf{x}.$$

So, the Rayleigh quotient of  $\mathbf{x}$  is at least  $\mu_1$ . As  $\mu_1$  is the maximum possible Rayleigh quotient, the Rayleigh quotient of  $\mathbf{x}$  must be  $\mu_1$  and  $\mathbf{x}$  must be an eigenvector of  $\mu_1$ .

So, we now know that  $\mathbf{M}$  has an eigenvector  $\mathbf{x}$  that is non-negative. We can then apply Lemma 6.9.2 to show that  $\mathbf{x}$  is strictly positive.

To prove part b, let  $\phi_n$  be the eigenvector of  $\mu_n$  and let  $\mathbf{y}$  be the vector for which  $\mathbf{y}(u) = |\phi_n(u)|$ . In the spirit of the previous argument, we can again show that

$$|\mu_n| = |\phi_n^T \mathbf{M} \phi_n| \leq \sum_{u,v} \mathbf{M}(u,v) \mathbf{y}(u) \mathbf{y}(v) \leq \mu_1 \mathbf{y}^T \mathbf{y} = \mu_1.$$

To show that the multiplicity of  $\mu_1$  is 1 (that is,  $\mu_2 < \mu_1$ ), consider an eigenvector  $\phi_2$ . As  $\phi_2$  is orthogonal to  $\phi_1$ , it must contain both positive and negative values. We now construct the vector  $\mathbf{y}$  such that  $\mathbf{y}(u) = |\phi_2(u)|$  and repeat the argument that we used for  $\mathbf{x}$ . We find that

$$\mu_2 = \frac{\phi_2^T \mathbf{M} \phi_2}{\phi_2^T \phi_2} \leq \frac{\mathbf{y}^T \mathbf{M} \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \leq \mu_1.$$

From here, we divide the proof into two cases. First, consider the case in which  $\mathbf{y}$  is never zero. In this case, there must be some edge  $(u, v)$  for which  $\phi_2(u) < 0 < \phi_2(v)$ . Then the above inequality must be strict because the edge  $(u, v)$  will make a negative contribution to  $\phi_2^T \mathbf{M} \phi_2$  and a positive contribution to  $\mathbf{y}^T \mathbf{M} \mathbf{y}$ .

We will argue by contradiction in the case that  $\mathbf{y}$  has a zero value. In this case, if  $\mu_2 = \mu_1$  then  $\mathbf{y}$  will be an eigenvector of eigenvalue  $\mu_1$ . This is a contradiction, as Lemma 6.9.2 says that a non-negative eigenvector cannot have a zero value. So, if  $\mathbf{y}$  has a zero value then  $\mathbf{y}^T \mathbf{M} \mathbf{y} < \mu_1$  and  $\mu_2 < \mu_1$  as well.  $\square$

The following characterization of bipartite graphs follows from similar ideas.

**Proposition A.1.** *If  $G$  is a connected graph, then  $\mu_n = -\mu_1$  if and only if  $G$  is bipartite.*

*Proof.* First, assume that  $G$  is bipartite. That is, we have a decomposition of  $V$  into sets  $U$  and  $W$  such that all edges go between  $U$  and  $W$ . Let  $\phi_1$  be the eigenvector of  $\mu_1$ . Define

$$\mathbf{x}(u) = \begin{cases} \phi_1(u) & \text{if } u \in U, \text{ and} \\ -\phi_1(u) & \text{if } u \in W. \end{cases}$$

For  $u \in U$ , we have

$$(\mathbf{M} \mathbf{x})(u) = \sum_{(u,v) \in E} \mathbf{x}(v) = - \sum_{(u,v) \in E} \phi_1(v) = -\mu_1 \phi_1(u) = -\mu_1 \mathbf{x}(u).$$

Using a similar argument for  $u \notin U$ , we can show that  $\mathbf{x}$  is an eigenvector of eigenvalue  $-\mu_1$ .

To go the other direction, assume that  $\mu_n = -\mu_1$ . We then construct  $\mathbf{y}$  as in the previous proof, and again observe

$$|\mu_n| = |\phi_n \mathbf{M} \phi_n| = \left| \sum_{u,v} \mathbf{M}(u,v) \phi_n(u) \phi_n(v) \right| \leq \sum_{u,v} \mathbf{M}(u,v) \mathbf{y}(u) \mathbf{y}(v) \leq \mu_1 \mathbf{y}^T \mathbf{y} = \mu_1.$$

For this to be an equality, it must be the case that  $\mathbf{y}$  is an eigenvalue of  $\mu_1$ , and so  $\mathbf{y} = \phi_1$ . For the first inequality above to be an equality, it must also be the case that all the terms  $\phi_n(u) \phi_n(v)$  have the same sign. In this case that sign must be negative. So, we every edge goes between a vertex for which  $\phi_n(u)$  is positive and a vertex for which  $\phi_n(v)$  is negative. Thus, the signs of  $\phi_n$  give the bi-partition.  $\square$

## B Proofs for Hoffman's lower bound on chromatic number

To prove Lemma 6.5.2, we begin with the case of  $k = 2$ . The general case follows from this one by induction.

**Lemma B.1.** *Let*

$$\mathbf{A} = \begin{bmatrix} \mathbf{B} & \mathbf{C} \\ \mathbf{C}^T & \mathbf{D} \end{bmatrix}$$

*be a symmetric matrix. Then*

$$\lambda_{\min}(\mathbf{A}) + \lambda_{\max}(\mathbf{A}) \leq \lambda_{\max}(\mathbf{B}) + \lambda_{\max}(\mathbf{D}).$$

*Proof.* Let  $\mathbf{x}$  be an eigenvector of  $\mathbf{A}$  of eigenvalue  $\lambda_{\max}(\mathbf{A})$ . To simplify formulae, let's also assume that  $\mathbf{x}$  is a unit vector. Write  $\mathbf{x} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix}$ , using the same partition as we did for  $\mathbf{A}$ .

We first consider the case in which neither  $\mathbf{x}_1$  nor  $\mathbf{x}_2$  is an all-zero vector. In this case, we set

$$\mathbf{y} = \begin{pmatrix} \frac{\|\mathbf{x}_2\|}{\|\mathbf{x}_1\|} \mathbf{x}_1 \\ -\frac{\|\mathbf{x}_1\|}{\|\mathbf{x}_2\|} \mathbf{x}_2 \end{pmatrix}.$$

The reader may verify that  $\mathbf{y}$  is also a unit vector, so

$$\mathbf{y}^T \mathbf{A} \mathbf{y} \geq \lambda_{\min}(\mathbf{A}).$$

We have

$$\begin{aligned}
\lambda_{\max}(\mathbf{A}) + \lambda_{\min}(\mathbf{A}) &\leq \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{y}^T \mathbf{A} \mathbf{y} \\
&= \mathbf{x}_1^T \mathbf{B} \mathbf{x}_1 + \mathbf{x}_1^T \mathbf{C} \mathbf{x}_2 + \mathbf{x}_2^T \mathbf{C}^T \mathbf{x}_1 + \mathbf{x}_2^T \mathbf{D} \mathbf{x}_2 + \\
&\quad + \frac{\|\mathbf{x}_2\|^2}{\|\mathbf{x}_1\|^2} \mathbf{x}_1^T \mathbf{B} \mathbf{x}_1 - \mathbf{x}_1^T \mathbf{C} \mathbf{x}_2 - \mathbf{x}_2^T \mathbf{C}^T \mathbf{x}_1 + \frac{\|\mathbf{x}_1\|^2}{\|\mathbf{x}_2\|^2} \mathbf{x}_2^T \mathbf{D} \mathbf{x}_2 \\
&= \mathbf{x}_1^T \mathbf{B} \mathbf{x}_1 + \mathbf{x}_2^T \mathbf{D} \mathbf{x}_2 + \frac{\|\mathbf{x}_2\|^2}{\|\mathbf{x}_1\|^2} \mathbf{x}_1^T \mathbf{B} \mathbf{x}_1 + \frac{\|\mathbf{x}_1\|^2}{\|\mathbf{x}_2\|^2} \mathbf{x}_2^T \mathbf{D} \mathbf{x}_2 \\
&\leq \left(1 + \frac{\|\mathbf{x}_2\|^2}{\|\mathbf{x}_1\|^2}\right) \mathbf{x}_1^T \mathbf{B} \mathbf{x}_1 + \left(1 + \frac{\|\mathbf{x}_1\|^2}{\|\mathbf{x}_2\|^2}\right) \mathbf{x}_2^T \mathbf{D} \mathbf{x}_2 \\
&\leq \lambda_{\max}(\mathbf{B}) (\|\mathbf{x}_1\|^2 + \|\mathbf{x}_2\|^2) + \lambda_{\max}(\mathbf{D}) (\|\mathbf{x}_1\|^2 + \|\mathbf{x}_2\|^2) \\
&= \lambda_{\max}(\mathbf{B}) + \lambda_{\max}(\mathbf{D}),
\end{aligned}$$

as  $\mathbf{x}$  is a unit vector.

We now return to the case in which  $\|\mathbf{x}_2\| = 0$  (or  $\|\mathbf{x}_1\| = 0$ , which is really the same case). Lemma 6.8.1 tells us that  $\lambda_{\max}(\mathbf{B}) \leq \lambda_{\max}(\mathbf{A})$ . So, it must be the case that  $\mathbf{x}_1$  is an eigenvector of eigenvalue  $\lambda_{\max}(\mathbf{A})$  of  $\mathbf{B}$ , and thus  $\lambda_{\max}(\mathbf{B}) = \lambda_{\max}(\mathbf{A})$ . To finish the proof, also observe that Lemma 6.8.1 implies

$$\lambda_{\max}(\mathbf{D}) \geq \lambda_{\min}(\mathbf{D}) \geq \lambda_{\min}(\mathbf{A}).$$

□

*Proof of Lemma 6.5.2.* For  $k = 2$ , this is exactly Lemma B.1. For  $k > 2$ , we apply induction. Let

$$B = \begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \cdots & \mathbf{A}_{1,k-1} \\ \mathbf{A}_{1,2}^T & \mathbf{A}_{2,2} & \cdots & \mathbf{A}_{2,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{1,k-1}^T & \mathbf{A}_{2,k-1}^T & \cdots & \mathbf{A}_{k-1,k-1} \end{bmatrix}.$$

Lemma 6.8.1 now implies.

$$\lambda_{\min}(\mathbf{B}) \geq \lambda_{\min}(\mathbf{A}).$$

Applying Lemma B.1 to  $B$  and the  $k$ th row and column of  $\mathbf{A}$ , we find

$$\begin{aligned}
\lambda_{\min}(\mathbf{A}) + \lambda_{\max}(\mathbf{A}) &\leq \lambda_{\max}(\mathbf{B}) + \lambda_{\max}(\mathbf{A}_{k,k}) \\
&\leq -(k-2)\lambda_{\min}(\mathbf{B}) + \sum_{i=1}^{k-1} \lambda_{\max}(\mathbf{A}_{i,i}) + \lambda_{\max}(\mathbf{A}_{k,k}) \quad (\text{by induction}) \\
&\leq -(k-1)\lambda_{\min}(\mathbf{A}) + \sum_{i=1}^k \lambda_{\max}(\mathbf{A}_{i,i}),
\end{aligned}$$

which proves the lemma. □

## References

- [AH77a] Kenneth Appel and Wolfgang Haken. Every planar map is four colorable part i. discharging. *Illinois Journal of Mathematics*, 21:429–490, 1977.
- [AH77b] Kenneth Appel and Wolfgang Haken. Every planar map is four colorable part ii. reducibility. *Illinois Journal of Mathematics*, 21:491–567, 1977.
- [AK97] Alon and Kahale. A spectral technique for coloring random 3-colorable graphs. *SICOMP: SIAM Journal on Computing*, 26, 1997.
- [BRSW12] Boaz Barak, Anup Rao, Ronen Shaltiel, and Avi Wigderson. 2-source dispersers for  $n^{o(1)}$  entropy, and ramsey graphs beating the frankl-wilson construction. *Annals of Mathematics*, pages 1483–1543, 2012.
- [Erd47] Paul Erdős. Some remarks on the theory of graphs. *Bulletin of the American Mathematical Society*, 53(4):292–294, 1947.
- [GN08] C.D. Godsil and M.W. Newman. Eigenvalue bounds for independent sets. *Journal of Combinatorial Theory, Series B*, 98(4):721 – 734, 2008.

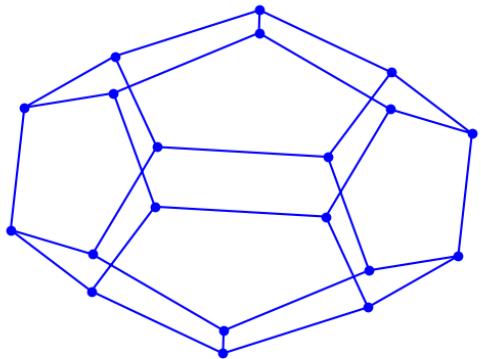
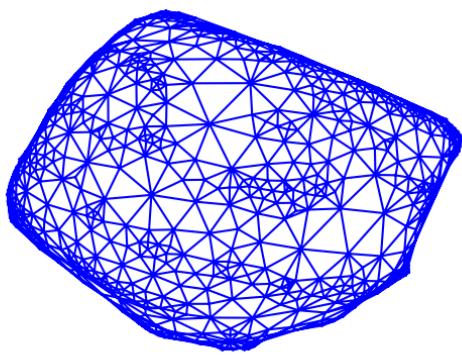
## Fiedler's Theorems on Nodal Domains

Daniel A. Spielman

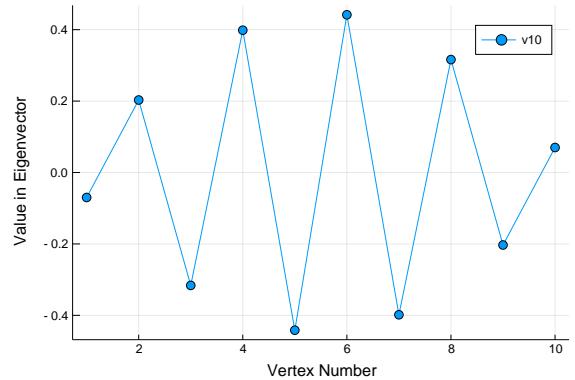
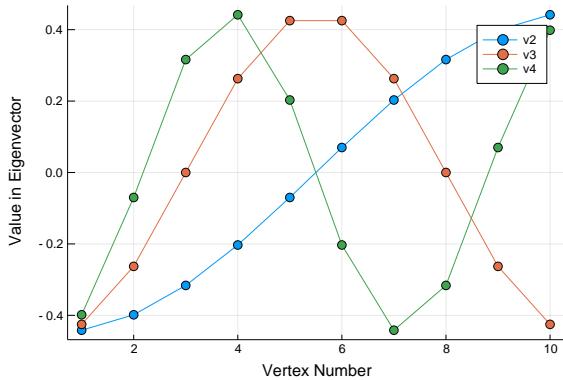
September 19, 2018

## 7.1 Overview

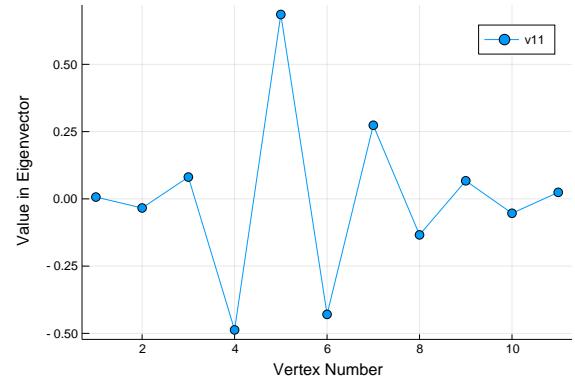
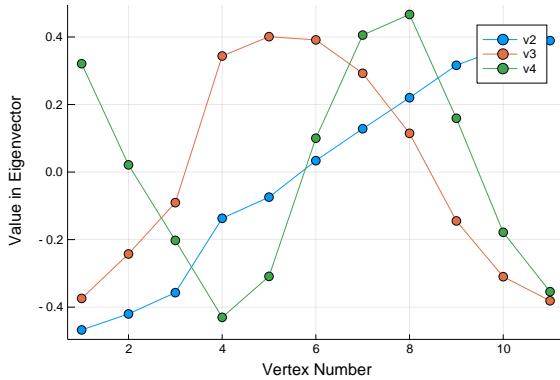
In today's lecture we will justify some of the behavior we observed when using eigenvectors to draw graphs in the first lecture. First, recall some of the drawings we made of graphs:



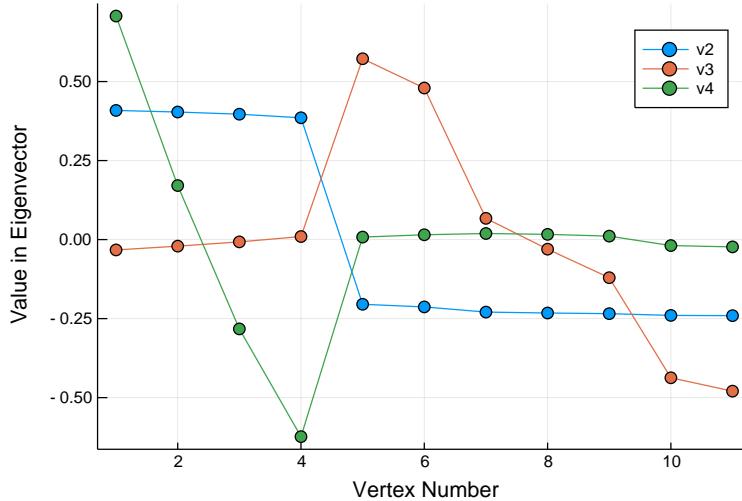
We will show that the subgraphs obtained in the right and left halves of each image are connected. Path graphs exhibited more interesting behavior: their  $k$ th eigenvector changes sign  $k$  times:



Here are the analogous plots for a path graph with edge weights randomly chosen in  $[0, 1]$ :



Here are the first few eigenvectors of another:



```

Random.seed!(1)
M = spdiagm(1=>rand(10))
M = M + M'
L = lap(M)
E = eigen(Matrix(L))
Plots.plot(E.vectors[:,2],label="v2",marker = 5)
Plots.plot!(E.vectors[:,3],label="v3",marker = 5)
Plots.plot!(E.vectors[:,4],label="v4",marker = 5)
xlabel!("Vertex Number")
ylabel!("Value in Eigenvector")
savefig("rpath2v24.pdf")

```

We see that the  $k$ th eigenvector still changes sign  $k$  times. We will see that this always happens. These are some of Fiedler's theorems about "nodal domains". Nodal domains are the connected parts of a graph on which an eigenvector is negative or positive.

## 7.2 Sylvester's Law of Interia

Let's begin with something obvious.

**Claim 7.2.1.** *If  $A$  is positive semidefinite, then so is  $B^T AB$  for every matrix  $B$ .*

*Proof.* For any  $x$ ,

$$x^T B^T ABx = (Bx)^T A(Bx) \geq 0,$$

since  $A$  is positive semidefinite.  $\square$

In this lecture, we will make use of Sylvester's law of inertia, which is a powerful generalization of this fact. I will state and prove it now.

**Theorem 7.2.2** (Sylvester's Law of Intertia). *Let  $A$  be any symmetric matrix and let  $B$  be any non-singular matrix. Then, the matrix  $BAB^T$  has the same number of positive, negative and zero eigenvalues as  $A$ .*

Note that if the matrix  $B$  were orthonormal, or if we used  $B^{-1}$  in place of  $B^T$ , then these matrices would have the same eigenvalues. What we are doing here is different, and corresponds to a change of variables.

*Proof.* It is clear that  $A$  and  $BAB^T$  have the same rank, and thus the same number of zero eigenvalues.

We will prove that  $A$  has at least as many positive eigenvalues as  $BAB^T$ . One can similarly prove that  $A$  has at least as many negative eigenvalues, which proves the theorem.

Let  $\gamma_1, \dots, \gamma_k$  be the positive eigenvalues of  $BAB^T$  and let  $Y_k$  be the span of the corresponding eigenvectors. Now, let  $S_k$  be the span of the vectors  $B^T y$ , for  $y \in Y_k$ . As  $B$  is non-singular,  $S_k$  has dimension  $k$ . Let  $\alpha_1 \geq \dots \geq \alpha_n$  be the eigenvalues of  $A$ . By the Courant-Fischer Theorem, we have

$$\alpha_k = \max_{\substack{S \subseteq \mathbb{R}^n \\ \dim(S)=k}} \min_{x \in S} \frac{x^T Ax}{x^T x} \geq \min_{x \in S_k} \frac{x^T Ax}{x^T x} = \min_{y \in Y_k} \frac{y^T BAB^T y}{y^T BB^T y} \geq \frac{\gamma_k y^T y}{y^T BB^T y} > 0.$$

So,  $A$  has at least  $k$  positive eigenvalues (The point here is that the denominators are always positive, so we only need to think about the numerators.)

To finish, either apply the symmetric argument to the negative eigenvalues, or apply the same argument with  $B^{-1}$ .  $\square$

## 7.3 Weighted Trees

We will now examine a theorem of Fiedler [Fie75].

**Theorem 7.3.1.** Let  $T$  be a weighted tree graph on  $n$  vertices, let  $\mathbf{L}_T$  have eigenvalues  $0 = \lambda_1 < \lambda_2 \dots \leq \lambda_n$ , and let  $\psi_k$  be an eigenvector of  $\lambda_k$ . If there is no vertex  $u$  for which  $\psi_k(u) = 0$ , then there are exactly  $k - 1$  edges for which  $\psi_k(u)\psi_k(v) < 0$ .

One can extend this theorem to accomodate zero entries and prove that the eigenvector changes  $k - 1$  times. We will just prove this theorem for weighted path graphs.

Our analysis will rest on an understanding of Laplacians of paths that are allowed to have negative edges weights.

**Lemma 7.3.2.** Let  $\mathbf{M}$  be the Laplacian matrix of a weighted path that can have negative edge weights:

$$\mathbf{M} = \sum_{1 \leq a < n} w_{a,a+1} \mathbf{L}_{a,a+1},$$

where the weights  $w_{a,a+1}$  are non-zero and we recall that  $\mathbf{L}_{a,b}$  is the Laplacian of the edge  $(a, b)$ . The number of negative eigenvalues of  $\mathbf{M}$  equals the number of negative edge weights.

*Proof.* Note that

$$\mathbf{x}^T \mathbf{M} \mathbf{x} = \sum_{(u,v) \in E} w_{u,v} (\mathbf{x}(u) - \mathbf{x}(v))^2.$$

We now perform a change of variables that will diagonalize the matrix  $\mathbf{M}$ . Let  $\delta(1) = \mathbf{x}(1)$ , and for every  $a > 1$  let  $\delta(a) = \mathbf{x}(a) - \mathbf{x}(a-1)$ .

Every variable  $\mathbf{x}(1), \dots, \mathbf{x}(n)$  can be expressed as a linear combination of the variables  $\delta(1), \dots, \delta(n)$ . In particular,

$$\mathbf{x}(a) = \delta(1) + \delta(2) + \dots + \delta(a).$$

So, there is a square matrix  $\mathbf{L}$  of full rank such that

$$\mathbf{x} = \mathbf{L}\delta.$$

By Sylvester's law of inertia, we know that

$$\mathbf{L}^T \mathbf{M} \mathbf{L}$$

has the same number of positive, negative, and zero eigenvalues as  $\mathbf{M}$ . On the other hand,

$$\delta^T \mathbf{L}^T \mathbf{M} \mathbf{L} \delta = \sum_{1 \leq a < n} w_{a,a+1} (\delta(a))^2.$$

So, this matrix clearly has one zero eigenvalue, and as many negative eigenvalues as there are negative  $w_{a,a+1}$ .  $\square$

*Proof of Theorem 7.3.1. We assume that  $\lambda_k$  has multiplicity 1. One can prove it, but we will skip it.*

Let  $\Psi_k$  denote the diagonal matrix with  $\psi_k$  on the diagonal, and let  $\lambda_k$  be the corresponding eigenvalue. Consider the matrix

$$\mathbf{M} = \Psi_k (\mathbf{L}_P - \lambda_k \mathbf{I}) \Psi_k.$$

The matrix  $\mathbf{L}_P - \lambda_k \mathbf{I}$  has one zero eigenvalue and  $k - 1$  negative eigenvalues. As we have assumed that  $\psi_k$  has no zero entries,  $\Psi_k$  is non-singular, and so we may apply Sylvester's Law of Intertia to show that the same is true of  $\mathbf{M}$ .

I claim that

$$\mathbf{M} = \sum_{(u,v) \in E} w_{u,v} \psi_k(u) \psi_k(v) \mathbf{L}_{u,v}.$$

To see this, first check that this agrees with the previous definition on the off-diagonal entries. To verify that these expression agree on the diagonal entries, we will show that the sum of the entries in each row of both expressions agree. As we know that all the off-diagonal entries agree, this implies that the diagonal entries agree. We compute

$$\Psi_k(\mathbf{L}_P - \lambda_k \mathbf{I}) \Psi_k \mathbf{1} = \Psi_k(\mathbf{L}_P - \lambda_k \mathbf{I}) \psi_k = \Psi_k(\lambda_k \psi_k - \lambda_k \psi_k) = \mathbf{0}.$$

As  $\mathbf{L}_{u,v} \mathbf{1} = \mathbf{0}$ , the row sums agree. Lemma 7.3.2 now tells us that the matrix  $\mathbf{M}$ , and thus  $\mathbf{L}_P - \lambda_k \mathbf{I}$ , has as many negative eigenvalues as there are edges  $(u, v)$  for which  $\psi_k(u) \psi_k(v) < 0$ .  $\square$

## 7.4 More linear algebra

There are a few more facts from linear algebra that we will need for the rest of this lecture. We stop to prove them now.

### 7.4.1 The Perron-Frobenius Theorem for Laplacians

In Lecture 3, we proved the Perron-Frobenius Theorem for non-negative matrices. I wish to quickly observe that this theory may also be applied to Laplacian matrices, to principal sub-matrices of Laplacian matrices, and to any matrix with non-positive off-diagonal entries. The difference is that it then involves the eigenvector of the smallest eigenvalue, rather than the largest eigenvalue.

**Corollary 7.4.1.** *Let  $M$  be a matrix with non-positive off-diagonal entries, such that the graph of the non-zero off-diagonally entries is connected. Let  $\lambda_1$  be the smallest eigenvalue of  $M$  and let  $\mathbf{v}_1$  be the corresponding eigenvector. Then  $\mathbf{v}_1$  may be taken to be strictly positive, and  $\lambda_1$  has multiplicity 1.*

*Proof.* Consider the matrix  $A = \sigma \mathbf{I} - M$ , for some large  $\sigma$ . For  $\sigma$  sufficiently large, this matrix will be non-negative, and the graph of its non-zero entries is connected. So, we may apply the Perron-Frobenius theory to  $A$  to conclude that its largest eigenvalue  $\alpha_1$  has multiplicity 1, and the corresponding eigenvector  $\mathbf{v}_1$  may be assumed to be strictly positive. We then have  $\lambda_1 = \sigma - \alpha_1$ , and  $\mathbf{v}_1$  is an eigenvector of  $\lambda_1$ .  $\square$

### 7.4.2 Eigenvalue Interlacing

We will often use the following elementary consequence of the Courant-Fischer Theorem. I will assign it as homework.

**Theorem 7.4.2** (Eigenvalue Interlacing). *Let  $\mathbf{A}$  be an  $n$ -by- $n$  symmetric matrix and let  $\mathbf{B}$  be a principal submatrix of  $\mathbf{A}$  of dimension  $n - 1$  (that is,  $\mathbf{B}$  is obtained by deleting the same row and column from  $\mathbf{A}$ ). Then,*

$$\alpha_1 \geq \beta_1 \geq \alpha_2 \geq \beta_2 \geq \cdots \geq \alpha_{n-1} \geq \beta_{n-1} \geq \alpha_n,$$

where  $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_n$  and  $\beta_1 \geq \beta_2 \geq \cdots \geq \beta_{n-1}$  are the eigenvalues of  $\mathbf{A}$  and  $\mathbf{B}$ , respectively.

## 7.5 Fiedler's Nodal Domain Theorem

Given a graph  $G = (V, E)$  and a subset of vertices,  $W \subseteq V$ , recall that the *graph induced by  $G$  on  $W$*  is the graph with vertex set  $W$  and edge set

$$\{(i, j) \in E : i \in W \text{ and } j \in W\}.$$

This graph is sometimes denoted  $G(W)$ .

**Theorem 7.5.1** ([Fie75]). *Let  $G = (V, E, w)$  be a weighted connected graph, and let  $L_G$  be its Laplacian matrix. Let  $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$  be the eigenvalues of  $L_G$  and let  $\psi_1, \dots, \psi_n$  be the corresponding eigenvectors. For any  $k \geq 2$ , let*

$$W_k = \{i \in V : \psi_k(i) \geq 0\}.$$

*Then, the graph induced by  $G$  on  $W_k$  has at most  $k - 1$  connected components.*

*Proof.* To see that  $W_k$  is non-empty, recall that  $\psi_1 = \mathbf{1}$  and that  $\psi_k$  is orthogonal to  $\psi_1$ . So,  $\psi_k$  must have both positive and negative entries.

Assume that  $G(W_k)$  has  $t$  connected components. After re-ordering the vertices so that the vertices in one connected component of  $G(W_k)$  appear first, and so on, we may assume that  $L_G$  and  $\psi_k$  have the forms

$$L_G = \begin{bmatrix} B_1 & \mathbf{0} & \mathbf{0} & \cdots & C_1 \\ \mathbf{0} & B_2 & \mathbf{0} & \cdots & C_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & B_t & C_t \\ C_1^T & C_2^T & \cdots & C_t^T & D \end{bmatrix} \quad \psi_k = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_t \\ \mathbf{y} \end{pmatrix},$$

and

$$\begin{bmatrix} B_1 & \mathbf{0} & \mathbf{0} & \cdots & C_1 \\ \mathbf{0} & B_2 & \mathbf{0} & \cdots & C_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & B_t & C_t \\ C_1^T & C_2^T & \cdots & C_t^T & D \end{bmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_t \\ \mathbf{y} \end{pmatrix} = \lambda_k \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_t \\ \mathbf{y} \end{pmatrix}.$$

The first  $t$  sets of rows and columns correspond to the  $t$  connected components. So,  $\mathbf{x}_i \geq 0$  for  $1 \leq i \leq t$  and  $\mathbf{y} < 0$  (when I write this for a vector, I mean it holds for each entry). We also know

that the graph of non-zero entries in each  $B_i$  is connected, and that each  $C_i$  is non-positive, and has at least one non-zero entry (otherwise the graph  $G$  would be disconnected).

We will now prove that the smallest eigenvalue of  $B_i$  is smaller than  $\lambda_k$ . We know that

$$B_i \mathbf{x}_i + C_i \mathbf{y} = \lambda_k \mathbf{x}_i.$$

As each entry in  $C_i$  is non-positive and  $\mathbf{y}$  is strictly negative, each entry of  $C_i \mathbf{y}$  is non-negative and some entry of  $C_i \mathbf{y}$  is positive. Thus,  $\mathbf{x}_i$  cannot be all zeros,

$$B_i \mathbf{x}_i = \lambda_k \mathbf{x}_i - C_i \mathbf{y} \leq \lambda_k \mathbf{x}_i$$

and

$$\mathbf{x}_i^T B_i \mathbf{x}_i \leq \lambda_k \mathbf{x}_i^T \mathbf{x}_i.$$

If  $\mathbf{x}_i$  has *any* zero entries, then the Perron-Frobenius theorem tells us that  $\mathbf{x}_i$  cannot be an eigenvector of smallest eigenvalue, and so the smallest eigenvalue of  $B_i$  is less than  $\lambda_k$ . On the other hand, if  $\mathbf{x}_i$  is strictly positive, then  $\mathbf{x}_i^T C_i \mathbf{y} > 0$ , and

$$\mathbf{x}_i^T B_i \mathbf{x}_i = \lambda_k \mathbf{x}_i^T \mathbf{x}_i - \mathbf{x}_i^T C_i \mathbf{y} < \lambda_k \mathbf{x}_i^T \mathbf{x}_i.$$

Thus, the matrix

$$\begin{bmatrix} B_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & B_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & B_t \end{bmatrix}$$

has at least  $t$  eigenvalues less than  $\lambda_k$ . By the eigenvalue interlacing theorem, this implies that  $L_G$  has at least  $t$  eigenvalues less than  $\lambda_k$ . We may conclude that  $t$ , the number of connected components of  $G(W_k)$ , is at most  $k - 1$ .  $\square$

We remark that Fiedler actually proved a somewhat stronger theorem. He showed that the same holds for

$$W = \{i : \psi_k(i) \geq t\},$$

for every  $t \leq 0$ .

This theorem breaks down if we instead consider the set

$$W = \{i : \psi_k(i) > 0\}.$$

The star graphs provide counter-examples.

## References

- [Fie75] M. Fiedler. A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory. *Czechoslovak Mathematical Journal*, 25(100):618–633, 1975.

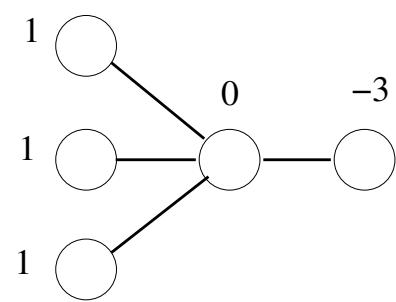


Figure 7.1: The star graph on 5 vertices, with an eigenvector of  $\lambda_2 = 1$ .

## Testing Isomorphism of Graphs with Distinct Eigenvalues

Daniel A. Spielman

September 24, 2018

## 8.1 Introduction

I will present an algorithm of Leighton and Miller [LM82] for testing isomorphism of graphs in which all eigenvalues have multiplicity 1. This algorithm was never published, as the results were technically subsumed by those in a paper of Babai, Grigoriev and Mount [BGM82], which gave a polynomial time algorithm for testing isomorphism of graphs in which all eigenvalues have multiplicity bounded by a constant.

I present the weaker result in the interest of simplicity.

Testing isomorphism of graphs is a notorious problem. Until very recently, the fastest-known algorithm for it took time  $2\sqrt{O(n \log n)}$  (See [Bab81, BL83, ZKT85]). Babai [Bab16] recently announced a breakthrough that reduces the complexity to  $2^{(\log n)^{O(1)}}$ .

However, testing graph isomorphism seems easy in almost all practical instances. Today's lecture and one next week will give you some idea as to why.

## 8.2 Graph Isomorphism

Recall that two graphs  $G = (V, E)$  and  $H = (V, F)$  are isomorphic if there exists a permutation  $\pi$  of  $V$  such that

$$(a, b) \in E \iff (\pi(a), \pi(b)) \in F.$$

Of course, we can express this relation in terms of matrices associated with the graphs. It doesn't matter much which matrices we use. So for this lecture we will use the adjacency matrices.

Every permutation may be realized by a *permutation matrix*. For the permutation  $\pi$ , this is the matrix  $\Pi$  with entries given by

$$\Pi(a, b) = \begin{cases} 1 & \text{if } \pi(a) = b \\ 0 & \text{otherwise.} \end{cases}$$

For a vector  $\psi$ , we see<sup>1</sup> that

$$(\Pi\psi)(a) = \psi(\pi(a)).$$

---

<sup>1</sup>I hope I got that right. It's very easy to confuse the permutation and its inverse.

Let  $A$  be the adjacency matrix of  $G$  and let  $B$  be the adjacency matrix of  $H$ . We see that  $G$  and  $H$  are isomorphic if and only if there exists a permutation matrix  $\Pi$  such that

$$\Pi A \Pi^T = B.$$

### 8.3 Using Eigenvalues and Eigenvectors

If  $G$  and  $H$  are isomorphic, then  $A$  and  $B$  must have the same eigenvalues. However, there are many pairs of graphs that are non-isomorphic but which have the same eigenvalues. We will see some tricky ones next lecture. But, for now, we note that if  $A$  and  $B$  have different eigenvalues, then we know that the corresponding graphs are non-isomorphic, and we don't have to worry about them.

For the rest of this lecture, we will assume that  $A$  and  $B$  have the same eigenvalues, and that each of these eigenvalues has multiplicity 1. We will begin our study of this situation by considering some cases in which testing isomorphism is easy.

Recall that we can write

$$A = \Psi \Lambda \Psi^T,$$

where  $\Lambda$  is the diagonal matrix of eigenvalues of  $A$  and  $\Psi$  is an orthonormal matrix holding its eigenvectors. If  $B$  has the same eigenvalues, we can write

$$B = \Phi \Lambda \Phi^T.$$

If  $\Pi$  is the matrix of an isomorphism from  $G$  to  $H$ , then

$$\Pi \Psi \Lambda \Psi^T \Pi^T = \Phi \Lambda \Phi^T.$$

As each entry of  $\Lambda$  is distinct, this looks like it would imply  $\Pi \Psi = \Phi$ . But, the eigenvectors (columns of  $\Phi$  and  $\Psi$ ) are only determined up to sign. So, it just implies

$$\Pi \Psi = \Phi S,$$

where  $S$  is a diagonal matrix with  $\pm 1$  entries on its diagonal.

**Lemma 8.3.1.** *Let  $A = \Psi \Lambda \Psi^T$  and  $B = \Phi \Lambda \Phi^T$  where  $\Lambda$  is a diagonal matrix with distinct entries and  $\Psi$  and  $\Phi$  are orthogonal matrices. A permutation matrix  $\Pi$  satisfies  $\Pi A \Pi^T = B$  if and only if there exists a diagonal  $\pm 1$  matrix  $S$  for which*

$$\Pi \Psi = \Phi S.$$

*Proof.* Let  $\psi_1, \dots, \psi_n$  be the columns of  $\Psi$  and let  $\phi_1, \dots, \phi_n$  be the columns of  $\Phi$ . Assuming there is a  $\Pi$  for which  $\Pi A \Pi^T = B$ ,

$$\Phi \Lambda \Phi^T = \sum_{i=1}^n \phi_i \lambda_i \phi_i^T = \sum_{i=1}^n (\Pi \psi_i) \lambda_i (\psi_i^T \Pi^T),$$

which implies that for all  $i$

$$\phi_i \phi_i^T = (\Pi \psi_i)(\Pi \psi_i)^T.$$

This in turn implies that

$$\phi_i = \pm \Pi \psi_i.$$

To go the other direction, assume  $\Pi \Psi = \Phi S$ . Then,

$$\Pi A \Pi^T = \Pi \Psi \Lambda \Psi^T \Pi^T = \Phi S \Lambda S \Phi^T = \Phi \Lambda S S \Phi^T = \Phi \Lambda \Phi^T = B,$$

as  $S$  and  $\Lambda$  are diagonal and thus commute, and  $S^2 = I$ .  $\square$

Our algorithm for testing isomorphism will determine all such matrices  $S$ . Let  $\mathcal{S}$  be the set of all diagonal  $\pm 1$  matrices. We will find diagonal matrices  $S \in \mathcal{S}$  such that the set of rows of  $\Phi S$  is the same as the set of rows of  $\Psi$ . As the rows of  $\Psi$  are indexed by vertices  $a \in V$ , we will write the row indexed by  $a$  as the row-vector

$$\mathbf{v}_a \stackrel{\text{def}}{=} (\psi_1(a), \dots, \psi_n(a)).$$

Similarly denote the rows of  $\Phi$  by vectors  $\mathbf{u}_a$ . In this notation, we are searching for matrices  $S \in \mathcal{S}$  for which the set of vectors  $\{\mathbf{v}_a\}_{a \in V}$  is identical to the set of vectors  $\{\mathbf{u}_a S\}_{a \in V}$ . We have thus transformed the graph isomorphism problem into a problem about vectors:

## 8.4 An easy case

I will say that an eigenvector  $\psi_i$  is *helpful* if for all  $a \neq b \in V$ ,  $|\psi_i(a)| \neq |\psi_i(b)|$ . In this case, it is very easy to test if  $G$  and  $H$  are isomorphic, because this helpful vector gives us a canonical name for every vertex. If  $\Pi$  is an isomorphism from  $G$  to  $H$ , then  $\Pi \psi_i$  must be an eigenvector of  $B$ . In fact, it must be  $\pm \phi_i$ . If the sets of absolute values of entries of  $\psi_i$  and  $\phi_i$  are the same, then we may find the permutation that maps  $A$  to  $B$  by mapping every vertex  $a$  to the vertex  $b$  for which  $|\psi_i(a)| = |\phi_i(b)|$ .

The reason that I put absolute values in the definition of helpful, rather than just taking values, is that eigenvectors are only determined up to sign. On the other hand, a single eigenvector determines the isomorphism if  $\psi_i(a) \neq \psi_i(b)$  for all  $a \neq b$  and there is a canonical way to choose a sign for the vector  $\psi_i$ . For example, if the sum of the entries in  $\psi_i$  is not zero, we can choose its sign to make the sum positive. In fact, unless  $\psi_i$  and  $-\psi_i$  have exactly the same set of values, there is a canonical choice of the sign for this vector.

Even if there is no canonical choice of sign for this vector, it leaves at most two choices for the isomorphism.

## 8.5 All the Automorphisms

The graph isomorphism problem is complicated by the fact that there can be many isomorphisms from one graph to another. So, any algorithm for finding isomorphisms must be able to find many

of them.

Recall that an *automorphism* of a graph is an isomorphism from the graph to itself. These form a group which we denote  $\text{aut}(G)$ : if  $\Pi$  and  $\Gamma$  are automorphisms of  $A$  then so is  $\Pi\Gamma$ . Let  $\mathcal{A} \subseteq \mathcal{S}$  denote the corresponding set of diagonal  $\pm 1$  matrices. The set  $\mathcal{A}$  is in fact a group and is isomorphic to  $\text{aut}(G)$ .

Here is a way to make this isomorphism very concrete: Lemma 8.3.1 implies that the  $\Pi \in \text{aut}(G)$  and the  $S \in \mathcal{A}$  are related by

$$\Pi = \Psi S \Psi^T \quad \text{and} \quad S = \Psi^T \Pi \Psi.$$

As diagonal matrices commute, we have that for every  $\Pi_1$  and  $\Pi_2$  in  $\text{aut}(G)$  and for  $S_1 = \Psi^T \Pi_1 \Psi$  and  $S_2 = \Psi^T \Pi_2 \Psi$ ,

$$\Pi_1 \Pi_2 = \Psi S_1 \Psi^T \Psi S_2 \Psi^T = \Psi S_1 S_2 \Psi^T = \Psi S_2 S_1 \Psi^T = \Psi S_2 \Psi^T \Psi S_1 \Psi^T = \Pi_2 \Pi_1.$$

Thus, the automorphism group of a graph with distinct eigenvalues is commutative, and it is isomorphic to a subgroup of  $\mathcal{S}$ .

It might be easier to think about these subgroups by realizing that they are isomorphic to subspaces of  $(\mathbf{Z}/2\mathbf{Z})^n$ . Let  $f : \mathcal{S} \rightarrow (\mathbf{Z}/2\mathbf{Z})^n$  be the function that maps the group of diagonal matrices with  $\pm 1$  entries to vectors  $t$  modulo 2 by setting  $t(i)$  so that  $S(i, i) = (-1)^{t(i)}$ . You should check that this is a group homomorphism:  $f(S_1 S_2) = f(S_1) + f(S_2)$ . You should also confirm that  $f$  is invertible.

For today's lecture, we will focus on the problem of finding the group of automorphisms of a graph with distinct eigenvalues. We will probably save the slight extension to finding isomorphisms for homework. Note that we will not try to list all the isomorphisms, as there could be many. Rather, we will give a basis of the corresponding subspace of  $(\mathbf{Z}/2\mathbf{Z})^n$ .

## 8.6 Equivalence Classes of Vertices

Recall that the *orbit* of an element under the action of a group is the set of elements to which it is mapped by the elements of the group. Concretely, the orbit of a vertex  $a$  in the graph is the set of vertices to which it can be mapped by automorphisms. We will discover the orbits by realizing that the orbit of a vertex  $a$  is the set of  $b$  for which  $v_a S = v_b$  for some  $S \in \mathcal{A}$ .

The set of orbits of vertices forms a partition of the vertices. We say that a partition of the vertices is *valid* if every orbit is contained entirely within one set in the partition. That is, each class of the partition is a union of orbits. Our algorithm will proceed by constructing a valid partition of the vertices and then splitting classes in the partition until each is exactly an orbit.

Recall that a set is *stabilized* by a group if the set is unchanged when the group acts on all of its members. We will say that a group  $\mathcal{G} \subseteq \mathcal{S}$  stabilizes a set of vertices  $C$  if it stabilizes the set of vectors  $\{v_a\}_{a \in C}$ . Thus,  $\mathcal{A}$  is the group that stabilizes  $V$ .

An orbit is stabilized by  $\mathcal{A}$ , and so are unions of orbits and thus classes of valid partitions. We would like to construct the subgroup of  $\mathcal{S}$  that stabilizes each orbit  $C_j$ . However, I do not yet see how to do that directly. Instead, we will construct a particular valid partition of the vertices, and find for each class in the partition  $C_j$  the subgroup of  $\mathcal{A}_j \subseteq \mathcal{S}$  that stabilizes  $C_j$ , where here we are considering the actions of matrices  $\mathbf{S} \in \mathcal{S}$  on vectors  $\mathbf{v}_a$ . In fact,  $\mathcal{A}_j$  will act transitively<sup>2</sup> on the class  $C_j$ . As  $\mathcal{A}$  stabilizes every orbit, and thus every union of orbits, it is a subgroup of  $\mathcal{A}_j$ . In fact,  $\mathcal{A}$  is exactly the intersection of all the groups  $\mathcal{A}_j$ .

We now observe that we can use linear algebra to efficiently construct  $\mathcal{A}$  from the groups  $\mathcal{A}_j$  by exploiting the isomorphism between  $\mathbf{S}$  and  $(\mathbf{Z}/2)^n$ . Each subgroup  $\mathcal{A}_j$  is isomorphic to a subgroup of  $(\mathbf{Z}/2)^n$ . Each subgroup of  $(\mathbf{Z}/2)^n$  is precisely a vector space modulo 2, and thus may be described by a basis. It will eventually become clear that by “compute  $\mathcal{A}_j$ ” we mean to compute such a basis. From the basis, we may compute a basis of the nullspace. The subgroup of  $(\mathbf{Z}/2)^n$  corresponding to  $\mathcal{A}$  is then the nullspace of the span of the nullspaces of the subspaces corresponding to the  $\mathcal{A}_j$ . We can compute all these using Gaussian elimination.

## 8.7 The first partition

We may begin by dividing vertices according to the absolute values of their entries in eigenvectors. That is, if  $|\psi_i(a)| \neq |\psi_i(b)|$  for some  $i$ , then we may place vertices  $a$  and  $b$  in different classes, as there can be no  $\mathbf{S} \in \mathcal{S}$  for which  $\mathbf{v}_a \mathbf{S} = \mathbf{v}_b$ . The partition that we obtain this way is thus valid, and is the starting point of our algorithm.

## 8.8 Unbalanced vectors

We say that an eigenvector  $\psi_i$  is *unbalanced* if there is some value  $x$  for which

$$|\{a : \psi_i(a) = x\}| \neq |\{a : \psi_i(a) = -x\}|.$$

Such vectors cannot change sign in an automorphism. That is,  $\mathbf{S}(i, i)$  must equal 1. The reason is that an automorphism with  $\mathbf{S}(i, i) = -1$  must induce a bijection between the two sets above, but this is impossible if their sizes are different.

Thus, an unbalanced vector tells us that all vertices for which  $\psi_i(a) = x$  are in different orbits from those for which  $\psi_i(a) = -x$ . This lets us refine classes.

We now extend this idea in two ways. First, we say that  $\psi_i$  is *unbalanced* on a class  $C$  if there is some value  $x$  for which

$$|\{a \in C : \psi_i(a) = x\}| \neq |\{a \in C : \psi_i(a) = -x\}|.$$

By the same reasoning, we can infer that the sign of  $\mathbf{S}(i, i)$  must be fixed to 1. Assuming, as will be the case, that  $C$  is a class in a valid partition and thus a union of orbits, we are now able to

---

<sup>2</sup>That is, for every  $a$  and  $b$  in  $C_j$ , there is an  $\mathbf{S} \in \mathcal{A}_j$  for which  $\mathbf{v}_a \mathbf{S} = \mathbf{v}_b$ .

split  $C$  into two smaller classes

$$C_0 = \{a \in C : \psi_i(a) = x\} \quad \text{and} \quad C_1 = \{a \in C : \psi_i(a) = -x\}.$$

The partition we obtain by splitting  $C$  into  $C_1$  and  $C_2$  is thus also valid. Of course, it is only useful if both sets are non-empty.

Finally, we consider vectors formed from products of eigenvectors. For  $R \subseteq \{1, \dots, n\}$ , define  $\psi_R$  to be the component-wise product of the  $\psi_i$  for  $i \in R$ :

$$\psi_R(a) = \prod_{i \in R} \psi_i(a).$$

We say that the vector  $\psi_R$  is unbalanced on class  $C$  if there is some value  $x$  for which

$$|\{a \in C : \psi_R(a) = x\}| \neq |\{a \in C : \psi_R(a) = -x\}|.$$

An unbalanced vector of this form again tells us that the vertices in the two sets belong to different orbits. So, if both sets are nonempty we can use such a vector to split the class  $C$  in two to obtain a more refined valid partition. It also provides some relations between the entries of  $S$ , but we will not exploit those.

We say that a vector is *balanced* if it is not unbalanced.

We say that a subset of the vertices  $C \subseteq V$  is *balanced* if *every* non-constant product of eigenvectors is balanced on  $C$ . Thus, orbits are balanced. Our algorithm will partition the vertices into balanced classes.

My confusion over this lecture stemmed from thinking that all balanced classes must be orbits. But, I don't know if this is true.

**Question:** Is every balanced class an orbit of  $\mathcal{A}$ ?

## 8.9 The structure of the balanced classes

Let  $C_j$  be a balanced class. By definition, the product of every subset of eigenvectors is either constant or balanced on  $C_j$ . We say that a subset of eigenvectors  $Q$  is *independent* on  $C_j$  if all products of subsets of eigenvectors in  $Q$  are balanced on  $C_j$  (except for the empty product). In particular, none of these eigenvectors is zero or constant on  $C_j$ . Construct a matrix  $M_{C_j, Q}$  whose rows are indexed by vertices in  $a \in C_j$ , whose columns are indexed by subsets  $R \subseteq Q$ , and whose entries are given by

$$M_{C_j, Q}(a, R) = \operatorname{sgn}(\psi_R(a)), \text{ where I recall } \operatorname{sgn}(x) = \begin{cases} 1 & \text{if } x > 0 \\ -1 & \text{if } x < 0, \text{ and} \\ 0 & \text{if } x = 0. \end{cases}$$

**Lemma 8.9.1.** *If  $Q$  is independent on  $C$  then the columns of  $M_{C_j, Q}$  are orthogonal.*

*Proof.* Let  $R_1$  and  $R_2$  index two columns of  $M_{C,Q}$ . That is,  $R_1$  and  $R_2$  are two different subsets of  $Q$ . Let  $R_0$  be their symmetric difference. We have

$$\begin{aligned} M_{C,Q}(a, R_1)M_{C,Q}(a, R_2) &= \text{sgn}(\psi_{R_1}(a))\text{sgn}(\psi_{R_2}(a)) = \\ &\prod_{i \in R_1} \text{sgn}(\psi_i(a)) \prod_{i \in R_2} \text{sgn}(\psi_i(a)) = \prod_{i \in R_0} \text{sgn}(\psi_i(a)) = \text{sgn}(\psi_{R_0}(a)) = M_{C,Q}(a, R_0). \end{aligned}$$

As all the nonempty products of subsets of eigenvectors in  $Q$  are balanced on  $C$ ,  $M_{C,Q}(a, R_0)$  is positive for half the  $a \in C$  and negative for the other half. So,

$$M_{C,Q}(:, R_1)^T M_{C,Q}(:, R_2) = \sum_{a \in C} M_{C,Q}(a, R_1)M_{C,Q}(a, R_2) = \sum_{a \in C} M_{C,Q}(a, R_0) = 0.$$

□

**Lemma 8.9.2.** *If  $C$  is a balanced class of vertices and  $Q$  is a maximal set of eigenvectors that are independent on  $C$ , then for every  $a$  and  $b$  in  $C$  there is an  $i \in Q$  for which  $\psi_i(a) \neq \psi_i(b)$ .*

*Proof.* Assume by way of contradiction that this does not hold. There must be some eigenvector  $i$  for which  $\psi_i(a) \neq \psi_i(b)$ . We will show that if we added  $i$  to  $Q$ , the product of every subset would still be balanced. As we already know this for subsets of  $Q$ , we just have to prove it for subsets of the form  $R \cup \{i\}$ , where  $R \subseteq Q$ . As  $\psi_h(a) = \psi_h(b)$  for every  $h \in Q$ ,  $\psi_R(a) = \psi_R(b)$ . This implies  $\psi_{R \cup \{i\}}(a) \neq \psi_{R \cup \{i\}}(b)$ . Thus,  $\psi_{R \cup \{i\}}$  is not uniform on  $C$ , and so it must be balanced on  $C$ . □

**Lemma 8.9.3.** *If  $C$  is a balanced class of vertices and  $Q$  is a maximal set of eigenvectors that are independent on  $C$ , then the rows of  $M_{C,Q}$  are orthogonal.*

*Proof.* Let  $a$  and  $b$  be in  $C$ . From Lemma 8.9.2 we know that there is an  $i \in Q$  for which  $\psi_i(a) = -\psi_i(b)$ . To prove that the rows  $M_{C,Q}(a, :)$  and  $M_{C,Q}(b, :)$  are orthogonal, we compute their inner product:

$$\begin{aligned} \sum_{R \subseteq Q} \text{sgn}(\psi_R(a)\psi_R(b)) &= \sum_{R \subseteq Q - \{i\}} \text{sgn}(\psi_R(a)\psi_R(b)) + \text{sgn}(\psi_{R \cup \{i\}}(a)\psi_{R \cup \{i\}}(b)) \\ &= \sum_{R \subseteq Q - \{i\}} \text{sgn}(\psi_R(a)\psi_R(b)) + \text{sgn}(\psi_R(a)\psi_i(a)\psi_R(b)\psi_i(b)) \\ &= \sum_{R \subseteq Q - \{i\}} \text{sgn}(\psi_R(a)\psi_R(b)) + \text{sgn}(\psi_R(a)\psi_R(b))\text{sgn}(\psi_i(a)\psi_i(b)) \\ &= \sum_{R \subseteq Q - \{i\}} \text{sgn}(\psi_R(a)\psi_R(b)) - \text{sgn}(\psi_R(a)\psi_R(b)) \\ &= 0. \end{aligned}$$

□

**Corollary 8.9.4.** *Let  $C$  be a balanced subset of vertices. Then the size of  $C$  is a power of 2. If  $Q$  is an independent set of eigenvectors on  $C$ , then  $|Q| \leq \log_2 |C|$ .*

*Proof.* Let  $C$  be an orbit and let  $Q$  be a maximal set of eigenvectors that are independent on  $C$ . As the rows and columns of  $M_{C,Q}$  are both orthogonal,  $M_{C,Q}$  must be square. This implies that  $|C| = 2^{|Q|}$ . If we drop the assumption that  $Q$  is maximal, we still know that all the columns of  $M_{C,Q}$  are orthogonal. This matrix has  $2^{|Q|}$  columns. As they are vectors in  $|C|$  dimensions, there can be at most  $|C|$  of them.  $\square$

We can now describe the structure of a balanced subset of vertices  $C$ . We call a maximal set of eigenvectors that are independent on  $C$  a *base* for  $C$ . Every other eigenvector  $j$  is either constant on  $C$  or becomes constant when multiplied by the product of some subset  $R$  of eigenvectors in  $Q$ . In either case, we can write

$$\psi_j(a) = \gamma \prod_{i \in R} \psi_i(a) \quad \text{for all } a \in C, \quad (8.1)$$

for some constant  $\gamma$ .

Let  $\mathbf{v}_a(Q)$  denote the vector  $(\mathbf{v}_a(i))_{i \in Q}$ —the restriction of the vector  $\mathbf{v}_a$  to the coordinates in  $Q$ . I claim that every one of the  $2^{|Q|} \pm$  sign patterns of length  $|Q|$  must appear in exactly one of the vectors  $\mathbf{v}_q(Q)$ . The reason is that there are  $|C| = 2^{|Q|}$  of these vectors, and we established in Lemma 8.9.2 that  $\mathbf{v}_a(Q) \neq \mathbf{v}_b(Q)$  for all  $a \neq b$  in  $Q$ . Thus, for every diagonal  $\pm$  matrix  $\mathbf{S}_Q$  of dimension  $|Q|$ , we have

$$\{\mathbf{v}_a(Q)\mathbf{S}_Q : a \in C\} = \{\mathbf{v}_a(Q) : a \in C\}.$$

That is, this set of vectors is stabilized by  $\pm 1$  diagonal matrices.

As equation (8.1) gives a formula for the value taken on  $C$  by every eigenvector not in  $Q$  in terms of the eigenvectors in  $Q$ , we have described the structure of the subgroup of  $\mathbf{S}$  that stabilizes  $C$ : the diagonals corresponding to  $Q$  are unconstrained, and every other diagonal is some product of these. This structure is something that you are used to seeing in subspaces. Apply  $f$  to map this subgroup of  $\mathbf{S}$  to  $(\mathbf{Z}/2)^n$ , and let  $\mathbf{B}$  be a  $n$ -by- $\log_2(|C|)$  matrix containing a basis of the subspace in its columns. Any independent subset of  $\log_2(|C|)$  rows of  $\mathbf{B}$  will form a basis of the row-space, and is isomorphic to a base for  $C$  of the eigenvectors.

## 8.10 Algorithms

Let  $C_j$  be a balanced class. We just saw how to compute  $\mathcal{A}_j$ , assuming that we know  $C_j$  and a base  $Q$  for it. Of course, by “compute” we mean computing a basis of  $f(\mathcal{A}_j)$ . We now show how to find a base for a balanced class  $C_j$ . We do this by building up a set  $Q$  of eigenvectors that are independent on  $C_j$ . To do this, we go through the eigenvectors in order. For each eigenvector  $\psi_i$ , we must determine whether or not its values on  $C_j$  can be expressed as a product of eigenvectors already present in  $Q$ . If it can be, then we record this product as part of the structure of  $\mathcal{A}_j$ . If not, we add  $i$  to  $Q$ .

The eigenvector  $\psi_i$  is a product of eigenvectors in  $Q$  on  $C_j$  if and only if there is a constant  $\gamma$  and  $y_h \in \{0, 1\}$  for  $h \in Q$  such that

$$\psi_i(a) = \gamma \prod_{h \in Q} (\psi_h(a))^{y_h},$$

for all vertices  $a \in C_j$ . This happens if and only if

$$\text{sgn}(\psi_i(a)) = \prod_{h \in Q} \text{sgn}(\psi_h(a))^{y_h}.$$

We can tell whether or not these equations have a solution using linear algebra modulo 2. Let  $\mathbf{B}$  be the matrix over  $\mathbf{Z}/2$  such that

$$\psi_i(a) = (-1)^{\mathbf{B}(i,a)}.$$

Then, the above equations become

$$\mathbf{B}(i, a) = \sum_{h \in Q} y_h \mathbf{B}(h, a) \quad \text{for all } a \in C_j.$$

Thus, we can solve for the coefficients  $y_h$  in polynomial time, if they exist. If they do not, we add  $i$  to  $Q$ .

Once we have determined a base  $Q$  and how to express on  $C_j$  the values of every other eigenvector as a product of eigenvectors in  $Q$ , we have determine  $A_j$ .

It remains to explain how we partition the vertices into balanced classes. Consider applying the above procedure to a class  $C_j$  that is not balanced. We will discover that  $C_j$  is not balanced by finding a product of eigenvectors that is neither constant nor balanced on  $C_j$ . Every time we add an eigenvector  $\psi_i$  to  $Q$ , we will examine *every* product of vectors in  $Q$  to check if any are unbalanced on  $C_j$ . We can do this efficiently, because there are at most  $2^{|Q|} \leq |C_j|$  such products to consider. As we have added  $\psi_i$  to  $Q$ , none of the products of vectors in  $Q$  can be constant on  $C_j$ . If we find a product that it not balanced on  $C_j$ , then it must also be non-constant, and thus provide a way of splitting class  $C_j$  into two.

We can now summarize the entire algorithm. We first compute the partition by absolute values of entries described in section 8.7. We then go through the classes of the partition one-by-one. For each, we use the above procedure until we have either split it in two or we have determined that it is balanced and we have computed its automorphism group. If we do split the class in two, we refine the partition and start over. As the total number of times we split classes is at most  $n$ , this algorithm runs in polynomial time.

After we have computed a partition into balanced classes and have computer their automorphisms groups, we combine them to find the automorphisms group of the entire graph as described at the end of section 8.6.

## References

- [Bab81] László Babai. Moderately exponential bound for graph isomorphism. In *Fundamentals of Computation Theory*, number 117 in Lecture Notes in Math, pages 34–50. Springer-Verlag, Berlin-Heidelberg-New York, 1981.
- [Bab16] László Babai. Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 684–697. ACM, 2016.

- [BGM82] László Babai, D Yu Grigoryev, and David M Mount. Isomorphism of graphs with bounded eigenvalue multiplicity. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 310–324. ACM, 1982.
- [BL83] László Babai and Eugene M Luks. Canonical labeling of graphs. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pages 171–183. ACM, 1983.
- [LM82] F. Tom Leighton and Gary Miller. Certificates for graphs with distinct eigenvalues. Manuscript, 1982.
- [ZKT85] V. M. Zemlyachenko, N. M. Kornienko, and R. I. Tyshkevich. Graph isomorphism problem. *Journal of Soviet Mathematics*, 29:1426–1481, 1985.

## Testing Isomorphism of Strongly Regular Graphs

*Daniel A. Spielman*

September 26, 2018

## 9.1 Introduction

In the last lecture we saw how to test isomorphism of graphs in which every eigenvalue is distinct. So, in this lecture we will consider the opposite case: graphs that only have 3 distinct eigenvalues. These are the strongly regular graphs.

Our algorithm for testing isomorphism of these will not run in polynomial time. Rather, it takes time  $n^{O(n^{1/2} \log n)}$ . This is at least much faster than the naive algorithm of checking all  $n!$  possible permutations. In fact, this was the best known running time for general algorithms for graph isomorphism until three years ago.

## 9.2 Definitions

A graph  $G$  is strongly regular if

1. it is  $d$ -regular, for some integer  $d$ ;
2. there exists an integer  $\alpha$  such that for every pair of vertices  $x$  and  $y$  that are neighbors in  $G$ , there are exactly  $\alpha$  vertices  $z$  that are neighbors of *both*  $x$  and  $y$ ;
3. there exists an integer  $\beta$  such that for every pair of vertices  $x$  and  $y$  that are *not* neighbors in  $G$ , there are exactly  $\beta$  vertices  $z$  that are neighbors of *both*  $x$  and  $y$ .

These conditions are very strong, and it might not be obvious that there are any non-trivial graphs that satisfy these conditions. Of course, the complete graph and disjoint unions of complete graphs satisfy these conditions. Before proceeding, I warn you that there is a standard notation in the literature about strongly regular graphs, and I am trying not to use it. In this literature,  $d$  becomes  $k$ ,  $\alpha$  becomes  $\lambda$  and  $\beta$  becomes  $\mu$ . Many other letters are bound as well.

For the rest of this lecture, we will only consider strongly regular graphs that are connected and that are not the complete graph. I will now give you some examples.

### 9.3 Paley Graphs and The Pentagon

The Paley graphs we encountered are strongly regular. The simplest of these is the pentagon. It has parameters

$$n = 5, \quad d = 2, \quad \alpha = 0, \quad \beta = 1.$$

### 9.4 Lattice Graphs

For a positive integer  $n$ , the *lattice graph*  $L_n$  is the graph with vertex set  $\{1, \dots, n\}^2$  in which vertex  $(a, b)$  is connected to vertex  $(c, d)$  if  $a = c$  or  $b = d$ . Thus, the vertices may be arranged at the points in an  $n$ -by- $n$  grid, with vertices being connected if they lie in the same row or column. Alternatively, you can understand this graph as the product of two complete graphs on  $n$  vertices.

The parameters of this graph are:

$$d = 2(n - 1), \quad \alpha = n - 2, \quad \beta = 2.$$

### 9.5 Latin Square Graphs

A Latin square is an  $n$ -by- $n$  grid, each entry of which is a number between 1 and  $n$ , such that no number appears twice in any row or column. For example,

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 4 & 1 & 2 & 3 \\ 3 & 4 & 1 & 2 \\ 2 & 3 & 4 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}, \quad \text{and} \quad \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 1 & 3 \\ 3 & 1 & 4 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}$$

are Latin squares. Let me remark that the number of different Latin squares of size  $n$  grows very quickly—at least as fast as  $n!(n-1)!(n-2)!\dots 2!$ . Two Latin squares are said to be *isomorphic* if there is a renumbering of their rows, columns, and entries, or a permutation of these, that makes them the same. As this provides  $6(n!)^3$  isomorphisms, and this is much less than the number of Latin squares, there must be many non-isomorphic Latin squares of the same size. The two of the Latin squares above are isomorphic, but one is not.

From such a Latin square, we construct a Latin square graph. It will have  $n^2$  nodes, one for each cell in the square. Two nodes are joined by an edge if

1. they are in the same row,
2. they are in the same column, or
3. they hold the same number.

So, such a graph has degree  $d = 3(n - 1)$ . Any two nodes in the same row will both be neighbors with every other pair of nodes in their row. They will have two more common neighbors: the nodes in

their columns holding the other's number. So, they have  $n$  common neighbors. The same obviously holds for columns, and is easy to see for nodes that have the same number. So, every pair of nodes that are neighbors have exactly  $\alpha = n$  common neighbors.

On the other hand, consider two vertices that are not neighbors, say  $(1, 1)$  and  $(2, 2)$ . They lie in different rows, lie in different columns, and we are assuming that they hold different numbers. The vertex  $(1, 1)$  has two common neighbors of  $(2, 2)$  in its row: the vertex  $(1, 2)$  and the vertex holding the same number as  $(2, 2)$ . Similarly, it has two common neighbors of  $(2, 2)$  in its column. Finally, we can find two more common neighbors of  $(2, 2)$  that are in different rows and columns by looking at the nodes that hold the same number as  $(1, 1)$ , but which are in the same row or column as  $(2, 2)$ . So,  $\beta = 6$ .

## 9.6 The Eigenvalues of Strongly Regular Graphs

We will consider the adjacency matrices of strongly regular graphs. Let  $A$  be the adjacency matrix of a strongly regular graph with parameters  $(d, \alpha, \beta)$ . We already know that  $A$  has an eigenvalue of  $d$  with multiplicity 1. We will now show that  $A$  has just two other eigenvalues.

To prove this, first observe that the  $(a, b)$  entry of  $A^2$  is the number of common neighbors of vertices  $a$  and  $b$ . For  $a = b$ , this is just the degree of vertex  $a$ . We will use this fact to write  $A^2$  as a linear combination of  $A$ ,  $I$  and  $J$ , the all 1s matrix. To this end, observe that the adjacency matrix of the complement of  $A$  (the graph with non-edges where  $A$  has edges) is  $J - I - A$ . So,

$$A^2 = \alpha A + \beta(J - I - A) + dI = (\alpha - \beta)A + \beta J + (d - \beta)I.$$

For every vector  $\mathbf{v}$  orthogonal to  $\mathbf{1}$ ,

$$A^2\mathbf{v} = (\alpha - \beta)A\mathbf{v} + (d - \beta)\mathbf{v}.$$

So, every eigenvalue  $\lambda$  of  $A$  other than  $d$  satisfies

$$\lambda^2 = (\alpha - \beta)\lambda + d - \beta.$$

Thus, these are given by

$$\lambda = \frac{\alpha - \beta \pm \sqrt{(\alpha - \beta)^2 + 4(d - \beta)}}{2}.$$

These eigenvalues are traditionally denoted  $r$  and  $s$ , with  $r > s$ . By convention, the multiplicity of the eigenvalue  $r$  is always denoted  $f$ , and the multiplicity of  $s$  is always denoted  $g$ .

For example, for the pentagon we have

$$r = \frac{\sqrt{5} - 1}{2}, \quad s = -\frac{\sqrt{5} + 1}{2}.$$

For the lattice graph  $L_n$ , we have

$$r = n - 2, \quad s = -2.$$

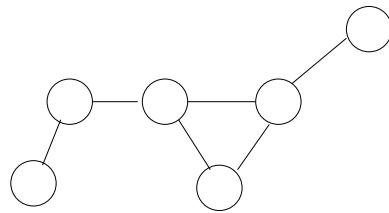
For the Latin square graphs of order  $n$ , we have

$$r = n - 3, \quad s = -3.$$

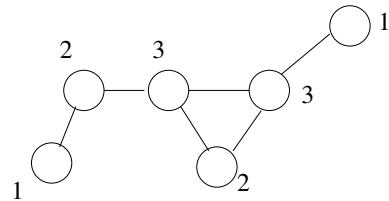
One can prove that every connected regular graph whose adjacency (or Laplacian) matrix has just three distinct eigenvalues is a strongly regular graph.

## 9.7 Testing Isomorphism by Individualization and Refinement

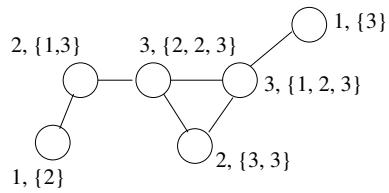
The problem of testing isomorphism of graphs is often reduced to the problem of giving each vertex in a graph a unique name. If we have a way of doing this that does not depend upon the initial ordering of the vertices, then we can use it to test graph isomorphism: find the unique names of vertices in both graphs, and then see if it provides an isomorphism. For example, consider the graph below.



We could begin by labeling every vertex by its degree.



The degrees distinguish between many nodes, but not all of them. We may refine this labeling by appending the labels of every neighbor of a node.



Now, every vertex has its own unique label. If we were given another copy of this graph, we could use these labels to determine the isomorphism between them. This procedure is called *refinement*, and it can be carried out until it stops producing new labels. However, it is clear that this procedure

will fail to produce unique labels if the graph has automorphisms, or if it is a regular graph. In these cases, we need a way to break symmetry.

The procedure called *individualization* breaks symmetry arbitrarily. It chooses some nodes in the graph, arbitrarily, to give their own unique names. Ideally, we pick one vertex to give a unique name, and then refine the resulting labeling. We could then pick another troubling vertex, and continue. We call a set of vertices  $S \subset V$  a *distinguishing set* if individualizing this set of nodes results in a unique name for every vertex, after refinement. How would we use a distinguishing set to test isomorphism? Assume that  $S$  is a distinguishing set for  $G = (V, E)$ . To test if  $H = (W, F)$  is isomorphic to  $G$ , we could enumerate over *every* possible set of  $|S|$  vertices of  $W$ , and check if they are a distinguishing set for  $H$ . If  $G$  and  $H$  are isomorphic, then  $H$  will also have an isomorphic distinguishing set that we can use to find an isomorphism between  $G$  and  $H$ . We would have to check  $\binom{n}{|S|}$  sets, and try  $|S|!$  labelings for each, so we had better hope that  $S$  is small.

## 9.8 Distinguishing Sets for Strongly Regular Graphs

We will now prove a result of Babai [Bab80] which says that every strongly regular graph has a distinguishing set of size  $O(\sqrt{n} \log n)$ . Babai's result won't require any refinement beyond naming every vertex by the set of individualized nodes that are its neighbors. So, we will prove that a set of nodes  $S$  is a distinguishing set by proving that for every pair of distinct vertices  $a$  and  $b$ , either there is an  $s \in S$  that is a neighbor of  $a$  but not of  $b$ , or the other way around. This will suffice to distinguish  $a$  and  $b$ . As our algorithm will work in a brute-force fashion, enumerating over all sets of a given size, we merely need to show that such a set  $S$  exists. We will do so by proving that a random set of vertices probably works.

I first observe that it suffices to consider strongly-regular graphs with  $d < n/2$ , as the complement of a strongly regular graph is also a strongly regular graph (that would have been too easy to assign as a homework problem). We should also observe that every strongly-regular graph has diameter 2, and so  $d \geq \sqrt{n-1}$ .

**Lemma 9.8.1.** *Let  $G = (V, E)$  be a connected strongly regular graph with  $n$  vertices and degree  $d < n/2$ . Then for every pair of vertices  $a$  and  $b$ , there are at least  $d/3$  vertices that are neighbors of  $a$  but not  $b$ .*

Before I prove this, let me show how we may use it to prove the theorem. This lemma tells us that there are at least  $\sqrt{n-1}/3$  nodes that are neighbors of  $a$  but not of  $b$ . Let  $T$  be the set of nodes that are neighbors of  $a$  but not neighbors of  $b$ . So, if we choose a vertex at random, the probability that it is in  $T$  is at least

$$\frac{|T|}{n} \geq \frac{\sqrt{n-1}}{3n} \geq \frac{1}{3\sqrt{n+2}}.$$

If we choose a set  $S$  of  $3\sqrt{n+2} \ln n^2$  vertices at random, the probability that none of them is in  $T$  is

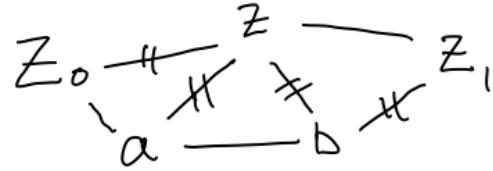
$$\left(1 - \frac{1}{3\sqrt{n+2}}\right)^{3\sqrt{n+2} \ln n^2} \leq \frac{1}{n^2}.$$

So, the probability that a random set of this many nodes fails to distinguish all  $\binom{n}{2}$  pairs is at most  $1/2$ .

*Proof of Lemma 9.8.1.* Write  $a \sim b$  if  $a$  is a neighbor of  $b$ , and  $a \not\sim b$  otherwise. If  $a \sim b$ , then the number of nodes that are neighbors of  $a$  but not of  $b$  is  $d - 1 - \alpha$ , and if  $a \not\sim b$  the number is  $d - \beta$ . So, we need to prove that neither  $\alpha$  nor  $\beta$  is too close to  $d$ .

We will do this by establishing some elementary relations between these parameters. First, consider the case in which  $a \sim b$ . Let  $z$  be any vertex such that  $a \not\sim z$  and  $b \not\sim z$ . We will use  $z$  to prove an upper bound on the number of vertices  $w$  that are neighbors of  $a$  but not of  $b$  (I know this looks like the wrong direction, but be patient). Let

$$Z_0 = \{w : w \sim a, w \not\sim z\}, \quad \text{and} \quad Z_1 = \{w : w \not\sim b, w \sim z\}.$$



Clearly, every  $w$  that is a neighbor of  $a$  but not of  $b$  lies in either  $Z_0$  or  $Z_1$ . As  $z$  is neither a neighbor of  $a$  nor of  $b$ ,

$$|Z_0| = |Z_1| = d - \beta.$$

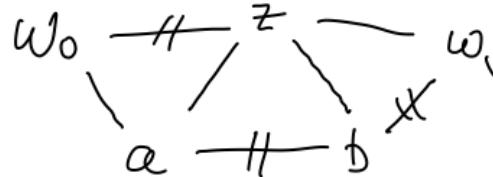
So,

$$d - \alpha - 1 \leq 2(d - \beta) \implies 2\beta \leq d + \alpha + 1. \quad (9.1)$$

So, if  $\beta$  is close to  $d$ ,  $\alpha$  must also be close to  $d$ .

We will obtain an inequality in the other direction when  $a \not\sim b$  by exploiting a  $z$  such that  $z \sim a$  and  $z \sim b$ . Now, for any  $w \sim a$  but  $w \not\sim b$ , we have either

$$(w \sim a \text{ and } w \not\sim z) \text{ or } (w \sim z \text{ and } w \not\sim b).$$

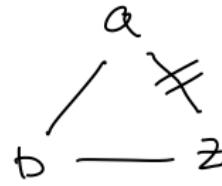


So,

$$d - \beta \leq 2(d - \alpha - 1) \implies 2(\alpha + 1) \leq d + \beta. \quad (9.2)$$

This tells us that if  $\alpha$  is close to  $d$ , then  $\beta$  is also.

We require one more relation between  $\alpha$  and  $\beta$ . We obtain this relation by picking any vertex  $a$ , and counting the pairs  $b, z$  such that  $b \sim z$ ,  $a \sim b$  and  $a \not\sim z$ .



Every node  $b$  that is a neighbor of  $a$  has  $\alpha$  neighbors in common with  $a$ , and so has  $d - \alpha - 1$  neighbors that are not neighbors of  $a$ . This gives

$$|\{(b, z) : b \sim z, a \sim b, a \not\sim z\}| = d(d - \alpha - 1).$$

On the other hand, there are  $n - d - 1$  nodes  $z$  that are not neighbors of  $a$ , and each of them has  $\beta$  neighbors in common with  $a$ , giving

$$|\{(b, z) : b \sim z, a \sim b, a \not\sim z\}| = (n - d - 1)\beta.$$

Combining, we find

$$(n - d - 1)\beta = d(d - \alpha - 1). \quad (9.3)$$

As  $d < n/2$ , this equation tells us

$$d(d - \alpha - 1) \geq d\beta \implies d - \alpha - 1 \geq \beta. \quad (9.4)$$

Adding inequality 9.1 to (9.4) gives

$$2d \geq 3\beta \implies \beta \leq \frac{2}{3}d,$$

while adding inequality 9.2 to (9.4) gives

$$\alpha + 1 \leq \frac{2}{3}d.$$

Thus, for every  $a \neq b$  the number of vertices that are neighbors of  $a$  but not of  $b$  is at least  $\min(d - \alpha - 1, d - \beta) \geq d/3$ .

□

## 9.9 Notes

You should wonder if we can make this faster by analyzing refinement steps. In, [Spi96], I improved the running time bound to  $2^{O(n^{1/3} \log n)}$  by analyzing two refinement phases. The algorithm required us to handle certain special families of strongly regular graphs separately: Latin square graphs and Steiner graphs. Algorithms for testing isomorphism of strongly regular graphs were recently improved by Babai, Chen, Sun, Teng, and Wilmes [BCS<sup>+</sup>13, BW13, SW15]. The running times of all these algorithms are subsumed by that in Babai's breakthrough algorithm for testing graph isomorphism [Bab16].

## References

- [Bab80] László Babai. On the complexity of canonical labeling of strongly regular graphs. *SIAM Journal on Computing*, 9(1):212–216, 1980.
- [Bab16] László Babai. Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 684–697. ACM, 2016.
- [BCS<sup>+</sup>13] László Babai, Xi Chen, Xiaorui Sun, Shang-Hua Teng, and John Wilmes. Faster canonical forms for strongly regular graphs. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 157–166. IEEE, 2013.
- [BW13] László Babai and John Wilmes. Quasipolynomial-time canonical form for steiner designs. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 261–270. ACM, 2013.
- [Spi96] Daniel A. Spielman. Faster isomorphism testing of strongly regular graphs. In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 576–584, New York, NY, USA, 1996. ACM.
- [SW15] Xiaorui Sun and John Wilmes. Faster canonical forms for primitive coherent configurations. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 693–702. ACM, 2015.

## Random Walks on Graphs

*Daniel A. Spielman*

October 1, 2018

## 10.1 Overview

We will examine how the eigenvalues of a graph govern the convergence of a random walk on the graph.

## 10.2 Random Walks

In this lecture, we will consider random walks on undirected graphs. Let's begin with the definitions. Let  $G = (V, E, w)$  be a weighted undirected graph. A random walk on a graph is a process that begins at some vertex, and at each time step moves to another vertex. When the graph is unweighted, the vertex the walk moves to is chosen uniformly at random among the neighbors of the present vertex. When the graph is weighted, it moves to a neighbor with probability proportional to the weight of the corresponding edge. While the transcript (the list of vertices in the order they are visited) of a particular random walk is sometimes of interest, it is often more productive to reason about the expected behavior of a random walk. To this end, we will investigate the probability distribution over vertices after a certain number of steps.

We will let the vector  $\mathbf{p}_t \in \mathbb{R}^V$  denote the probability distribution at time  $t$ . I will write  $\mathbf{p}_t(a)$  to indicate the value of  $\mathbf{p}_t$  at a vertex  $a$ —that is the probability of being at vertex  $a$  at time  $t$ . A probability vector  $\mathbf{p}$  is a vector such that  $\mathbf{p}(a) \geq 0$ , for all  $a \in V$ , and

$$\sum_u \mathbf{p}(a) = 1.$$

Our initial probability distribution,  $\mathbf{p}_0$ , will typically be concentrated one vertex. That is, there will be some vertex  $a$  for which  $\mathbf{p}_0(a) = 1$ . In this case, we say that the walk starts at  $a$ .

To derive a  $\mathbf{p}_{t+1}$  from  $\mathbf{p}_t$ , note that the probability of being at a vertex  $a$  at time  $t + 1$  is the sum over the neighbors  $b$  of  $a$  of the probability that the walk was at  $b$  at time  $t$ , times the probability it moved from  $b$  to  $a$  in time  $t + 1$ . We can state this algebraically as

$$\mathbf{p}_{t+1}(a) = \sum_{b:(a,b) \in E} \frac{w(a, b)}{\mathbf{d}(b)} \mathbf{p}_t(b), \quad (10.1)$$

where  $\mathbf{d}(b) = \sum_a w(a, b)$  is the weighted degree of vertex  $b$ .

We will often consider lazy random walks, which are the variant of random walks that stay put with probability 1/2 at each time step, and walk to a random neighbor the other half of the time. These evolve according to the equation

$$\mathbf{p}_{t+1}(a) = (1/2)\mathbf{p}_t(a) + (1/2) \sum_{b:(a,b) \in E} \frac{w(a,b)}{d(b)} \mathbf{p}_t(b). \quad (10.2)$$

### 10.3 Diffusion

There are a few types of diffusion that people study in a graph, but the most common is closely related to random walks. In a diffusion process, we imagine that we have some substance that can occupy the vertices, such as a gas or fluid. At each time step, some of the substance diffuses out of each vertex. If we say that half the substance stays at a vertex at each time step, and the other half is distributed among its neighboring vertices, then the distribution of the substance will evolve according to equation (10.2). That is, probability mass obeys this diffusion equation.

I remark that often people consider finer time steps in which smaller fractions of the mass leave the vertices. In the limit, this results in continuous random walks that are modeled by the matrix exponential. These are in many ways more natural than discrete time random walks. But, I do not think we will discuss them in this course.

### 10.4 Matrix form

The right way to understand the behavior of random walks is through linear algebra.

Equation (10.2) is equivalent to:

$$\mathbf{p}_{t+1} = (1/2) (\mathbf{I} + \mathbf{M} \mathbf{D}^{-1}) \mathbf{p}_t. \quad (10.3)$$

You can verify this by checking that it is correct for any entry  $\mathbf{p}_{t+1}(u)$ , and you should do this yourself. It will prevent much confusion later.

For the rest of the course, I will let  $\mathbf{W}_G$  denote the *lazy walk matrix* of the graph  $G$ , where

$$\mathbf{W}_G \stackrel{\text{def}}{=} (1/2) (\mathbf{I} + \mathbf{M}_G \mathbf{D}_G^{-1}). \quad (10.4)$$

This is the one asymmetric matrix that we will deal with in this course. It is related to the *normalized Laplacian*, which is symmetric and which is defined by

$$\mathbf{N} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{M} \mathbf{D}^{-1/2}.$$

Thus, the normalized Laplacian is positive semidefinite, and has the same rank as the ordinary (sometimes called “combinatorial”) Laplacian. There are many advantages of working with the normalized Laplacian: the mean of its eigenvalues is 1, so they are always on a degree-independent

scale. One can prove that  $\nu_n \leq 2$ , with equality if and only if the graph is bipartite. I recommend proving  $\nu_n \leq 2$  by showing that

$$\mathbf{L} \succ -\mathbf{M},$$

which follows from consideration of the quadratic form  $\sum_{(a,b) \in E} w_{a,b}(\mathbf{x}(a) - \mathbf{x}(b))^2$ .

The walk matrix is similar to a symmetric matrix that is related to the normalized Laplacian:

$$\begin{aligned} W &= \mathbf{I} - \frac{1}{2} (\mathbf{I} - \mathbf{MD}^{-1}) \\ &= \mathbf{I} - \frac{1}{2} \mathbf{D}^{1/2} \left( \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{M} \mathbf{D}^{-1/2} \right) \mathbf{D}^{-1/2} \\ &= \mathbf{I} - \frac{1}{2} \mathbf{D}^{1/2} \mathbf{ND}^{-1/2}, \end{aligned}$$

So, we know that  $\mathbf{W}$  is diagonalizable, and that for every eigenvector  $\psi_i$  of  $\mathbf{N}$  with eigenvalue  $\nu_i$ , the vector  $\mathbf{D}^{1/2}\psi_i$  is a right-eigenvector of  $\mathbf{W}$  of eigenvalue  $1 - \nu_i/2$ :

$$\begin{aligned} \mathbf{W} \left( \mathbf{D}^{1/2} \psi_i \right) &= \left( \mathbf{I} - \frac{1}{2} \mathbf{D}^{1/2} \mathbf{ND}^{-1/2} \right) \mathbf{D}^{1/2} \psi_i \\ &= \mathbf{D}^{1/2} \psi_i - \frac{1}{2} \mathbf{D}^{1/2} \mathbf{N} \psi_i \\ &= \mathbf{D}^{1/2} \psi_i - \frac{\nu_i}{2} \mathbf{D}^{1/2} \psi_i \\ &= (1 - \nu_i/2) \mathbf{D}^{1/2} \psi_i. \end{aligned}$$

The key thing to remember in the asymmetric case is that the eigenvectors of  $\mathbf{W}$  are not necessarily orthogonal.

You may be wonder why I have decided to consider only lazy walks, rather than the more natural walk given by  $\mathbf{MD}^{-1}$ . There are two equivalent reasons. The first is that all the eigenvalues of  $\mathbf{W}$  are between 1 and 0. The second reason is explained in the next section.

For the rest of the semester, we will let the eigenvalues of  $\mathbf{W}$  be:

$$1 = \omega_1 \geq \omega_2 \geq \dots \geq \omega_n \geq 0, \quad \text{where } \omega_i = (1 - \nu_i/2).$$

Yes, I know that  $\omega$  is not a greek equivalent of  $w$ , but it sure looks like it.

## 10.5 The stable distribution

Regardless of the starting distribution, the lazy random walk on a connected graph always converges to one distribution: the *stable distribution*. This is the other reason that we forced our random walk to be lazy. Without laziness<sup>1</sup>, there can be graphs on which the random walks never converge. For example, consider a non-lazy random walk on a bipartite graph. Every-other step will bring it to

---

<sup>1</sup>Strictly speaking, any nonzero probability of staying put at any vertex in a connected graph will guarantee convergence. We don't really need a half probability at every vertex.

the other side of the graph. So, if the walk starts on one side of the graph, its limiting distribution at time  $t$  will depend upon the parity of  $t$ .

In the stable distribution, every vertex is visited with probability proportional to its weighted degree. We denote the vector encoding this distribution by  $\pi$ , where

$$\pi = \mathbf{d}/(\mathbf{1}^T \mathbf{d})$$

and we recall that  $\mathbf{d}$  is the vector of degrees. We can see that  $\pi$  is a right-eigenvector of  $\mathbf{W}$  of eigenvalue 1:

$$\mathbf{M}\mathbf{D}^{-1}\pi = \mathbf{M}\mathbf{D}^{-1} \mathbf{d}/(\mathbf{1}^T \mathbf{d}) = \mathbf{M}\mathbf{1}/(\mathbf{1}^T \mathbf{d}) = \mathbf{d}/(\mathbf{1}^T \mathbf{d}) = \pi,$$

so

$$\mathbf{W}\pi = (1/2)\mathbf{I}\pi + (1/2)\mathbf{M}\mathbf{D}^{-1}\pi = (1/2)\pi + (1/2)\pi = \pi.$$

This agrees with the translation we have established between eigenvectors of  $\mathbf{W}$  and eigenvectors of  $\mathbf{N}$ . For a connected graph the nullspace of  $\mathbf{N}$  is spanned by  $\mathbf{d}^{1/2}$ , and  $\pi$  is a multiple of  $\mathbf{D}^{1/2}\mathbf{d}^{1/2}$ .

To see that the walk converges to  $\pi$ , we expand  $\mathbf{D}^{-1/2}$  times the initial distribution in the eigenvectors  $\psi_1, \dots, \psi_n$  of  $\mathbf{N}$ . Let

$$\mathbf{D}^{-1/2}\mathbf{p}_0 = \sum_i c_i \psi_i.$$

Note that

$$c_1 = \psi_1^T (\mathbf{D}^{-1/2}\mathbf{p}_0) = \frac{(\mathbf{d}^{1/2})^T}{\|\mathbf{d}^{1/2}\|} (\mathbf{D}^{-1/2}\mathbf{p}_0) = \frac{\mathbf{1}^T \mathbf{p}_0}{\|\mathbf{d}^{1/2}\|} = \frac{1}{\|\mathbf{d}^{1/2}\|},$$

as  $\mathbf{p}_0$  is a probability vector. We have

$$\begin{aligned} \mathbf{p}_t &= \mathbf{W}^t \mathbf{p}_0 \\ &= (\mathbf{D}^{1/2}(\mathbf{I} - \mathbf{N}/2)\mathbf{D}^{-1/2})^t \mathbf{p}_0 \\ &= (\mathbf{D}^{1/2}(\mathbf{I} - \mathbf{N}/2)^t \mathbf{D}^{-1/2}) \mathbf{p}_0 \\ &= \mathbf{D}^{1/2}(\mathbf{I} - \mathbf{N}/2)^t \sum_i c_i \psi_i \\ &= \mathbf{D}^{1/2} \sum_i (1 - \nu_i/2)^t c_i \psi_i \\ &= \mathbf{D}^{1/2} c_1 \psi_1 + \mathbf{D}^{1/2} \sum_{i \geq 2} (1 - \nu_i/2)^t c_i \psi_i. \end{aligned}$$

As  $0 < \nu_i \leq 2$  for  $i \geq 2$ , the right-hand term must go to zero. On the other hand,  $\psi_1 = \mathbf{d}^{1/2}/\|\mathbf{d}^{1/2}\|$ , so

$$\mathbf{D}^{1/2} c_1 \psi_1 = \mathbf{D}^{1/2} \left( \frac{1}{\|\mathbf{d}^{1/2}\|} \right) \frac{\mathbf{d}^{1/2}}{\|\mathbf{d}^{1/2}\|} = \frac{\mathbf{d}}{\|\mathbf{d}^{1/2}\|^2} = \frac{\mathbf{d}}{\sum_j \mathbf{d}(j)} = \pi.$$

This is a perfect example of one of the main uses of spectral theory: to understand what happens when we repeatedly apply an operator.

## 10.6 The Rate of Convergence

The rate of convergence of a lazy random walk to the stable distribution is dictated by  $\omega_2$ . There are many ways of saying this. We will do so point-wise. Assume that the random walk starts at some vertex  $a \in V$ . For every vertex  $b$ , we will bound how far  $\mathbf{p}_t(b)$  can be from  $\boldsymbol{\pi}(b)$ .

**Theorem 10.6.1.** *For all  $a, b$  and  $t$ , if  $\mathbf{p}_0 = \boldsymbol{\delta}_a$ , then*

$$|\mathbf{p}_t(b) - \boldsymbol{\pi}(b)| \leq \sqrt{\frac{\mathbf{d}(b)}{\mathbf{d}(a)}} \omega_2^t.$$

*Proof.* Observe that

$$\mathbf{p}_t(b) = \boldsymbol{\delta}_b^T \mathbf{p}_t.$$

From the analysis in the previous section, we know

$$\mathbf{p}_t(b) = \boldsymbol{\delta}_b^T \mathbf{p} = \boldsymbol{\pi}(b) + \boldsymbol{\delta}_b^T \mathbf{D}^{1/2} \sum_{i \geq 2} \omega_i^t c_i \boldsymbol{\psi}_i.$$

We need merely prove an upper bound on the magnitude of the right-hand term. To this end, recall that

$$c_i = \boldsymbol{\psi}_i^T \mathbf{D}^{-1/2} \boldsymbol{\delta}_a.$$

So,

$$\boldsymbol{\delta}_b^T \mathbf{D}^{1/2} \sum_{i \geq 2} \omega_i^t c_i \boldsymbol{\psi}_i = \sqrt{\frac{\mathbf{d}(b)}{\mathbf{d}(a)}} \boldsymbol{\delta}_b^T \sum_{i \geq 2} \omega_i^t \boldsymbol{\psi}_i \boldsymbol{\psi}_i^T \boldsymbol{\delta}_a.$$

Analyzing the right-hand part of this last expression, we find

$$\begin{aligned} \left| \boldsymbol{\delta}_b^T \sum_{i \geq 2} \omega_i^t \boldsymbol{\psi}_i \boldsymbol{\psi}_i^T \boldsymbol{\delta}_a \right| &= \left| \sum_{i \geq 2} \omega_i^t (\boldsymbol{\delta}_b^T \boldsymbol{\psi}_i) (\boldsymbol{\psi}_i^T \boldsymbol{\delta}_a) \right| \\ &\leq \sum_{i \geq 2} \omega_i^t |\boldsymbol{\delta}_b^T \boldsymbol{\psi}_i| |\boldsymbol{\psi}_i^T \boldsymbol{\delta}_a| \\ &\leq \omega_2^t \sum_{i \geq 2} |\boldsymbol{\delta}_b^T \boldsymbol{\psi}_i| |\boldsymbol{\psi}_i^T \boldsymbol{\delta}_a|. \end{aligned}$$

To prove an upper bound on this last term, let  $\boldsymbol{\Psi}$  be the matrix having the eigenvectors  $\boldsymbol{\psi}_1, \dots, \boldsymbol{\psi}_n$  in its columns. This is an orthonormal matrix, and so its rows must be orthonormal as well. Thus,

$$\begin{aligned} \sum_{i \geq 2} |\boldsymbol{\delta}_b^T \boldsymbol{\psi}_i| |\boldsymbol{\psi}_i^T \boldsymbol{\delta}_a| &\leq \sum_{i \geq 1} |\boldsymbol{\delta}_b^T \boldsymbol{\psi}_i| |\boldsymbol{\psi}_i^T \boldsymbol{\delta}_a| \\ &\leq \sqrt{\sum_{i \geq 1} (\boldsymbol{\delta}_b^T \boldsymbol{\psi}_i)^2} \sqrt{\sum_{i \geq 1} (\boldsymbol{\delta}_a^T \boldsymbol{\psi}_i)^2} \\ &= \|\boldsymbol{\Psi}(b, :)\| \|\boldsymbol{\Psi}(a, :)\| \\ &= 1. \end{aligned}$$

□

As  $\omega_2 = 1 - \nu_2/2$ , and

$$\omega_2^t = (1 - \nu_2/2)^t \approx e^{-t\nu_2/2},$$

we should expect random walks to converge once  $t$  reaches the order of  $(\log n)/\nu_2$  or  $1/\nu_2$ .

## 10.7 Examples

We should now do some examples. I'd like to understand each in two ways: by examining  $\nu_2$  for each graph and by thinking about how a random walk on each graph should behave. While we have explicitly worked out  $\lambda_2$  for many graphs, we have not done this for  $\nu_2$ . The following lemma will allow us to relate bounds on  $\lambda_2$  into bounds on  $\nu_2$ :

**Lemma 10.7.1.** *Let  $\mathbf{L}$  be the Laplacian matrix of a graph, with eigenvalues  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ , and let  $\mathbf{N}$  be its normalized Laplacian, with eigenvalues  $\nu_1 \leq \nu_2 \leq \dots \leq \nu_n$ . Then, for all  $i$*

$$\frac{\lambda_i}{d_{\min}} \geq \nu_i \geq \frac{\lambda_i}{d_{\max}},$$

where  $d_{\min}$  and  $d_{\max}$  are the minimum and maximum degrees of vertices in the graph.

*Proof.* The Courant-Fischer theorem tells us that

$$\nu_i = \min_{\dim(S)=i} \max_{\mathbf{x} \in S} \frac{\mathbf{x}^T \mathbf{N} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

As the change of variables  $\mathbf{y} = \mathbf{D}^{-1/2} \mathbf{x}$  is non-singular, this equals

$$\min_{\dim(T)=i} \max_{\mathbf{y} \in T} \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}}.$$

So,

$$\min_{\dim(T)=i} \max_{\mathbf{y} \in T} \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}} \geq \min_{\dim(T)=i} \max_{\mathbf{y} \in T} \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{d_{\max} \mathbf{y}^T \mathbf{y}} = \frac{1}{d_{\max}} \min_{\dim(T)=i} \max_{\mathbf{y} \in T} \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{y}} = \frac{\lambda_i}{d_{\max}}.$$

The other bound may be proved similarly.  $\square$

### 10.7.1 The Path

As every vertex in the path on  $n$  vertices has degree 1 or 2,  $\nu_2$  is approximately  $\lambda_2$ , which is approximately  $c/n^2$  for some constant  $c$ .

To understand the random walk on the path, think about what happens when the walk starts in the middle. Ignoring the steps on which it stays put, it will either move to the left or the right with probability 1/2. So, the position of the walk after  $t$  steps is distributed as the sum of  $t$  random variables taking values in  $\{1, -1\}$ . Recall that the standard deviation of such a sum is  $\sqrt{t}$ . So, we need to have  $\sqrt{t}$  comparable to  $n/4$  for there to be a reasonable chance that the walk is on the left or right  $n/4$  vertices.

### 10.7.2 The Complete Binary Tree

As with the path,  $\nu_2$  for the tree is within a constant of  $\lambda_2$  for the tree, and so is approximately  $c/n$  for some constant  $c$ . To understand the random walk on  $T_n$ , first note that whenever it is at a vertex, it is twice as likely to step towards a leaf as it is to step towards the root. So, if the walk starts at a leaf, there is no way the walk can mix until it reaches the root. The height of the walk is like a sum of  $\pm 1$  random variables, except that they are twice as likely to be  $-1$  as they are to be  $1$ , and that their sum never goes below 0. One can show that we need to wait approximately  $n$  steps before such a walk will hit the root. Once it does hit the root, the walk mixes rapidly.

### 10.7.3 The Dumbbell

Now, let's consider another one of my favorite graphs, the dumbbell. The dumbbell graph  $D_n$  consists of two complete graphs on  $n$  vertices, joined by one edge. So, there are  $2n$  vertices in total. The isoperimetric number of this graph is

$$\theta_{D_n} \sim \frac{1}{n}.$$

Using the test vector that is 1 on one complete graph and  $-1$  on the other, we can show that

$$\lambda_2(D_n) \lesssim 1/n.$$

Lemma 10.7.1 then tells us that

$$\nu_2(D_n) \lesssim 1/n^2.$$

To prove that this bound is almost tight, we use the following lemma.

**Lemma 10.7.2.** *Let  $G$  be an unweighted graph of diameter at most  $r$ . Then,*

$$\lambda_2(G) \geq \frac{2}{r(n-1)}.$$

*Proof.* For every pair of vertices  $(a, b)$ , let  $P(a, b)$  be a path in  $G$  of length at most  $r$ . We have

$$\mathbf{L}_{(a,b)} \preccurlyeq r \cdot \mathbf{L}_{P(a,b)} \preccurlyeq r \mathbf{L}_G.$$

So,

$$K_n \preccurlyeq r \binom{n}{2} G,$$

and

$$n \leq r \binom{n}{2} \lambda_2(G),$$

from which the lemma follows. □

The diameter of  $D_n$  is 3, so we have  $\lambda_2(D_n) \geq 2/3(n - 1)$ . As every vertex of  $D_n$  has degree at least  $n - 1$ , we may conclude  $\nu_2(D_n) \gtrapprox 2/3(n - 1)^2$ .

To understand the random walk on this graph, consider starting it at some vertex that is not attached to the bridge edge. After the first step the walk will be well mixed on the vertices in the side on which it starts. Because of this, the chance that it finds the edge going to the other side is only around  $1/n^2$ : there is only a  $1/n$  chance of being at the vertex attached to the bridge edge, and only a  $1/n$  chance of choosing that edge when at that vertex. So, we must wait some multiple of  $n^2$  steps before there is a reasonable chance that the walk reaches the other side of the graph.

#### 10.7.4 The Bolas Graph

I define the bolas<sup>2</sup> graph  $B_n$  to be a graph containing two  $n$ -cliques connected by a path of length  $n$ . The bolas graph has a value of  $\nu_2$  that is almost as small as possible. Equivalently, random walks on a bolas graph mix almost as slowly as possible.

The analysis of the random walk on a bolas is similar to that on a dumbbell, except that when the walk is on the first vertex of the path the chance that it gets to the other end before moving back to the clique at which we started is only  $1/n$ . So, we must wait around  $n^3$  steps before there is a reasonable chance of getting to the other side.

Next lecture, we will learn that we can upper bound  $\nu_2$  with a test vector using the fact that

$$\nu_2 = \min_{\mathbf{x} \perp \mathbf{d}} \frac{\mathbf{x}^T \mathbf{L} \mathbf{x}}{\mathbf{x}^T \mathbf{D} \mathbf{x}}.$$

To prove an upper bound on  $\nu_2$ , form a test vector that is  $n/2$  on one clique,  $-n/2$  on the other, and increases by 1 along the path. We can use the symmetry of the construction to show that this vector is orthogonal to  $\mathbf{d}$ . The numerator of the generalized Rayleigh quotient is  $n$ , and the denominator is the sum of the squares of the entries of the vectors times the degrees of the vertices, which is some constant times  $n^4$ . This tells us that  $\nu_2$  is at most some constant over  $n^3$ .

To see that  $\nu_2$  must be at least some constant over  $n^3$ , and in fact that this must hold for every graph, apply Lemmas 10.7.1 and 10.7.2.

### 10.8 Final Notes

The procedure we have described—repeatedly multiplying a vector by  $\mathbf{W}$  and showing that the result approximates  $\boldsymbol{\pi}$ —is known in Numerical Linear Algebra as the power method. It is one of the common ways of approximately computing eigenvectors.

In the proof of Theorem 10.6.1, we were a little loose with some of the terms. The slack comes from two sources. First, we upper bounded  $\omega_i$  by  $\omega_2$  for all  $i$ , while many of the  $\omega_i$  are probably significantly less than  $\omega_2$ . This phenomenon is often called “eigenvalue decay”, and it holds in many graphs. This sloppiness essentially costs us a multiplicative factor of  $\log n$  in the number of

---

<sup>2</sup>A bolas is a hunting weapon consisting of two balls or rocks tied together with a cord.

steps  $t$  we need to achieve the claimed bound. You will note that in the examples above, the time to approximate convergence is typically on the order of  $1/\nu_2$ , not  $(\log n)/\nu_2$ . This is because of eigenvalue decay.

The second source of slack appeared when we upper bounded the absolute value of a sum by the sum of the absolute value. I am not sure how much this cost us.

## Conductance, the Normalized Laplacian, and Cheeger's Inequality

*Daniel A. Spielman*

October 3, 2018

## 11.1 Overview

As the title suggests, in this lecture I will introduce conductance, a measure of the quality of a cut, and the normalized Laplacian matrix of a graph. I will then prove Cheeger's inequality, which relates the second-smallest eigenvalue of the normalized Laplacian to the conductance of a graph.

Cheeger [Che70] first proved his famous inequality for manifolds. Many discrete versions of Cheeger's inequality were proved in the late 80's [SJ89, LS88, AM85, Alo86, Dod84, Var85]. Some of these consider the walk matrix instead of the normalized Laplacian, and some consider the isoperimetric ratio instead of conductance.

The proof that I present today follows an approach developed by Luca Trevisan [Tre11]. For simplicity, we do the proof in the unweighted case, and indicate how to extend it to weighted graphs at the end of the notes.

## 11.2 Conductance

Back in Lecture 2, we related to isoperimetric ratio of a subset of the vertices to the second eigenvalue of the Laplacian. We proved that for every  $S \subset V$

$$\theta(S) \geq \lambda_2(1 - s),$$

where  $s = |S| / |V|$  and

$$\theta(S) \stackrel{\text{def}}{=} \frac{|\partial(S)|}{|S|}.$$

Re-arranging terms slightly, this can be stated as

$$|V| \frac{|\partial(S)|}{|S| |V - S|} \geq \lambda_2.$$

Cheeger's inequality provides a relation in the other direction. However, the relation is tighter and cleaner when we look at two slightly different quantities: the *conductance* of the set and the second eigenvalue of the normalized Laplacian.

The formula for conductance has a different denominator that depends upon the sum of the degrees of the vertices in  $S$ . I will write  $d(S)$  for the sum of the degrees of the vertices in  $S$ . Thus,  $d(V)$  is

twice the number of edges in the graph. We define the conductance of  $S$  to be

$$\phi(S) \stackrel{\text{def}}{=} \frac{|\partial(S)|}{\min(d(S), d(V - S))}.$$

Note that many similar, although sometimes slightly different, definitions appear in the literature. For example, we would instead use

$$\frac{d(V)\partial(S)}{d(S)d(V - S)},$$

which appears below in (11.5).

We define the conductance of a graph  $G$  to be

$$\phi_G \stackrel{\text{def}}{=} \min_{S \subset V} \phi(S).$$

The conductance of a graph is more useful in many applications than the isoperimetric number. I usually find that conductance is the more useful quantity when you are concerned about edges, and that isoperimetric ratio is most useful when you are concerned about vertices. Conductance is particularly useful when studying random walks in graphs.

### 11.3 The Normalized Laplacian

It seems natural to try to relate the conductance to the following generalized Rayleigh quotient:

$$\frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}}. \quad (11.1)$$

If we make the change of variables

$$\mathbf{D}^{1/2} \mathbf{y} = \mathbf{x},$$

then this ratio becomes

$$\frac{\mathbf{x}^T \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

That is an ordinary Rayleigh quotient, which we understand a little better. The matrix in the middle is called the normalized Laplacian (see [Chu97]). We reserve the letter  $\mathbf{N}$  for this matrix:

$$\mathbf{N} \stackrel{\text{def}}{=} \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2}.$$

This matrix often proves more useful when examining graphs in which nodes have different degrees. We will let  $0 = \nu_1 \leq \nu_2 \leq \dots \leq \nu_n$  denote the eigenvalues of  $\mathbf{N}$ .

The conductance is related to  $\nu_2$  as the isoperimetric number is related to  $\lambda_2$ :

$$\nu_2/2 \leq \phi_G. \quad (11.2)$$

I include a proof of this in the appendix.

My goal for today's lecture is to prove Cheeger's inequality,

$$\phi_G \leq \sqrt{2\nu_2},$$

which is much more interesting. In fact, it is my favorite theorem in spectral graph theory.

The eigenvector of eigenvalue 0 of  $\mathbf{N}$  is  $\mathbf{d}^{1/2}$ , by which I mean the vector whose entry for vertex  $u$  is the square root of the degree of  $u$ . Observe that

$$\mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} \mathbf{d}^{1/2} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{1} = \mathbf{D}^{-1/2} \mathbf{0} = \mathbf{0}.$$

The eigenvector of  $\nu_2$  is given by

$$\arg \min_{\mathbf{x} \perp \mathbf{d}^{1/2}} \frac{\mathbf{x}^T \mathbf{N} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

Transferring back into the variable  $\mathbf{y}$ , and observing that

$$\mathbf{x}^T \mathbf{d}^{1/2} = \mathbf{y}^T \mathbf{D}^{1/2} \mathbf{d}^{1/2} = \mathbf{y}^T \mathbf{d},$$

we find

$$\nu_2 = \min_{\mathbf{y} \perp \mathbf{d}} \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}}.$$

## 11.4 Cheeger's Inequality

Cheeger's inequality proves that if we have a vector  $\mathbf{y}$ , orthogonal to  $\mathbf{d}$ , for which the generalized Rayleigh quotient (11.1) is small, then one can obtain a set of small conductance from  $\mathbf{y}$ . We obtain such a set by carefully choosing a real number  $t$ , and setting

$$S_t = \{u : \mathbf{y}(u) \leq t\}.$$

**Theorem 11.4.1.** *Let  $\mathbf{y}$  be a vector orthogonal to  $\mathbf{d}$ . Then, there is a number  $t$  for which the set  $S_t = \{u : \mathbf{y}(u) < t\}$  satisfies*

$$\phi(S_t) \leq \sqrt{2 \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}}}.$$

Before proving the theorem, I wish to make one small point about the denominator in the expression above. It is essentially minimized when  $\mathbf{y}^T \mathbf{d} = 0$ , at least with regards to shifts.

**Lemma 11.4.2.** *Let  $\mathbf{v}_z = \mathbf{y} + z\mathbf{1}$ . Then, the minimum of  $\mathbf{v}_z^T \mathbf{D} \mathbf{v}_z^T$  is achieved at the  $z$  for which  $\mathbf{v}_z^T \mathbf{d} = 0$ .*

*Proof.* The derivative with respect to  $z$  is

$$2\mathbf{d}^T \mathbf{v}_z,$$

and the minimum is achieved when this derivative is zero.  $\square$

We begin our proof of Cheeger's inequality by defining

$$\rho = \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}}.$$

So, we need to show that there is a  $t$  for which  $\phi(S_t) \leq \sqrt{2\rho}$ .

By renumbering the vertices, we may assume without loss of generality that

$$\mathbf{y}(1) \leq \mathbf{y}(2) \leq \cdots \leq \mathbf{y}(n).$$

We begin with some normalization. Let  $j$  be the least number for which

$$\sum_{u=1}^j d(u) \geq d(V)/2.$$

We would prefer a vector that is centered at  $j$ . So, set

$$\mathbf{z} = \mathbf{y} - \mathbf{y}(j)\mathbf{1}.$$

This vector  $\mathbf{z}$  satisfies  $\mathbf{z}(j) = 0$ , and, by Lemma 11.4.2,

$$\frac{\mathbf{z}^T \mathbf{L} \mathbf{z}}{\mathbf{z}^T \mathbf{D} \mathbf{z}} \leq \rho.$$

We also multiply  $\mathbf{z}$  by a constant so that

$$\mathbf{z}(1)^2 + \mathbf{z}(n)^2 = 1.$$

Recall that

$$\phi(S) = \frac{|\partial(S)|}{\min(d(S), d(V - S))}.$$

We will define a distribution on  $t$  for which we can prove that

$$\mathbb{E} [|\partial(S_t)|] \leq \sqrt{2\rho} \mathbb{E} [\min(d(S_t), d(V - S_t))].$$

This implies<sup>1</sup> that there is some  $t$  for which

$$|\partial(S_t)| \leq \sqrt{2\rho} \min(d(S_t), d(V - S_t)),$$

which means  $\phi(S) \leq \sqrt{2\rho}$ .

To switch from working with  $\mathbf{y}$  to working with  $\mathbf{z}$ , define We will set  $S_t = \{u : \mathbf{z}(u) \leq t\}$ . Trevisan had the remarkable idea of choosing  $t$  between  $\mathbf{z}(1)$  and  $\mathbf{z}(n)$  with probability density  $2|t|$ . That is, the probability that  $t$  lies in the interval  $[a, b]$  is

$$\int_{t=a}^b 2|t|.$$

---

<sup>1</sup>If this is not immediately clear, note that it is equivalent to assert that  $\mathbb{E} [\sqrt{2\rho} \min(d(S), d(V - S)) - |\partial(S)|] \geq 0$ , which means that there must be some  $S$  for which the expression is non-negative.

To see that the total probability is 1, observe that

$$\int_{t=z(1)}^{z(n)} 2|t| = \int_{t=z(1)}^0 2|t| + \int_{t=0}^{z(n)} 2|t| = z(n)^2 + z(1)^2 = 1,$$

as  $z(1) \leq z(j) \leq z(n)$  and  $z(j) = 0$ .

Similarly, the probability that  $t$  lies in the interval  $[a, b]$  is

$$\int_{t=a}^b 2|t| = \operatorname{sgn}(b)b^2 - \operatorname{sgn}(a)a^2,$$

where

$$\operatorname{sgn}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0, \text{ and} \\ -1 & \text{if } x < 0. \end{cases}$$

### Lemma 11.4.3.

$$\mathbb{E}_t [|\partial(S_t)|] = \sum_{(u,v) \in E} Pr_t [(u,v) \in \partial(S_t)] \leq \sum_{(u,v) \in E} |z(u) - z(v)| (|z(u)| + |z(v)|). \quad (11.3)$$

*Proof.* An edge  $(u, v)$  with  $z(u) \leq z(v)$  is on the boundary of  $S$  if

$$z(u) \leq t < z(v).$$

The probability that this happens is

$$\operatorname{sgn}(z(v))z(v)^2 - \operatorname{sgn}(z(u))z(u)^2 = \begin{cases} |z(u)^2 - z(v)^2| & \text{when } \operatorname{sgn}(u) = \operatorname{sgn}(v), \\ z(u)^2 + z(v)^2 & \text{when } \operatorname{sgn}(u) \neq \operatorname{sgn}(v). \end{cases}$$

We now show that both of these terms are upper bounded by

$$|z(u) - z(v)| (|z(u)| + |z(v)|).$$

Regardless of the signs,

$$|z(u)^2 - z(v)^2| = |(z(u) - z(v))(z(u) + z(v))| \leq |z(u) - z(v)| (|z(u)| + |z(v)|).$$

When  $\operatorname{sgn}(u) = -\operatorname{sgn}(v)$ ,

$$z(u)^2 + z(v)^2 \leq (z(u) - z(v))^2 = |z(u) - z(v)| (|z(u)| + |z(v)|).$$

□

We now derive a formula for the expected denominator of  $\phi$ .

### Lemma 11.4.4.

$$\mathbb{E}_t [\min(d(S_t), d(V - S_t))] = \mathbf{z}^T \mathbf{D} \mathbf{z}.$$

*Proof.* Observe that

$$\mathbb{E}_t [d(S_t)] = \sum_u \Pr_t [u \in S_t] d(u) = \sum_u \Pr_t [\mathbf{z}(u) \leq t] d(u).$$

The result of our centering of  $\mathbf{z}$  at  $j$  is that

$$\begin{aligned} t < 0 &\implies d(S) = \min(d(S), d(V - S)), \quad \text{and} \\ t \geq 0 &\implies d(V - S) = \min(d(S), d(V - S)). \end{aligned}$$

That is, for  $u < j$ ,  $u$  is in the smaller set if  $t < 0$ ; and, for  $u \geq j$ ,  $u$  is in the smaller set if  $t \geq 0$ . So,

$$\begin{aligned} \mathbb{E}_t [\min(d(S_t), d(V - S_t))] &= \sum_{u < j} \Pr [\mathbf{z}(u) < t \text{ and } t < 0] d(u) + \sum_{u \geq j} \Pr [\mathbf{z}(u) > t \text{ and } t \geq 0] d(u) \\ &= \sum_{u < j} \Pr [\mathbf{z}(u) < t < 0] d(u) + \sum_{u \geq j} \Pr [\mathbf{z}(u) > t \geq 0] d(u) \\ &= \sum_{u < j} \mathbf{z}(u)^2 d(u) + \sum_{u \geq j} \mathbf{z}(u)^2 d(u) \\ &= \sum_u \mathbf{z}(u)^2 d(u) \\ &= \mathbf{z}^T \mathbf{D} \mathbf{z}. \end{aligned}$$

□

Recall that our goal is to prove that

$$\mathbb{E} [|\partial(S_t)|] \leq \sqrt{2\rho} \mathbb{E} [\min(d(S_t), d(V - S_t))],$$

and we know that

$$\mathbb{E}_t [\min(d(S_t), d(V - S_t))] = \sum_u \mathbf{z}(u)^2 d(u)$$

and that

$$\mathbb{E}_t [|\partial(S_t)|] \leq \sum_{(u,v) \in E} |\mathbf{z}(u) - \mathbf{z}(v)| (|\mathbf{z}(u)| + |\mathbf{z}(v)|).$$

We may use the Cauchy-Schwartz inequality to upper bound the term above by

$$\sqrt{\sum_{(u,v) \in E} (\mathbf{z}(u) - \mathbf{z}(v))^2} \sqrt{\sum_{(u,v) \in E} (|\mathbf{z}(u)| + |\mathbf{z}(v)|)^2}. \tag{11.4}$$

We have defined  $\rho$  so that the term under the left-hand square root is at most

$$\rho \sum_u \mathbf{z}(u)^2 d(u).$$

To bound the right-hand square root, we observe

$$\sum_{(u,v) \in E} (|\mathbf{z}(u)| + |\mathbf{z}(v)|)^2 \leq 2 \sum_{(u,v) \in E} \mathbf{z}(u)^2 + \mathbf{z}(v)^2 = 2 \sum_u \mathbf{z}(u)^2 d(u).$$

Putting all these inequalities together yields

$$\begin{aligned}\mathbb{E} [|\partial(S)|] &\leq \sqrt{\rho \sum_u \mathbf{z}(u)^2 d(u)} \sqrt{2 \sum_u \mathbf{z}(u)^2 d(u)} \\ &= \sqrt{2\rho} \sum_u \mathbf{z}(u)^2 d(u) \\ &= \sqrt{2\rho} \mathbb{E} [\min(d(S), d(V - S))].\end{aligned}$$

I wish to point out two important features of this proof:

1. This proof does not require  $\mathbf{y}$  to be an eigenvector—it obtains a cut from any vector  $\mathbf{y}$  that is orthogonal to  $\mathbf{d}$ .
2. This proof goes through almost without change for weighted graphs. The main difference is that for weighted graphs we measure the sum of the weights of edges on the boundary instead of their number. The main difference in the proof is that lines (11.3) and (11.4) become

$$\begin{aligned}\mathbb{E} [w(\partial(S))] &= \sum_{(u,v) \in E} \Pr[(u,v) \in \partial(S)] w_{u,v} \\ &\leq \sum_{(u,v) \in E} |\mathbf{z}(u) - \mathbf{z}(v)| (|\mathbf{z}(u)| + |\mathbf{z}(v)|) w_{u,v} \\ &\leq \sqrt{\sum_{(u,v) \in E} w_{u,v} (\mathbf{z}(u) - \mathbf{z}(v))^2} \sqrt{\sum_{(u,v) \in E} w_{u,v} (|\mathbf{z}(u)| + |\mathbf{z}(v)|)^2},\end{aligned}$$

and we observe that

$$\sum_{(u,v) \in E} w_{u,v} (|\mathbf{z}(u)| + |\mathbf{z}(v)|)^2 \leq 2 \sum_u \mathbf{z}(u)^2 d(u).$$

The only drawback that I see to the approach that we took in this proof is that the application of Cauchy-Schwartz is a little mysterious. Shang-Hua Teng and I came up with a proof that avoids this by introducing one inequality for each edge. If you want to see that proof, look at my notes from 2009.

## A Proof of (11.2)

**Lemma A.1.** *For every  $S \subset V$ ,*

$$\phi(S) \geq \nu_2/2.$$

*Proof.* As in Lecture 2, we would like to again use  $\chi_S$  as a test vector. But, it is not orthogonal to  $\mathbf{d}$ . To fix this, we subtract a constant. Set

$$\mathbf{y} = \chi_S - \sigma \mathbf{1},$$

where

$$\sigma = d(S)/d(V).$$

You should now check that  $\mathbf{y}^T \mathbf{d} = 0$ :

$$\mathbf{y}^T \mathbf{d} = \chi_S^T \mathbf{d} - \sigma \mathbf{1}^T \mathbf{d} = d(S) - (d(S)/d(V))d(V) = 0.$$

We already know that

$$\mathbf{y}^T \mathbf{L} \mathbf{y} = |\partial(S)|.$$

It remains to compute  $\mathbf{y}^T \mathbf{D} \mathbf{y}$ . If you remember the previous computation like this, you would guess that it is  $d(S)(1 - \sigma) = d(S)d(V - S)/d(V)$ , and you would be right:

$$\begin{aligned} \mathbf{y}^T \mathbf{D} \mathbf{y} &= \sum_{u \in S} d(u)(1 - \sigma)^2 + \sum_{u \notin S} d(u)\sigma^2 \\ &= d(S)(1 - \sigma)^2 + d(V - S)\sigma^2 \\ &= d(S) - 2d(S)\sigma + d(V)\sigma^2 \\ &= d(S) - d(S)\sigma \\ &= d(S)d(V - S)/d(V). \end{aligned}$$

So,

$$\nu_2 \leq \frac{\mathbf{y}^T \mathbf{L} \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}} = \frac{|\partial(S)| d(V)}{d(S)d(V - S)}. \quad (11.5)$$

As the larger of  $d(S)$  and  $d(V - S)$  is at least half of  $d(V)$ , we find

$$\nu_2 \leq 2 \frac{|\partial(S)|}{\min(d(S), d(V - S))}.$$

□

## References

- [Alo86] N. Alon. Eigenvalues and expanders. *Combinatorica*, 6(2):83–96, 1986.
- [AM85] Noga Alon and V. D. Milman.  $\lambda_1$ , isoperimetric inequalities for graphs, and superconcentrators. *J. Comb. Theory, Ser. B*, 38(1):73–88, 1985.
- [Che70] J. Cheeger. A lower bound for smallest eigenvalue of the Laplacian. In *Problems in Analysis*, pages 195–199, Princeton University Press, 1970.
- [Chu97] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.
- [Dod84] Jozef Dodziuk. Difference equations, isoperimetric inequality and transience of certain random walks. *Transactions of the American Mathematical Society*, 284(2):787–794, 1984.
- [LS88] Gregory F. Lawler and Alan D. Sokal. Bounds on the  $l^2$  spectrum for Markov chains and Markov processes: A generalization of Cheeger’s inequality. *Transactions of the American Mathematical Society*, 309(2):557–580, 1988.

- [SJ89] Alistair Sinclair and Mark Jerrum. Approximate counting, uniform generation and rapidly mixing Markov chains. *Information and Computation*, 82(1):93–133, July 1989.
- [Tre11] Luca Trevisan. Lecture 4 from cs359g: Graph partitioning and expanders, stanford university, January 2011. available at <http://theory.stanford.edu/~trevisan/cs359g/lecture04.pdf>.
- [Var85] N. Th. Varopoulos. Isoperimetric inequalities and Markov chains. *Journal of Functional Analysis*, 63(2):215 – 239, 1985.

## 12.1 Overview

In this lecture we will see how the analysis of random walks, spring networks, and resistor networks leads to the consideration of systems of linear equations in Laplacian matrices. The main purpose of this lecture is to introduce concepts and language that we will use extensively in the rest of the course.

## 12.2 Harmonic Functions

The theme of this whole lecture will be *harmonic functions* on graphs. These will be defined in terms of a weighted graph  $G = (V, E, w)$  and a set of *boundary* vertices  $B \subseteq V$ . We let  $S = V - B$  (I use “-” for set-minus). We will assume throughout this lecture that  $G$  is connected and that  $B$  is nonempty.

A function  $\mathbf{x} : V \rightarrow \mathbf{R}$  is said to be *harmonic* at a vertex  $a$  if the value of  $\mathbf{x}$  at  $a$  is the weighted average of its values at the neighbors of  $a$  where the weights are given by  $w$ :

$$\mathbf{x}(a) = \frac{1}{d_a} \sum_{b \sim a} w_{a,b} \mathbf{x}(b). \quad (12.1)$$

The function  $\mathbf{x}$  is harmonic on  $S$  if it is harmonic for all  $a \in S$ .

## 12.3 Random Walks on Graphs

Consider the standard (not lazy) random walk on the graph  $G$ . Recall that when the walk is at a vertex  $a$ , the probability it moves to a neighbor  $b$  is

$$\frac{w_{a,b}}{d_a}.$$

Distinguish two special nodes in the graph that we will call  $s$  and  $t$ , and run the random walk until it hits either  $s$  or  $t$ . We view  $s$  and  $t$  as the boundary, so  $B = \{s, t\}$ .

Let  $\mathbf{x}(a)$  be the probability that a walk that starts at  $a$  will stop at  $s$ , rather than at  $t$ . We have the boundary conditions  $\mathbf{x}(s) = 1$  and  $\mathbf{x}(t) = 0$ . For every other node  $a$  the chance that the walk

stops at  $s$  is the sum over the neighbors  $b$  of  $a$  of the chance that the walk moves to  $b$ , times the chance that a walk from  $b$  stops at  $s$ . That is,

$$\mathbf{x}(a) = \sum_{b \sim a} \frac{w_{a,b}}{d_a} \mathbf{x}(b).$$

So, the function  $\mathbf{x}$  is harmonic at every vertex in  $V - B$ .

For example, consider the path graph  $P_n$ . Let's make  $s = n$  and  $t = 1$ . So, the walk stops at either end. We then have  $\mathbf{x}(n) = 1$ ,  $\mathbf{x}(1) = 0$ . It is easy to construction at least one solution to the harmonic equations (12.1): we can set

$$\mathbf{x}(a) = \frac{a-1}{n-1}.$$

It essentially follows from the definitions that there can be only one vector  $\mathbf{x}$  that solves these equations. But, we will prove this algebraically later in lecture.

These solutions tell us that if the walk starts at node  $a$ , the chance that it ends at node  $n$  is  $(a-1)/(n-1)$ . This justifies some of our analysis of the Bolas graph from Lecture 10.

Of course, the exact same analysis goes through for the lazy random walks: those give

$$\mathbf{x}(a) = (1/2)\mathbf{x}(a) + (1/2) \sum_{b \sim a} \frac{w_{a,b}}{d_a} \mathbf{x}(b) \iff \mathbf{x}(a) = \sum_{b \sim a} \frac{w_{a,b}}{d_a} \mathbf{x}(b).$$

## 12.4 Spring Networks

We begin by imagining that every edge of a graph  $G = (V, E)$  is an ideal spring or rubber band. They are joined together at the vertices. Given such a structure, we will pick a subset of the vertices  $B \subseteq V$  and fix the location of every vertex in  $B$ . For example, you could nail each vertex in  $B$  onto a point in the real line, or onto a board in  $\mathbb{R}^2$ . We will then study where the other vertices wind up.

We can use Hooke's law to figure this out. To begin, assume that each rubber band is an ideal spring with spring constant 1. If your graph is weighted, then the spring constant of each edge should be its weight. If a rubber band connects vertices  $a$  and  $b$ , then Hooke's law tells us that the force it exerts at node  $a$  is in the direction of  $b$  and is proportional to the distance between  $a$  and  $b$ . Let  $\mathbf{x}(a)$  be the position of each vertex  $a$ . You should begin by thinking of  $\mathbf{x}(a)$  being in  $\mathbb{R}$ , but you will see that it is just as easy to make it a vector in  $\mathbb{R}^2$  or  $\mathbb{R}^k$  for any  $k$ .

The force the rubber band between  $a$  and  $b$  exerts on  $a$  is

$$\mathbf{x}(b) - \mathbf{x}(a).$$

In a stable configuration, all of the vertices that have not been nailed down must experience a zero

net force. That is

$$\begin{aligned} \sum_{b \sim a} (\mathbf{x}(b) - \mathbf{x}(a)) = 0 &\iff \sum_{b \sim a} \mathbf{x}(b) = d_a \mathbf{x}(a) \\ &\iff \frac{1}{d_a} \sum_{b \sim a} \mathbf{x}(b) = \mathbf{x}(a). \end{aligned}$$

In a stable configuration, every vertex that is not on the boundary must be the average of its neighbors.

In the weighted case, we would have for each  $a \in V - B$

$$\frac{1}{d_a} \sum_{b \sim a} w_{a,b} \mathbf{x}(b) = \mathbf{x}(a).$$

That is,  $\mathbf{x}$  is *harmonic* on  $V - B$ .

We will next show that the equations (12.1) have a solution, and that it is unique<sup>1</sup> if the underlying graph is connected and  $B$  is nonempty. But first, consider again the path graph  $P_n$  with the endpoints fixed:  $B = \{1, n\}$ . Let us fix them to the values  $\mathbf{f}(1) = 1$  and  $\mathbf{f}(n) = n$ . The only solution to the equations (12.1) is the obvious one: vertex  $i$  is mapped to  $i$ :  $\mathbf{x}(i) = i$  for all  $i$ .

## 12.5 Laplacian linear equations

If we rewrite equation (12.1) as

$$d_a \mathbf{x}(a) - \sum_{b \sim a} w_{a,b} \mathbf{x}(b) = 0, \quad (12.2)$$

we see that it corresponds to the row of the Laplacian matrix corresponding to vertex  $a$ . So, we may find a solution to the equations (12.1) by solving a system of equations in the submatrix of the Laplacian indexed by vertices in  $V - B$ .

To be more concrete, I will set up those equations. For each vertex  $a \in B$ , let its position be fixed to  $\mathbf{f}(a)$ . Then, we can re-write equation (12.2) as

$$d_a \mathbf{x}(a) - \sum_{b \notin B: (a,b) \in E} w_{a,b} \mathbf{x}(b) = \sum_{b \in B: (a,b) \in E} w_{a,b} \mathbf{f}(b),$$

for each  $a \in V - B$ . So, all of the boundary terms wind up in the right-hand vector.

Let  $S = V - B$ . We now see that this is an equation of the form

$$\mathbf{L}(S, S) \mathbf{x}(S) = \mathbf{r}, \quad \text{with } \mathbf{r} = \mathbf{M}(S, :) \mathbf{f}.$$

By  $\mathbf{L}(S, S)$  I mean the submatrix of  $\mathbf{L}$  indexed by rows and columns of  $S$ , and by  $\mathbf{x}(S)$  I mean the sub-vector of  $\mathbf{x}$  indexed by  $S$ .

---

<sup>1</sup>It can only fail to be unique if there is a connected component that contains no vertices of  $B$ .

We can then write the condition that entries of  $B$  are fixed to  $\mathbf{f}$  by

$$\mathbf{x}(B) = \mathbf{f}(B).$$

We have reduced the problem to that of solving a system of equations in a submatrix of the Laplacian.

Submatrices of Laplacians are a lot like Laplacians, except that they are positive definite. To see this, note that all of the off-diagonals of the submatrix of  $\mathbf{L}$  agree with all the off-diagonals of the Laplacian of the induced subgraph on the internal vertices. But, some of the diagonals are larger: the diagonals of nodes in the submatrix account for both edges in the induced subgraph and edges to the vertices in  $B$ .

**Claim 12.5.1.** *Let  $\mathbf{L}$  be the Laplacian of  $G = (V, E, w)$ , let  $B \subseteq V$ , and let  $S = V - B$ . Then,*

$$\mathbf{L}(S, S) = \mathbf{L}_{G(S)} + \mathbf{X}_S,$$

where  $G(S)$  is the subgraph induced on the vertices in  $S$  and  $\mathbf{X}_S$  is the diagonal matrix with entries

$$\mathbf{X}_S(a, a) = \sum_{b \sim a, b \in B} w_{a,b}, \quad \text{for } a \in S.$$

**Lemma 12.5.2.** *Let  $\mathbf{L}$  be the Laplacian matrix of a connected graph and let  $\mathbf{X}$  be a nonnegative, diagonal matrix with at least one nonzero entry. Then,  $\mathbf{L} + \mathbf{X}$  is positive definite.*

*Proof.* We will prove that  $\mathbf{x}^T(\mathbf{L} + \mathbf{X})\mathbf{x} > 0$  for every nonzero vector  $\mathbf{x}$ . As both  $\mathbf{L}$  and  $\mathbf{X}$  are positive semidefinite, we have

$$\mathbf{x}^T(\mathbf{L} + \mathbf{X})\mathbf{x} \geq \min(\mathbf{x}^T \mathbf{L} \mathbf{x}, \mathbf{x}^T \mathbf{X} \mathbf{x}).$$

As the graph is connected,  $\mathbf{x}^T \mathbf{L} \mathbf{x}$  is positive unless  $\mathbf{x}$  is a constant vector. If  $\mathbf{x} = c\mathbf{1}$  for some  $c \neq 0$ , then we obtain

$$c^2 \mathbf{1}^T (\mathbf{L} + \mathbf{X}) \mathbf{1} = c^2 \mathbf{1}^T \mathbf{X} \mathbf{1} = c^2 \sum_i \mathbf{X}(i, i) > 0.$$

□

**Lemma 12.5.3.** *Let  $\mathbf{L}$  be the Laplacian matrix of a connected graph  $G = (V, E, w)$ , let  $B$  be a nonempty, proper subset of  $V$ , and let  $S = V - B$ . Then,  $\mathbf{L}(S, S)$  is positive definite.*

*Proof.* Let  $S_1, \dots, S_k$  be the connected components of vertices of  $G(S)$ . We can use these to write  $\mathbf{L}(S, S)$  as a block matrix with blocks equal to  $\mathbf{L}(S_i, S_i)$ . Each of these blocks can be written

$$\mathbf{L}(S_i, S_i) = \mathbf{L}_{G_{S_i}} + \mathbf{X}_{S_i}.$$

As  $G$  is connected, there must be some vertex in  $S_i$  with an edge to a vertex not in  $S_i$ . This implies that  $\mathbf{X}_{S_i}$  is not the zero matrix, and so we can apply Lemma 12.5.2 to prove that  $\mathbf{L}(S_i, S_i)$  is invertible. □

As the matrix  $\mathbf{L}(S, S)$  is invertible, the equations have a solution, and it must be unique.

## 12.6 Energy

Physics also tells us that the vertices will settle into the position that minimizes the potential energy. The potential energy of an ideal linear spring with constant  $w$  when stretched to length  $l$  is

$$\frac{1}{2}wl^2.$$

So, the potential energy in a configuration  $\mathbf{x}$  is given by

$$\mathcal{E}(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{2} \sum_{(a,b) \in E} w_{a,b}(\mathbf{x}(a) - \mathbf{x}(b))^2. \quad (12.3)$$

For any  $\mathbf{x}$  that minimizes the energy, the partial derivative of the energy with respect to each variable must be zero. In this case, the variables are  $\mathbf{x}(a)$  for  $a \in S$ . The partial derivative with respect to  $\mathbf{x}(a)$  is

$$\frac{1}{2} \sum_{b \sim a} w_{a,b} 2(\mathbf{x}(a) - \mathbf{x}(b)) = \sum_{b \sim a} w_{a,b} (\mathbf{x}(a) - \mathbf{x}(b)).$$

Setting this to zero gives the equations we previously derived: (12.1).

## 12.7 Resistor Networks

We now consider a related physical model of a graph in which we treat every edge as a resistor. If the graph is unweighted, we will assume that each resistor has resistance 1. If an edge  $e$  has weight  $w_e$ , we will give the corresponding resistor resistance  $r_e = 1/w_e$ . The reason is that when the weight of an edge is very small, the edge is barely there, so it should correspond to very high resistance. Having no edge corresponds to having a resistor of infinite resistance.

Recall Ohm's law:

$$V = IR.$$

That is, the potential drop across a resistor ( $V$ ) is equal to the current flowing over the resistor ( $I$ ) times the resistance ( $R$ ). To apply this in a graph, we will define for each edge  $(a, b)$  the current flowing from  $a$  to  $b$  to be  $\mathbf{i}(a, b)$ . As this is a directed quantity, we define

$$\mathbf{i}(b, a) = -\mathbf{i}(a, b).$$

I now let  $\mathbf{v} \in \mathbb{R}^V$  be a vector of potentials (voltages) at vertices. Given these potentials, we can figure out how much current flows on each edge by the formula

$$\mathbf{i}(a, b) = \frac{1}{r_{a,b}} (\mathbf{v}(a) - \mathbf{v}(b)) = w_{a,b} (\mathbf{v}(a) - \mathbf{v}(b)).$$

That is, we adopt the convention that current flows from high voltage to low voltage. We would like to write this equation in matrix form. The one complication is that each edge comes up twice in  $\mathbf{i}$ . So, to treat  $\mathbf{i}$  as a vector we will have each edge show up exactly once as  $(a, b)$  when  $a < b$ .

We now define the *signed edge-vertex adjacency matrix* of the graph  $\mathbf{U}$  to be the matrix with rows indexed by edges and columns indexed by vertices such that

$$\mathbf{U}((a, b), c) = \begin{cases} 1 & \text{if } a = c \\ -1 & \text{if } b = c \\ 0 & \text{otherwise.} \end{cases}$$

Thus the row of  $\mathbf{U}$  corresponding to edge  $(a, b)$  is  $\mathbf{U}((a, b), :) = \boldsymbol{\delta}_a^T - \boldsymbol{\delta}_b^T$ .

Define  $\mathbf{W}$  to be the diagonal matrix with rows and columns indexed by edges with the weights of the edges on the diagonals. We then have

$$\mathbf{i} = \mathbf{W} \mathbf{U} \mathbf{v}.$$

Also recall that resistor networks cannot hold current. So, all the current entering a vertex  $a$  from edges in the graph must exit  $a$  to an external source. Let  $\mathbf{i}_{ext} \in \mathbb{R}^V$  denote the external currents, where  $\mathbf{i}_{ext}(a)$  is the amount of current entering the graph through node  $a$ . We then have

$$\mathbf{i}_{ext}(a) = \sum_{b \sim a} \mathbf{i}(a, b).$$

In matrix form, this becomes

$$\mathbf{i}_{ext} = \mathbf{U}^T \mathbf{i} = \mathbf{U}^T \mathbf{W} \mathbf{U} \mathbf{v}. \quad (12.4)$$

The matrix

$$\mathbf{L} \stackrel{\text{def}}{=} \mathbf{U}^T \mathbf{W} \mathbf{U}$$

is, of course, the Laplacian. This is another way of writing the expression that we derived in Lecture 3:

$$\mathbf{L} = \sum_{a \sim b} w_{a,b} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T.$$

It is often helpful to think of the nodes  $a$  for which  $\mathbf{i}_{ext}(a) \neq 0$  as being *boundary* nodes. We will call the other nodes *internal*. Let's see what the equation

$$\mathbf{i}_{ext} = \mathbf{L} \mathbf{v}.$$

means for the internal nodes. If the graph is unweighted and  $a$  is an internal node, then the  $a$ th row of this equation is

$$0 = (\boldsymbol{\delta}_a^T \mathbf{L}) \mathbf{v} = \sum_{a \sim b} (\mathbf{v}(a) - \mathbf{v}(b)) = d_a \mathbf{v}(a) - \sum_{a \sim b} \mathbf{v}(b).$$

That is,

$$\mathbf{v}(a) = \frac{1}{d_a} \sum_{a \sim b} \mathbf{v}(b),$$

which means that  $\mathbf{v}$  is harmonic at  $a$ . Of course, the same holds in weighted graphs.

## 12.8 Solving for currents

We are often interested in applying (12.4) in the reverse: given a vector of external currents  $\mathbf{i}_{ext}$  we solve for the induced voltages by

$$\mathbf{v} = \mathbf{L}^{-1} \mathbf{i}_{ext}.$$

This at first appears problematic, as the Laplacian matrix does not have an inverse. The way around this problem is to observe that we are only interested in solving these equations for vectors  $\mathbf{i}_{ext}$  for which the system has a solution. In the case of a connected graph, this equation will have a solution if the sum of the values of  $\mathbf{i}_{ext}$  is zero. That is, if the current going in to the circuit equals the current going out. These are precisely the vectors that are in the span of the Laplacian.

To obtain the solution to this equation, we multiply  $\mathbf{i}_{ext}$  by the Moore-Penrose *pseudo-inverse* of  $\mathbf{L}$ .

**Definition 12.8.1.** *The pseudo-inverse of a symmetric matrix  $\mathbf{L}$ , written  $\mathbf{L}^+$ , is the matrix that has the same span as  $\mathbf{L}$  and that satisfies*

$$\mathbf{L}\mathbf{L}^+ = \mathbf{\Pi},$$

where  $\mathbf{\Pi}$  is the symmetric projection onto the span of  $\mathbf{L}$ .

I remind you that a matrix  $\mathbf{\Pi}$  is a symmetric projection if  $\mathbf{\Pi}$  is symmetric and  $\mathbf{\Pi}^2 = \mathbf{\Pi}$ . This is equivalent to saying that all of its eigenvalues are 0 or 1. We also know that  $\mathbf{\Pi} = (1/n)\mathbf{L}_{K_n}$ .

The symmetric case is rather special. As  $\mathbf{L}\mathbf{\Pi} = \mathbf{L}$ , the other following properties of the Moore-Penrose pseudo inverse follow from this one:

$$\begin{aligned} \mathbf{L}^+\mathbf{L} &= \mathbf{\Pi}, \\ \mathbf{L}\mathbf{L}^+\mathbf{L} &= \mathbf{L} \\ \mathbf{L}^+\mathbf{L}\mathbf{L}^+ &= \mathbf{L}^+. \end{aligned}$$

It is easy to find a formula for the pseudo-inverse. First, let  $\Psi$  be the matrix whose  $i$ th column is  $\psi_i$  and let  $\Lambda$  be the diagonal matrix with  $\lambda_i$  on the  $i$ th diagonal. Recall that

$$\mathbf{L} = \Psi \Lambda \Psi^T = \sum_i \lambda_i \psi_i \psi_i^T.$$

**Claim 12.8.2.**

$$\mathbf{L}^+ = \sum_{i>1} (1/\lambda_i) \psi_i \psi_i^T.$$

## 12.9 Electrical Flows and Effective Resistance

We now know that if a resistor network has external currents  $\mathbf{i}_{ext}$ , then the voltages induced at the vertices will be given by

$$\mathbf{v} = \mathbf{L}^+ \mathbf{i}_{ext}.$$

Consider what this means when  $\mathbf{i}_{ext}$  corresponds to a flow of one unit from vertex  $a$  to vertex  $b$ . The resulting voltages are

$$\mathbf{v} = \mathbf{L}^+(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b).$$

Now, let  $c$  and  $d$  be two other vertices. The potential difference between  $c$  and  $d$  is

$$\mathbf{v}(c) - \mathbf{v}(d) = (\boldsymbol{\delta}_c - \boldsymbol{\delta}_d)^T \mathbf{v} = (\boldsymbol{\delta}_c - \boldsymbol{\delta}_d)^T \mathbf{L}^+(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b).$$

Note the amazing reciprocity here: as  $\mathbf{L}$  is symmetric this is equal to

$$(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}^+(\boldsymbol{\delta}_c - \boldsymbol{\delta}_d).$$

So, the potential difference between  $c$  and  $d$  when we flow one unit from  $a$  to  $b$  is the same as the potential difference between  $a$  and  $b$  when we flow one unit from  $c$  to  $d$ .

The *effective resistance* between vertices  $a$  and  $b$  is the resistance between  $a$  and  $b$  when we view the entire network as one complex resistor.

To figure out what this is, recall the equation

$$\mathbf{i}(a, b) = \frac{\mathbf{v}(a) - \mathbf{v}(b)}{r_{a,b}},$$

which holds for one resistor. We use the same equation to define the effective resistance of the whole network between  $a$  and  $b$ . That is, we consider an electrical flow that sends one unit of current into node  $a$  and removes one unit of current from node  $b$ . We then measure the potential difference between  $a$  and  $b$  that is required to realize this current, define this to be the effective resistance between  $a$  and  $b$ , and write it  $R_{\text{eff}}(a, b)$ . As it equals the potential difference between  $a$  and  $b$  in a flow of one unit of current from  $a$  to  $b$ :

$$R_{\text{eff}}(a, b) = (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}^+(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b).$$

## 12.10 Exercise

Prove that for every  $p > 0$

$$\mathbf{L}^p = \boldsymbol{\Psi} \mathbf{A}^p \boldsymbol{\Psi}^T = \sum_i \lambda_i^p \boldsymbol{\psi}_i \boldsymbol{\psi}_i^T.$$

Moreover, this holds for any symmetric matrix. Not just Laplacians.

## References

## Effective Resistance and Schur Complements

Daniel A. Spielman

October 10, 2018

### 13.1 Introduction

In the last lecture, we encountered two types of physical problems on graphs. The first were problems with fixed boundary conditions. In these, we are given fixed values of  $\mathbf{x}(b)$  for some nodes  $b \in B$ , and must either find the values of  $\mathbf{x}(a)$  for  $a \in S = V - B$  that either minimize the Laplacian quadratic form (energy) or so that the vector  $\mathbf{x}$  is harmonic at all of these  $a$ . These are the same problems.

We learned that the solution is given by

$$\mathbf{x}(S) = \mathbf{L}(S, S)^{-1} \mathbf{M}(S, B) \mathbf{x}(B).$$

As  $\mathbf{M}$  is the adjacency matrix and  $S$  and  $B$  are disjoint, we have  $\mathbf{M}(S, B) = -\mathbf{L}(S, B)$ , giving the formula

$$\mathbf{x}(S) = -\mathbf{L}(S, S)^{-1} \mathbf{L}(S, B) \mathbf{x}(B).$$

The other problem we saw was that of computing the voltages that are induced by fixing certain external flows. These were solved by the equations

$$\mathbf{v} = \mathbf{L}^+ \mathbf{i}_{ext}.$$

For those vertices  $a$  for which  $\mathbf{i}_{ext}(a) = 0$ , this equation will result in  $\mathbf{v}$  being harmonic at  $a$ . The previous problem corresponds to fixing voltages at some vertices, rather than fixing flows.

We then defined the *effective* resistance between vertices  $a$  and  $b$  to be the potential difference between  $a$  and  $b$  in the unit flow of one unit from  $a$  to  $b$ :

$$R_{\text{eff}}(a, b) = (\delta_a - \delta_b)^T \mathbf{L}^+ (\delta_a - \delta_b).$$

That is, this is the resistance between  $a$  and  $b$  imposed by the network as a whole.

An alternative way of saying that is that if we only care about vertices  $a$  and  $b$ , we can reduce the rest of the network to a single edge.

In general, we will see that if we wish to restrict our attention to a subset of the vertices,  $B$ , and if we require all other vertices to be internal, then we can construct a network just on  $B$  that factors out the contributions of the internal vertices. The process by which we do this is Gaussian elimination, and the Laplacian of the resulting network on  $B$  is called a Schur Complement.

We will also show that effective resistance is a distance. Other important properties of effective resistance will appear in later lectures.

For now, I observe that effective resistance is the square of a Euclidean distance.

To this end, let  $\mathbf{L}^{+/2}$  denote the square root of  $\mathbf{L}^+$ . Recall that every positive semidefinite matrix has a square root: the square root of a symmetric matrix  $\mathbf{M}$  is the symmetric matrix  $\mathbf{M}^{1/2}$  such that  $(\mathbf{M}^{1/2})^2 = \mathbf{M}$ . If

$$\mathbf{M} = \sum_i \lambda_i \boldsymbol{\psi}_i \boldsymbol{\psi}_i^T$$

is the spectral decomposition of  $\mathbf{M}$ , then

$$\mathbf{M}^{1/2} = \sum_i \lambda_i^{1/2} \boldsymbol{\psi}_i \boldsymbol{\psi}_i^T.$$

We now have

$$\begin{aligned} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}^+ (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) &= \left( \mathbf{L}^{+/2} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \right)^T \mathbf{L}^{+/2} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) = \left\| \mathbf{L}^{+/2} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \right\|^2 \\ &= \left\| \mathbf{L}^{+/2} \boldsymbol{\delta}_a - \mathbf{L}^{+/2} \boldsymbol{\delta}_b \right\|^2 = \text{dist}(\mathbf{L}^{+/2} \boldsymbol{\delta}_a, \mathbf{L}^{+/2} \boldsymbol{\delta}_b)^2. \end{aligned}$$

## 13.2 Effective Resistance through Energy Minimization

As you would imagine, we can also define the effective resistance through effective spring constants. In this case, we view the network of springs as one large compound network. If we define the spring constant to be the number  $w$  so that when  $a$  and  $b$  are stretched to distance  $l$  the potential energy in the spring is  $wl^2/2$ , then we should define the effective spring constant to be twice the entire energy of the network,

$$2\mathcal{E}(\mathbf{x}) \stackrel{\text{def}}{=} \sum_{(u,v) \in E} w_{u,v} (\mathbf{x}(u) - \mathbf{x}(v))^2,$$

when  $\mathbf{x}(a)$  is fixed to 0 and  $\mathbf{x}(b)$  is fixed to 1.

Fortunately, we already know how compute such a vector  $\mathbf{x}$ . Set

$$\mathbf{y} = \mathbf{L}^+ (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a) / R_{\text{eff}}(a, b).$$

We have

$$\mathbf{y}(b) - \mathbf{y}(a) = (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a)^T \mathbf{L}^+ (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a) / R_{\text{eff}}(a, b) = 1,$$

and  $\mathbf{y}$  is harmonic on  $V - \{a, b\}$ . So, we choose

$$\mathbf{x} = \mathbf{y} - \mathbf{1}\mathbf{y}(a).$$

The vector  $\mathbf{x}$  satisfies  $\mathbf{x}(a) = 0$ ,  $\mathbf{x}(b) = 1$ , and it is harmonic on  $V - \{a, b\}$ . So, it is the vector that minimizes the energy subject to the boundary conditions.

To finish, we compute the energy to be

$$\begin{aligned}
\mathbf{x}^T \mathbf{L} \mathbf{x} &= \mathbf{y}^T \mathbf{L} \mathbf{y} \\
&= \frac{1}{(\text{R}_{\text{eff}}(a, b))^2} (\mathbf{L}^+ (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a))^T \mathbf{L} (\mathbf{L}^+ (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a)) \\
&= \frac{1}{(\text{R}_{\text{eff}}(a, b))^2} (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a)^T \mathbf{L}^+ \mathbf{L} \mathbf{L}^+ (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a) \\
&= \frac{1}{(\text{R}_{\text{eff}}(a, b))^2} (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a)^T \mathbf{L}^+ (\boldsymbol{\delta}_b - \boldsymbol{\delta}_a) \\
&= \frac{1}{\text{R}_{\text{eff}}(a, b)}.
\end{aligned}$$

As the weights of edges are the reciprocals of their resistances, and the spring constant corresponds to the weight, this is the formula we would expect.

Resistor networks have an analogous quantity: the energy dissipation (into heat) when current flows through the network. It has the same formula. The reciprocal of the effective resistance is sometimes called the effective conductance.

### 13.3 Examples: Series and Parallel

In the case of a path graph with  $n$  vertices and edges of weight 1, the effective resistance between the extreme vertices is  $n - 1$ .

In general, if a path consists of edges of resistance  $r_{1,2}, \dots, r_{n-1,n}$  then the effective resistance between the extreme vertices is

$$r_{1,2} + \dots + r_{n-1,n}.$$

To see this, set the potential of vertex  $i$  to

$$\mathbf{v}(i) = r_{i,i+1} + \dots + r_{n-1,n}.$$

Ohm's law then tells us that the current flow over the edge  $(i, i + 1)$  will be

$$(\mathbf{v}(i) - \mathbf{v}(i + 1)) / r_{i,i+1} = 1.$$

If we have  $k$  parallel edges between two nodes  $s$  and  $t$  of resistances  $r_1, \dots, r_k$ , then the effective resistance is

$$\text{R}_{\text{eff}}(s, t) = \frac{1}{1/r_1 + \dots + 1/r_k}.$$

To see this, impose a potential difference of 1 between  $s$  and  $t$ . This will induce a flow of  $1/r_i = w_i$  on edge  $i$ . So, the total flow will be

$$\sum_i = 1/r_i = \sum_i w_i.$$

### 13.4 Equivalent Networks, Elimination, and Schur Complements

We have shown that the impact of the entire network on two vertices can be reduced to a network with one edge between them. We will now see that we can do the same for a subset of the vertices. I will do this in two ways: first by viewing  $\mathbf{L}$  as an operator, and then by considering it as a quadratic form.

Let  $B$  be the subset of nodes that we would like to understand ( $B$  stands for *boundary*). All nodes not in  $B$  will be internal. Call them  $I = V - B$ .

As an operator, the Laplacian maps vectors of voltages to vectors of external currents. We want to examine what happens if we fix the voltages at vertices in  $B$ , and require the rest to be harmonic. Let  $\mathbf{v}(B) \in \mathbb{R}^B$  be the voltages at  $B$ . We want the matrix  $\mathbf{L}_B$  such that

$$\mathbf{i}_B = \mathbf{L}_B \mathbf{v}(B)$$

is the vector of external currents at vertices in  $B$  when we impose voltages  $\mathbf{v}(B)$  at vertices of  $B$ . As the internal vertices will have their voltages set to be harmonic, they will not have any external currents.

The remarkable fact that we will discover is that  $\mathbf{L}_B$  is in fact a Laplacian matrix, and that it is obtained by performing Gaussian elimination to remove the internal vertices. **Warning:**  $\mathbf{L}_B$  is not a submatrix of  $\mathbf{L}$ . To prove this, we will move from  $V$  to  $B$  by removing one vertex at a time. We'll start with a graph  $G = (V, E, w)$ , and we will set  $B = \{2, \dots, n\}$ , and we will treat vertex 1 as internal. Let  $N$  denote the set of neighbors of vertex 1.

We want to compute  $\mathbf{L}\mathbf{v}$  given that  $\mathbf{v}(b) = \mathbf{v}(B)(b)$  for  $b \in B$ , and

$$\mathbf{v}(1) = \frac{1}{d_1} \sum_{a \in N} w_{1,a} \mathbf{v}(a). \quad (13.1)$$

That is, we want to substitute the value on the right-hand side for  $\mathbf{v}(1)$  everywhere that it appears in the equation  $\mathbf{i}_{ext} = \mathbf{L}\mathbf{v}$ . The variable  $\mathbf{v}(1)$  only appears in the equation for  $\mathbf{i}_{ext}(a)$  when  $a \in N$ . When it does, it appears with coefficient  $w_{1,a}$ . Recall that the equation for  $\mathbf{i}_{ext}(b)$  is

$$\mathbf{i}_{ext}(b) = d_b \mathbf{v}(b) - \sum_{c \sim b} w_{b,c} \mathbf{v}(c).$$

For  $b \in N$  we expand this by making the substitution for  $\mathbf{v}(1)$  given by (13.1).

$$\begin{aligned} \mathbf{i}_{ext}(b) &= d_b \mathbf{v}(b) - w_{b,1} \mathbf{v}(1) - \sum_{c \sim b, c \neq 1} w_{b,c} \mathbf{v}(c) \\ &= d_b \mathbf{v}(b) - w_{b,1} \frac{1}{d_1} \sum_{a \in N} w_{1,a} \mathbf{v}(a) - \sum_{c \sim b, c \neq 1} w_{b,c} \mathbf{v}(c) \\ &= d_b \mathbf{v}(b) - \sum_{a \in N} \frac{w_{b,1} w_{a,1}}{d_1} \mathbf{v}(a) - \sum_{c \sim b, c \neq 1} w_{b,c} \mathbf{v}(c). \end{aligned}$$

To finish, observe that  $b \in N$ , so we are counting  $b$  in the middle sum above. Removing the double-count gives.

$$\mathbf{i}_{ext}(b) = (d_b - w_{b,1}^2/d_1)\mathbf{v}(b) - \sum_{a \in N, a \neq b} \frac{w_{b,1}w_{a,1}}{d_1}\mathbf{v}(a) - \sum_{c \sim b, c \neq 1} w_{b,c}\mathbf{v}(c).$$

We will show that these revised equations have two interesting properties: they are the result of applying Gaussian elimination to eliminate vertex 1, and the resulting equations are Laplacian.

Let's look at exactly how the matrix has changed. In the row for vertex  $b$ , the edge to vertex 1 was removed, and edges to every vertex  $a \in N$  were added with weights  $\frac{w_{b,1}w_{a,1}}{d_1}$ . And, the diagonal was decreased by  $\frac{w_{b,1}w_{b,1}}{d_1}$ . That should look familiar to you! Overall, the star of edges based at 1 were removed, and a clique on  $N$  was added in which edge  $(a, b)$  has weight

$$\frac{w_{b,1}w_{1,a}}{d_1}.$$

If the initial graph only consisted of a star centered at 1, then the graph we produce on eliminating vertex 1 is exactly the weighted clique you considered in Homework 2. In the next section, we will see that we can use this to solve that problem.

To see that this new system of equations comes from a Laplacian, we observe that

1. It is symmetric.
2. The off-diagonal entries that have been added are negative.
3. The sum of the changes in diagonal and off-diagonal entries is zero, so the row-sum is still zero. This follows from

$$\frac{w_{b,1}^2}{d_1} - \sum_{a \in N} \frac{w_{b,1}w_{a,1}}{d_1} = 0.$$

### 13.4.1 In matrix form by energy

I'm now going to try doing this in terms of the quadratic form. That is, we will compute the matrix  $\mathbf{L}_B$  so that

$$\mathbf{v}(B)^T \mathbf{L}_B \mathbf{v}(B) = \mathbf{v}^T \mathbf{L} \mathbf{v},$$

given that  $\mathbf{v}$  is harmonic at vertex 1 and agrees with  $\mathbf{v}(B)$  elsewhere. The quadratic form that we want to compute is thus given by

$$\begin{pmatrix} \frac{1}{d_1} \sum_{b \sim 1} w_{1,b} \mathbf{v}(b) \\ \mathbf{v}(B) \end{pmatrix}^T \mathbf{L} \begin{pmatrix} \frac{1}{d_1} \sum_{b \sim 1} w_{1,b} \mathbf{v}(b) \\ \mathbf{v}(B) \end{pmatrix}.$$

So that I can write this in terms of the entries of the Laplacian matrix, note that  $d_1 = \mathbf{L}(1, 1)$ , and so

$$\mathbf{v}(1) = \frac{1}{d_1} \sum_{b \sim 1} w_{1,b} \mathbf{v}(b) = -(1/\mathbf{L}(1, 1)) \mathbf{L}(1, B) \mathbf{v}(B).$$

Thus, we can write the quadratic form as

$$\begin{pmatrix} -(1/\mathbf{L}(1,1))\mathbf{L}(1,B)\mathbf{v}(B) \\ \mathbf{v}(B) \end{pmatrix}^T \mathbf{L} \begin{pmatrix} -(1/\mathbf{L}(1,1))\mathbf{L}(1,B)\mathbf{v}(B) \\ \mathbf{v}(B) \end{pmatrix}.$$

If we expand this out, we find that it equals

$$\begin{aligned} & \mathbf{v}(B)^T \mathbf{L}(B, B) \mathbf{v}(B) + \mathbf{L}(1, 1) (-1/\mathbf{L}(1, 1)) \mathbf{L}(1, B) \mathbf{v}(B))^2 + 2\mathbf{v}(1) \mathbf{L}(1, B) (-1/\mathbf{L}(1, 1)) \mathbf{L}(1, B) \mathbf{v}(B) \\ &= \mathbf{v}(B)^T \mathbf{L}(B, B) \mathbf{v}(B) + (\mathbf{L}(1, B) \mathbf{v}(B))^2 / \mathbf{L}(1, 1) - 2(\mathbf{L}(1, B) \mathbf{v}(B))^2 / \mathbf{L}(1, 1) \\ &= \mathbf{v}(B)^T \mathbf{L}(B, B) \mathbf{v}(B) - (\mathbf{L}(1, B) \mathbf{v}(B))^2 / \mathbf{L}(1, 1). \end{aligned}$$

Thus,

$$\mathbf{L}_B = \mathbf{L}(B, B) - \frac{\mathbf{L}(B, 1)\mathbf{L}(1, B)}{\mathbf{L}(1, 1)}.$$

To see that this is the matrix that appears in rows and columns 2 through  $n$  when we eliminate the entries in the first column of  $\mathbf{L}$  by adding multiples of the first row, note that we eliminate entry  $\mathbf{L}(a, 1)$  by adding  $-\mathbf{L}(a, 1)/\mathbf{L}(1, 1)$  times the first row of the matrix to  $\mathbf{L}(a, :)$ . Doing this for all rows in  $B = \{2, \dots, n\}$  results in this formula.

We can again check that  $\mathbf{L}_B$  is a Laplacian matrix. It is clear from the formula that it is symmetric and that the off-diagonal entries are negative. To check that the constant vectors are in the nullspace, we can show that the quadratic form is zero on those vectors. If  $\mathbf{v}(B)$  is a constant vector, then  $\mathbf{v}(1)$  must equal this constant, and so  $\mathbf{v}$  is a constant vector and the value of the quadratic form is 0.

### 13.5 Eliminating Many Vertices

We can of course use the same procedure to eliminate many vertices. We begin by partitioning the vertex set into *boundary* vertices  $B$  and *internal* vertices  $I$ . We can then use Gaussian elimination to eliminate all of the internal vertices. You should recall that the submatrices produced by Gaussian elimination do not depend on the order of the eliminations. So, you may conclude that the matrix  $\mathbf{L}_B$  is uniquely defined.

Or, observe that to eliminate the entries in row  $a \in B$  and columns in  $S$ , using the rows in  $S$ , we need to add those rows,  $\mathbf{L}(S, :)$  to row  $\mathbf{L}(a, :)$  with coefficients  $\mathbf{c}$  so that

$$\mathbf{L}(a, S) + \mathbf{c}\mathbf{L}(S, S) = 0.$$

This gives

$$\mathbf{c} = -\mathbf{L}(a, S)\mathbf{L}(S, S)^{-1},$$

and thus row  $a$  becomes

$$\mathbf{L}(a, :) - \mathbf{L}(a, S)\mathbf{L}(S, S)^{-1}\mathbf{L}(S, :).$$

Restricting to rows and columns in  $B$ , we are left with the matrix

$$\mathbf{L}(B, B) - \mathbf{L}(B, S)\mathbf{L}(S, S)^{-1}\mathbf{L}(S, B).$$

This is called the *schur* complement on  $B$  (or with respect to  $S$ ).

This is equivalent to requiring that the variables in  $S$  be harmonic. Partition a vector  $\mathbf{v}$  into  $\mathbf{v}(B)$  and  $\mathbf{v}(S)$ . The harmonic equations become

$$\mathbf{L}(S, S)\mathbf{v}(S) + \mathbf{L}(S, B)\mathbf{v}(B) = 0,$$

which implies

$$\mathbf{v}(S) = -\mathbf{L}(S, S)^{-1}\mathbf{L}(S, B)\mathbf{v}(B).$$

This gives

$$\mathbf{i}_{ext}(B) = \mathbf{L}(B, S)\mathbf{v}(S) + \mathbf{L}(B, B)\mathbf{v}(B) = -\mathbf{L}(B, S)\mathbf{L}(S, S)^{-1}\mathbf{L}(S, B)\mathbf{v}(B) + \mathbf{L}(B, B)\mathbf{v}(B),$$

and so

$$\mathbf{i}_{ext}(B) = \mathbf{L}_B\mathbf{v}(B), \quad \text{where } \mathbf{L}_B = \mathbf{L}(B, B) - \mathbf{L}(B, S)\mathbf{L}(S, S)^{-1}\mathbf{L}(S, B)\mathbf{v}(B).$$

## 13.6 Effective Resistance is a Distance

A distance is any function on pairs of vertices such that

1.  $\delta(a, a) = 0$  for every vertex  $a$ ,
2.  $\delta(a, b) \geq 0$  for all vertices  $a, b$ ,
3.  $\delta(a, b) = \delta(b, a)$ , and
4.  $\delta(a, c) \leq \delta(a, b) + \delta(b, c)$ .

We claim that the effective resistance is a distance. The only non-trivial part to prove is the triangle inequality, (4).

From the previous section, we know that it suffices to consider graphs with only three vertices: we can reduce any graph to one on just vertices  $a, b$  and  $c$  without changing the effective resistances between them.

**Lemma 13.6.1.** *Let  $a, b$  and  $c$  be vertices in a graph. Then*

$$R_{\text{eff}}(a, b) + R_{\text{eff}}(b, c) \geq R_{\text{eff}}(a, c).$$

*Proof.* Let

$$z = w_{a,b}, y = w_{a,c}, \quad \text{and} x = w_{b,c}.$$

If we eliminate vertex  $c$ , we create an edge between vertices  $a$  and  $b$  of weight

$$\frac{xy}{x+y}.$$

Adding this to the edge that is already there produces weight  $z + \frac{xy}{x+y}$ , for

$$R_{\text{eff } a,b} = \frac{1}{z + \frac{xy}{x+y}} = \frac{1}{\frac{zx+zy+xy}{x+y}} = \frac{x+y}{zx+zy+xy}$$

Working symmetrically, we find that we need to prove that for all positive  $x, y$ , and  $z$

$$\frac{x+y}{zx+zy+xy} + \frac{y+z}{zx+zy+xy} \geq \frac{x+z}{zx+zy+xy},$$

which is of course true. □

### 13.7 An interpretation of Gaussian elimination

This gives us a way of understand how Gaussian elimination solves a system of equations like  $\mathbf{i}_{\text{ext}} = \mathbf{L}\mathbf{v}$ . It constructs a sequence of graphs,  $G_2, \dots, G_n$ , so that  $G_i$  is the effective network on vertices  $i, \dots, n$ . It then solves for the entries of  $\mathbf{v}$  backwards. Given  $\mathbf{v}(i+1), \dots, \mathbf{v}(n)$  and  $\mathbf{i}_{\text{ext}}(i)$ , we can solve for  $\mathbf{v}(i)$ . If  $\mathbf{i}_{\text{ext}}(i) = 0$ , then  $\mathbf{v}(i)$  is set to the weighted average of its neighbors. If not, then we need to take  $\mathbf{i}_{\text{ext}}(i)$  into account here and in the elimination as well. In the case in which we fix some vertices and let the rest be harmonic, there is no such complication.

## More Effective Resistance

*Daniel A. Spielman*

October 15, 2018

## 14.1 Introduction

My plan for this lecture is to teach too much:

1. The Matrix Tree Theorem.
2. Effective Resistance / Leverage Scores, and the probability an edge appears in a random spanning tree.
3. Estimating effective resistances quickly.
4. Rayleigh's Monotonicity Theorem.

## 14.2 Effective Resistance and Energy Dissipation

In the last lecture we saw two ways of defining effective resistance. I will define it one more way, but skip the proof. If a current  $f$  flows through a resistor of resistance  $R$ , the amount of energy that is dissipated as heat is proportional to  $Rf^2$ . If the potential difference across the resistor is  $v$ , then  $f = v/R$ , and the energy dissipation is

$$Rf^2 = v^2/R = wv^2,$$

where  $w$  is the weight of the edge. We can define the effective resistance between vertices  $a$  and  $b$  to be the minimum of the total energy dissipation when we flow one unit of current from  $a$  to  $b$ . You could compute this by evaluating the Laplacian quadratic form on the vector of voltages induced by this flow.

## 14.3 Determinants

To begin, we review some facts about determinants of matrices and characteristic polynomials. We first recall the Leibniz formula for the determinant of a square matrix  $\mathbf{A}$ :

$$\det(\mathbf{A}) = \sum_{\pi} \left( \text{sgn}(\pi) \prod_{i=1}^n \mathbf{A}(i, \pi(i)) \right), \quad (14.1)$$

where the sum is over all permutations  $\pi$  of  $\{1, \dots, n\}$ .

Also recall that the determinant is multiplicative, so for square matrices  $\mathbf{A}$  and  $\mathbf{B}$

$$\det(\mathbf{AB}) = \det(\mathbf{A}) \det(\mathbf{B}). \quad (14.2)$$

Elementary row operations do not change the determinant. If the columns of  $\mathbf{A}$  are the vectors  $\mathbf{a}_1, \dots, \mathbf{a}_n$ , then for every  $c$

$$\det(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n) = \det(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n + c\mathbf{a}_1).$$

This fact gives us two ways of computing the determinant. The first comes from the fact that we can apply elementary row operations to transform  $\mathbf{A}$  into an upper triangular matrix, and (14.1) tells us that the determinant of an upper triangular matrix is the product of its diagonal entries.

The second comes from the observation that the determinant is the volume of the parallelepiped with axes  $\mathbf{a}_1, \dots, \mathbf{a}_n$ : the polytope whose corners are the origin and  $\sum_{i \in S} \mathbf{a}_i$  for every  $S \subseteq \{1, \dots, n\}$ . Let

$$\Pi_{\mathbf{a}_1}$$

be the symmetric projection orthogonal to  $\mathbf{a}_1$ . As this projection amounts to subtracting off a multiple of  $\mathbf{a}_1$  and elementary row operations do not change the determinant,

$$\det(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n) = \det(\mathbf{a}_1, \Pi_{\mathbf{a}_1} \mathbf{a}_2, \dots, \Pi_{\mathbf{a}_1} \mathbf{a}_n).$$

The volume of this parallelepiped is  $\|\mathbf{a}_1\|$  times the volume of the parallelepiped formed by the vectors  $\Pi_{\mathbf{a}_1} \mathbf{a}_2, \dots, \Pi_{\mathbf{a}_1} \mathbf{a}_n$ . I would like to write this as a determinant, but must first deal with the fact that these are  $n - 1$  vectors in an  $n$  dimensional space. The way we first learn to handle this is to project them into an  $n - 1$  dimensional space where we can take the determinant. Instead, we will employ other elementary symmetric functions of the eigenvalues.

## 14.4 Characteristic Polynomials

Recall that the characteristic polynomial of a matrix  $\mathbf{A}$  is

$$\det(x\mathbf{I} - \mathbf{A}).$$

I will write this as

$$\sum_{k=0}^n x^{n-k} (-1)^k \sigma_k(\mathbf{A}),$$

where  $\sigma_k(\mathbf{A})$  is the  $k$ th elementary symmetric function of the eigenvalues of  $\mathbf{A}$ , counted with algebraic multiplicity:

$$\sigma_k(\mathbf{A}) = \sum_{|S|=k} \prod_{i \in S} \lambda_i.$$

Thus,  $\sigma_1(\mathbf{A})$  is the trace and  $\sigma_n(\mathbf{A})$  is the determinant. From this formula, we know that these functions are invariant under similarity transformations.

In Exercise 3 from Lecture 2, you were asked to prove that

$$\sigma_k(\mathbf{A}) = \sum_{|S|=k} \det(\mathbf{A}(S, S)). \quad (14.3)$$

This follows from applying the Leibnitz formula (14.1) to  $\det(x\mathbf{I} - \mathbf{A})$ .

If we return to the vectors  $\Pi_{\mathbf{a}_1}\mathbf{a}_2, \dots, \Pi_{\mathbf{a}_1}\mathbf{a}_n$  from the previous section, we see that the volume of their parallelepiped may be written

$$\sigma_{n-1}(\mathbf{0}_n, \Pi_{\mathbf{a}_1}\mathbf{a}_2, \dots, \Pi_{\mathbf{a}_1}\mathbf{a}_n),$$

as this will be the product of the  $n - 1$  nonzero eigenvalues of this matrix.

Recall that the matrices  $\mathbf{B}\mathbf{B}^T$  and  $\mathbf{B}^T\mathbf{B}$  have the same eigenvalues, up to some zero eigenvalues if they are rectangular. So,

$$\sigma_k(\mathbf{B}\mathbf{B}^T) = \sigma_k(\mathbf{B}^T\mathbf{B}).$$

This gives us one other way of computing the absolute value of the product of the nonzero eigenvalues of the matrix

$$(\Pi_{\mathbf{a}_1}\mathbf{a}_2, \dots, \Pi_{\mathbf{a}_1}\mathbf{a}_n).$$

We can instead compute their square by computing the determinant of the square matrix

$$\begin{pmatrix} \Pi_{\mathbf{a}_1}\mathbf{a}_2 \\ \vdots \\ \Pi_{\mathbf{a}_1}\mathbf{a}_n \end{pmatrix} (\Pi_{\mathbf{a}_1}\mathbf{a}_2, \dots, \Pi_{\mathbf{a}_1}\mathbf{a}_n).$$

When  $\mathbf{B}$  is a singular matrix of rank  $k$ ,  $\sigma_k(\mathbf{B})$  acts as the determinant of  $\mathbf{B}$  restricted to its span. Thus, there are situations in which  $\sigma_k$  is multiplicative. For example, if  $\mathbf{A}$  and  $\mathbf{B}$  both have rank  $k$  and the range of  $\mathbf{A}$  is orthogonal to the nullspace of  $\mathbf{B}$ , then

$$\sigma_k(\mathbf{B}\mathbf{A}) = \sigma_k(\mathbf{B})\sigma_k(\mathbf{A}). \quad (14.4)$$

We will use this identity in the case that  $\mathbf{A}$  and  $\mathbf{B}$  are symmetric and have the same nullspace.

## 14.5 The Matrix Tree Theorem

We will state a slight variant of the standard Matrix-Tree Theorem. Recall that a spanning tree of a graph is a subgraph that is a tree.

**Theorem 14.5.1.** *Let  $G = (V, E, w)$  be a connected, weighted graph. Then*

$$\sigma_{n-1}(\mathbf{L}_G) = n \sum_{\text{spanning trees } T} \prod_{e \in T} w_e.$$

Thus, the eigenvalues allow us to count the sum over spanning trees of the product of the weights of edges in those trees. When all the edge weights are 1, we just count the number of spanning trees in  $G$ .

We first prove this in the case that  $G$  is just a tree.

**Lemma 14.5.2.** *Let  $G = (V, E, w)$  be a weighted tree. Then,*

$$\sigma_{n-1}(\mathbf{L}_G) = n \prod_{e \in E} w_e.$$

*Proof.* For  $a \in V$ , let  $S_a = V - \{a\}$ . We know from (14.3)

$$\sigma_{n-1}(\mathbf{L}_G) = \sum_{a \in V} \det(\mathbf{L}_G(S_a, S_a)).$$

We will prove that for every  $a \in V$ ,

$$\det(\mathbf{L}_G(S_a, S_a)) = \prod_{e \in E} w_e.$$

Write  $\mathbf{L}_G = \mathbf{U}^T \mathbf{W} \mathbf{U}$ , where  $\mathbf{U}$  is the signed edge-vertex adjacency matrix and  $\mathbf{W}$  is the diagonal matrix of edge weights. Write  $\mathbf{B} = \mathbf{W}^{1/2} \mathbf{U}$ , so

$$\mathbf{L}_G(S_a, S_a) = \mathbf{B}(:, S_a)^T \mathbf{B}(:, S_a),$$

and

$$\det(\mathbf{L}_G(S_a, S_a)) = \det(\mathbf{B}(:, S_a))^2,$$

where we note that  $\mathbf{B}(:, S_a)$  is square because a tree has  $n - 1$  edges and so  $\mathbf{B}$  has  $n - 1$  rows.

To see what is going on, first consider the case in which  $G$  is a weighted path and  $a$  is the first vertex. Then,

$$\mathbf{U} = \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & -1 \end{pmatrix}, \quad \text{and} \quad \mathbf{B}(:, S_1) = \begin{pmatrix} -\sqrt{w_1} & 0 & \cdots & 0 \\ \sqrt{w_2} & -\sqrt{w_2} & \cdots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \cdots & -\sqrt{w_{n-1}} \end{pmatrix}.$$

We see that  $\mathbf{B}(:, S_1)$  is a lower-triangular matrix, and thus its determinant is the product of its diagonal entries,  $-\sqrt{w_i}$ .

To see that the same happens for every tree, renumber the vertices (permute the columns) so that  $a$  comes first, and that the other vertices are ordered by increasing distance from 1, breaking ties arbitrarily. This permutations can change the sign of the determinant, but we do not care because we are going to square it. For every vertex  $c \neq 1$ , the tree now has exactly one edge  $(b, c)$  with  $b < c$ . Put such an edge in position  $c - 1$  in the ordering, and let  $w_c$  indicate its weight. Now, when we remove the first column to form  $\mathbf{B}(:, S_1)$ , we produce a lower triangular matrix with the entry  $-\sqrt{w_c}$  on the  $c$ th diagonal. So, its determinant is the product of these terms and

$$\det(\mathbf{B}(:, S_a))^2 = \prod_{c=2}^n w_c.$$

□

*Proof of Theorem 14.5.1.* As in the previous lemma, let  $\mathbf{L}_G = \mathbf{U}^T \mathbf{W} \mathbf{U}$  and  $\mathbf{B} = \mathbf{W}^{1/2} \mathbf{U}$ . So,

$$\begin{aligned}\sigma_{n-1}(\mathbf{L}_G) &= \sigma_{n-1}(\mathbf{B}^T \mathbf{B}) \\ &= \sigma_{n-1}(\mathbf{B} \mathbf{B}^T) \\ &= \sum_{|S|=n-1, S \subseteq E} \sigma_{n-1}(\mathbf{B}(S,:) \mathbf{B}(S,:)^T) \quad (\text{by (14.3)}) \\ &= \sum_{|S|=n-1, S \subseteq E} \sigma_{n-1}(\mathbf{B}(S,:)^T \mathbf{B}(S,:)) \\ &= \sum_{|S|=n-1, S \subseteq E} \sigma_{n-1}(\mathbf{L}_{G_S}),\end{aligned}$$

where by  $G_S$  we mean the graph containing just the edges in  $S$ . As  $S$  contains  $n - 1$  edges, this graph is either disconnected or a tree. If it is disconnected, then its Laplacian has at least two zero eigenvalues and  $\sigma_{n-1}(\mathbf{L}_{G_S}) = 0$ . If it is a tree, we apply the previous lemma. Thus, the sum equals

$$\sum_{\text{spanning trees } T \subseteq E} \sigma_{n-1}(\mathbf{L}_{G_T}) = n \sum_{\text{spanning trees } T} \prod_{e \in T} w_e.$$

□

## 14.6 Leverage Scores and Marginal Probabilities

The *leverage score* of an edge, written  $\ell_e$  is defined to be  $w_e R_{\text{eff}}(e)$ . That is, the weight of the edge times the effective resistance between its endpoints. The leverage score serves as a measure of how important the edge is. For example, if removing an edge disconnects the graph, then  $R_{\text{eff}}(e) = 1/w_e$ , as all current flowing between its endpoints must use the edge itself, and  $\ell_e = 1$ .

Consider sampling a random spanning tree with probability proportional to the product of the weights of its edges. We will now show that the probability that edge  $e$  appears in the tree is exactly its leverage score.

**Theorem 14.6.1.** *If we choose a spanning tree  $T$  with probability proportional to the product of its edge weights, then for every edge  $e$*

$$\Pr[e \in T] = \ell_e.$$

For simplicity, you might want to begin by thinking about the case where all edges have weight 1.

Recall that the effective resistance of edge  $e = (a, b)$  is

$$(\delta_a - \delta_b)^T \mathbf{L}_G^+ (\delta_a - \delta_b),$$

and so

$$\ell_{a,b} = w_{a,b} (\delta_a - \delta_b)^T \mathbf{L}_G^+ (\delta_a - \delta_b).$$

We can write a matrix  $\Gamma$  that has all these terms on its diagonal by letting  $\mathbf{U}$  be the edge-vertex adjacency matrix,  $\mathbf{W}$  be the diagonal edge weight matrix,  $\mathbf{B} = \mathbf{W}^{1/2} \mathbf{U}$ , and setting

$$\Gamma = \mathbf{B} \mathbf{L}_G^+ \mathbf{B}^T.$$

The rows and columns of  $\Gamma$  are indexed by edges, and for each edge  $e$ ,

$$\Gamma(e, e) = \ell_e.$$

For off-diagonal entries corresponding to edges  $(a, b)$  and  $(c, d)$ , we have

$$\Gamma((a, b), (c, d)) = \sqrt{w_{a,b}} \sqrt{w_{c,d}} (\delta_a - \delta_b)^T \mathbf{L}_G^+ (\delta_c - \delta_d).$$

**Claim 14.6.2.** *The matrix  $\Gamma$  is a symmetric projection matrix and has trace  $n - 1$ .*

*Proof.* The matrix  $\Gamma$  is clearly symmetric. To show that it is a projection, it suffices to show that all of its eigenvalues are 0 or 1. This is true because, excluding the zero eigenvalues,  $\Gamma$  has the same eigenvalues as

$$\mathbf{L}_G^+ \mathbf{B}^T \mathbf{B} = \mathbf{L}_G^+ \mathbf{L}_G = \mathbf{\Pi},$$

where  $\mathbf{\Pi}$  is the projection orthogonal to the all 1 vector. As  $\mathbf{\Pi}$  has  $n - 1$  eigenvalues that are 1, so does  $\Gamma$ .  $\square$

As the trace of  $\Gamma$  is  $n - 1$ , so is the sum of the leverage scores:

$$\sum_e \ell_e = n - 1.$$

This is a good sanity check on Theorem 14.6.1: every spanning tree has  $n - 1$  edges, and thus the probabilities that each edge is in the tree must sum to  $n - 1$ .

We also obtain another formula for the leverage score. As a symmetric projection is its own square,

$$\Gamma(e, e) = \Gamma(e, :) \Gamma(e, :)^T = \|\Gamma(e, :)\|^2.$$

This is the formula I introduced in Section 14.2. If we flow 1 unit from  $a$  to  $b$ , the potential difference between  $c$  and  $d$  is  $(\delta_a - \delta_b)^T \mathbf{L}_G^+ (\delta_c - \delta_d)$ . If we plug these potentials into the Laplacian quadratic form, we obtain the effective resistance. Thus this formula says

$$w_{a,b} R_{\text{eff } a,b} = w_{a,b} \sum_{(c,d) \in E} w_{c,d} ((\delta_a - \delta_b)^T \mathbf{L}_G^+ (\delta_c - \delta_d))^2.$$

*Proof of Theorem 14.6.1.* Let  $\text{Span}(G)$  denote the set of spanning trees of  $G$ . For an edge  $e$ ,

$$\begin{aligned} \Pr_T [e \in T] &= \sum_{T \in \text{Span}(G): e \in T} \frac{\sigma_{n-1}(\mathbf{L}_{G_T})}{\sigma_{n-1}(\mathbf{L}_G)} \\ &= \sum_{T \in \text{Span}(G): e \in T} \sigma_{n-1}(\mathbf{L}_{G_T}) \sigma_{n-1}(\mathbf{L}_G^+) \\ &= \sum_{T \in \text{Span}(G): e \in T} \sigma_{n-1}(\mathbf{L}_{G_T} \mathbf{L}_G^+), \end{aligned}$$

by (14.4). Recalling that the subsets of  $n - 1$  edges that are not spanning trees contribute 0 allows us to re-write this sum as

$$\sum_{|S|=n-1, e \in S} \sigma_{n-1}(\mathbf{L}_{G_S} \mathbf{L}_G^+).$$

To evaluate the terms in the sum, we compute

$$\begin{aligned} \sigma_{n-1}(\mathbf{L}_{G_S} \mathbf{L}_G^+) &= \sigma_{n-1}(\mathbf{B}(:, S) \mathbf{B}(:, S)^T \mathbf{L}_G^+) \\ &= \sigma_{n-1}(\mathbf{B}(:, S)^T \mathbf{L}_G^+ \mathbf{B}(:, S)) \\ &= \sigma_{n-1}(\mathbf{\Gamma}(S, S)) \\ &= \sigma_{n-1}(\mathbf{\Gamma}(S, :) \mathbf{\Gamma}(:, S)). \end{aligned}$$

Let  $\boldsymbol{\gamma}_e = \mathbf{\Gamma}(e, :)$  and let  $\mathbf{\Pi}_{\boldsymbol{\gamma}_e}$  denote the projection orthogonal to  $\boldsymbol{\gamma}_e$ . As  $e \in S$ , we have

$$\sigma_{n-1}(\mathbf{\Gamma}(S, :) \mathbf{\Gamma}(:, S)) = \|\boldsymbol{\gamma}_e\|^2 \sigma_{n-2}(\mathbf{\Gamma}(S, :) \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma}(:, S)) = \|\boldsymbol{\gamma}_e\|^2 \sigma_{n-2}((\mathbf{\Gamma} \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma})(S, S)).$$

As  $\boldsymbol{\gamma}_e$  is in the span on  $\mathbf{\Gamma}$ , the matrix  $\mathbf{\Gamma} \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma}$  is a symmetric projection onto an  $n - 2$  dimensional space, and so

$$\sigma_{n-2}(\mathbf{\Gamma} \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma}) = 1.$$

To exploit this identity, we return to our summation:

$$\begin{aligned} \sum_{|S|=n-1, e \in S} \sigma_{n-1}(\mathbf{L}_{G_S} \mathbf{L}_G^+) &= \sum_{|S|=n-1, e \in S} \|\boldsymbol{\gamma}_e\|^2 \sigma_{n-2}((\mathbf{\Gamma} \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma})(S, S)) \\ &= \|\boldsymbol{\gamma}_e\|^2 \sum_{|S|=n-1, e \in S} \sigma_{n-2}((\mathbf{\Gamma} \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma})(S, S)) \\ &= \|\boldsymbol{\gamma}_e\|^2 \sigma_{n-2}(\mathbf{\Gamma} \mathbf{\Pi}_{\boldsymbol{\gamma}_e} \mathbf{\Gamma}) \\ &= \|\boldsymbol{\gamma}_e\|^2 \\ &= \ell_e. \end{aligned}$$

□

## 14.7 Quickly estimating effective resistances

We can compute  $R_{\text{eff}}(a, b)$  by solving a system of equations in  $\mathbf{L}$ . We know how to solve such systems of linear equations to high accuracy in time nearly linear in the number of nonzero entries in  $\mathbf{L}$  [?]. But, what if we want to know the effective resistance of every edge or between every pair of vertices?

We will see that we can do this by solving on  $O(\log n)$  systems of equations in  $\mathbf{L}$ . The reason is that the effective resistances are the squares of Euclidean distances:

$$R_{\text{eff}}(a, b) = \left\| \mathbf{L}_G^{+/-2} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \right\|^2 = \left\| \mathbf{L}_G^{+/-2} \boldsymbol{\delta}_a - \mathbf{L}_G^{+/-2} \boldsymbol{\delta}_b \right\|^2.$$

The reason is that we can exploit the Johnson-Lindenstrauss Theorem [?].

**Theorem 14.7.1.** [Johnson Lindenstrauss] Let  $\mathbf{v}_1, \dots, \mathbf{v}_n$  be vectors in an  $m$  dimensional vector space. Let  $\mathbf{R}$  be a  $d$ -by- $m$  matrix of independent Gaussian random variables of variance  $1/d$ . If

$$d \geq \frac{8}{\delta^2} \ln(n/\epsilon),$$

then with probability at most  $\epsilon$  for all  $i \neq j$ ,

$$1 - \delta \leq \frac{\|\mathbf{R}\mathbf{v}_i - \mathbf{R}\mathbf{v}_j\|}{\|\mathbf{v}_i - \mathbf{v}_j\|} \leq 1 + \delta.$$

That is, the distances between all pairs of vectors  $\mathbf{R}\mathbf{v}_i$  are approximately the same as between the vectors  $\mathbf{v}_i$ .

One can prove this by using tail bounds on  $\chi$ -square random variables. I'll include a proof in the Appendix.

Here's one way we could try to use this. If we want to estimate all effective resistances to within error  $\delta$ , with probability at least  $1 - \epsilon$ , we set

$$d = \left\lceil \frac{8}{\delta^2} \ln(n/\epsilon) \right\rceil,$$

choose  $\mathbf{R}$  to be a  $d$ -by- $n$  matrix of independent random Gaussians, and then compute

$$\mathbf{R}\mathbf{L}^{+/2}.$$

This requires solving  $d$  systems of linear equations in  $\mathbf{L}^{1/2}$ .

But, that is not quite the same as solving systems in  $\mathbf{L}$ . To turn this into a problem of solving systems in  $\mathbf{L}$ , we exploit a slightly different formula for effective resistances. As before, write  $\mathbf{L} = \mathbf{U}^T \mathbf{W} \mathbf{U}$ . We then have

$$\mathbf{L}^+ \mathbf{U}^T \mathbf{W}^{1/2} \mathbf{W}^{1/2} \mathbf{U} \mathbf{L}^+ = \mathbf{L}^+ \mathbf{L} \mathbf{L}^+ = \mathbf{L}^+.$$

So,

$$\left\| \mathbf{W}^{1/2} \mathbf{U} \mathbf{L}^+ (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \right\|^2 = R_{\text{eff}}(a, b).$$

Now, we let  $\mathbf{R}$  be a  $d$ -by- $|E|$  matrix of random Gaussians of variance  $1/d$ , and compute

$$\mathbf{R} \mathbf{W}^{1/2} \mathbf{U} \mathbf{L}^+ = (\mathbf{R} \mathbf{W}^{1/2} \mathbf{U}) \mathbf{L}^+ \boldsymbol{\delta}_a.$$

This requires solving  $d$  systems of linear equations in  $\mathbf{L}$ . We then set

$$\mathbf{v}_a = (\mathbf{R} \mathbf{W}^{1/2} \mathbf{U}) \mathbf{L}^+ \boldsymbol{\delta}_a.$$

Each of these is a vector in  $d$  dimensions, and with high probability  $\|\mathbf{v}_a - \mathbf{v}_b\|^2$  is a good approximation of  $R_{\text{eff}}(a, b)$ .

## 14.8 Monotonicity

Rayleigh's Monotonicity Principle tells us that if we alter the spring network by decreasing some of the spring constants, then the effective spring constant between  $s$  and  $t$  will not increase. In terms of effective resistance, this says that if we increase the resistance of some resistors then the effective resistance can not decrease. This sounds obvious. But, it is in fact a very special property of linear elements like springs and resistors.

**Theorem 14.8.1.** *Let  $G = (V, E, w)$  be a weighted graph and let  $\hat{G} = (V, E, \hat{w})$  be another weighted graph with the same edges and such that*

$$\hat{w}_{a,b} \leq w_{a,b}$$

*for all  $(a, b) \in E$ . For vertices  $s$  and  $t$ , let  $c_{s,t}$  be the effective spring constant between  $s$  and  $t$  in  $G$  and let  $\hat{c}_{s,t}$  be the analogous quantity in  $\hat{G}$ . Then,*

$$\hat{c}_{s,t} \leq c_{s,t}.$$

*Proof.* Let  $\mathbf{x}$  be the vector of minimum energy in  $G$  such that  $\mathbf{x}(s) = 0$  and  $\mathbf{x}(t) = 1$ . Then, the energy of  $\mathbf{x}$  in  $\hat{G}$  is no greater:

$$\frac{1}{2} \sum_{(a,b) \in E} \hat{w}_{a,b} (\mathbf{x}(a) - \mathbf{x}(b))^2 \leq \frac{1}{2} \sum_{(a,b) \in E} w_{a,b} (\mathbf{x}(a) - \mathbf{x}(b))^2 = c_{s,t}.$$

So, the minimum energy of a vector  $\mathbf{x}$  in  $\hat{G}$  such that  $\mathbf{x}(s) = 0$  and  $\mathbf{x}(t) = 1$  will be at most  $c_{s,t}$ , and so  $\hat{c}_{s,t} \leq c_{s,t}$ .  $\square$

While this principle seems very simple and intuitively obvious, it turns out to fail in just slightly more complicated situations.

## 14.9 Notes

### A Proof of Johnson Lindenstrauss

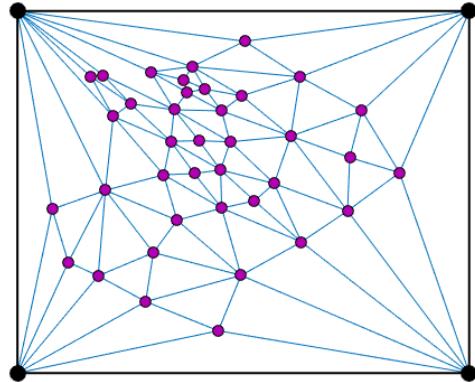
## Tutte's Theorem: How to draw a graph

Daniel A. Spielman

October 22, 2018

## 15.1 Overview

We prove Tutte's theorem [Tut63], which shows how to use spring embeddings to obtain planar drawings of 3-connected planar graphs. One begins by selecting a face, and then nailing down the positions of its vertices to the corners of a strictly convex polygon. Of course, the edges of the face should line up with the edges of the polygon. Every other vertex goes where the springs say they should—to the center of gravity of their neighbors. Tutte proved that the result is a planar embedding of the planar graph. Here is an image of such an embedding



The presentation in this lecture is based on notes given to me by Jim Geelen. I begin by recalling some standard results about planar graphs that we will assume.

## 15.2 3-Connected, Planar Graphs

A graph  $G = (V, E)$  is  $k$ -connected if there is no set of  $k - 1$  vertices whose removal disconnects the graph. That is, for every  $S \subset V$  with  $|S| \geq |V| - (k - 1)$ ,  $G(S)$  is connected. In a classical graph theory course, one usually spends a lot of time studying things like 3-connectivity.

A *planar drawing* of a graph  $G = (V, E)$  consists of mapping from the vertices to the plane,  $\mathbf{z} : V \rightarrow \mathbb{R}^2$ , along with interior-disjoint curves for each edge. The curve for edge  $(a, b)$  starts at  $\mathbf{z}(a)$ , ends at  $\mathbf{z}(b)$ , never crosses itself, and its interior does not intersect the curve for any other edge. A graph is *planar* if it has a planar drawing. There can, of course, be many planar drawings of a graph.

If one removes the curves corresponding to the edges in a planar drawing, one divides the plane into connected regions called *faces*. In a 3-connected planar graph, the sets of vertices and edges that border each face are the same in every planar drawing. There are planar graphs that are not 3-connected, like those in Figures 15.2 and 15.2, in which different planar drawings result in combinatorially different faces. We will only consider 3-connected planar graphs.

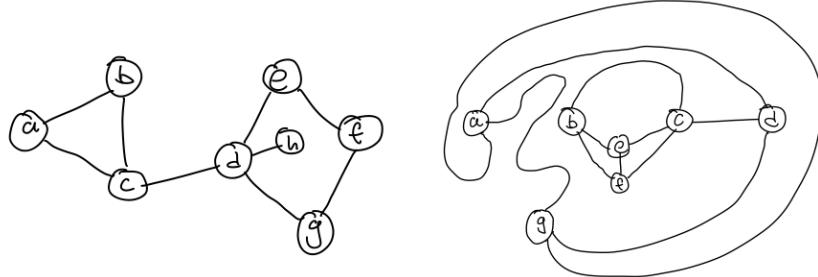


Figure 15.1: Planar graphs that are merely one-connected. Edge  $(c, d)$  appears twice on a face in each of them.

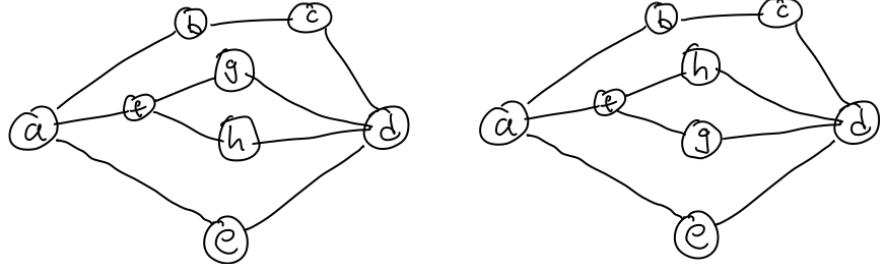


Figure 15.2: Two different planar drawings of a planar graph that is merely two-connected. Vertices  $g$  and  $h$  have switched positions, and thus appear in different faces in each drawing.

We state a few properties of 3-connected planar graphs that we will use. We will not prove these properties, as we are more concerned with algebra and these properly belong in a class on combinatorial graph theory.

**Claim 15.2.1.** *Let  $G = (V, E)$  be a planar graph. Then, there exists a set of faces  $F$ , each of which corresponds to a cycle in  $G$ , so that no vertex appears twice in a face, no edge appears twice in a face, and every edge appears in exactly two faces.*

We call the face on the outside of the drawing the *outside face*. The edges that lie along the outside face are the *boundary edges*.

Another standard fact about planar graphs is that they remain planar under edge contractions. Contracting an edge  $(a, b)$  creates a new graph in which  $a$  and  $b$  become the same vertex, and all edges that went from other vertices to  $a$  or  $b$  now go to the new vertex. Contractions also preserve

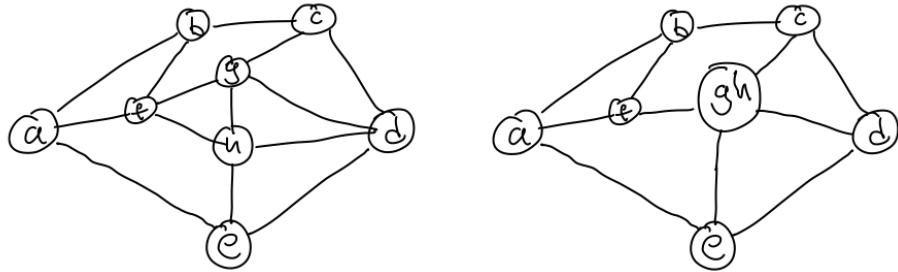


Figure 15.3: 3-connected planar graphs. Some faces of the graph on the left are  $abf$ ,  $fhg$ , and  $afhe$ . The outer face is  $abcde$ . The graph on the right is obtained by contracting edge  $(g, h)$ .

3-connectivity. Figure 15.2 depicts a 3-connected planar graph and the result of contracting an edge.

A graph  $H = (W, F)$  is a minor of a graph  $G = (V, E)$  if  $H$  can be obtained from  $G$  by contracting some edges and possibly deleting other edges and vertices. This means that each vertex in  $W$  corresponds to a connected subset of vertices in  $G$ , and that there is an edge between two vertices in  $W$  precisely when there is some edge between the two corresponding subsets. This leads to Kuratowski's Theorem [Kur30], one of the most useful characterizations of planar graphs.

**Theorem 15.2.2.** *A graph  $G$  is planar if and only if it does not have a minor isomorphic to the complete graph on 5 vertices,  $K_5$ , or the bipartite complete graph between two sets of 3 vertices,  $K_{3,3}$ .*

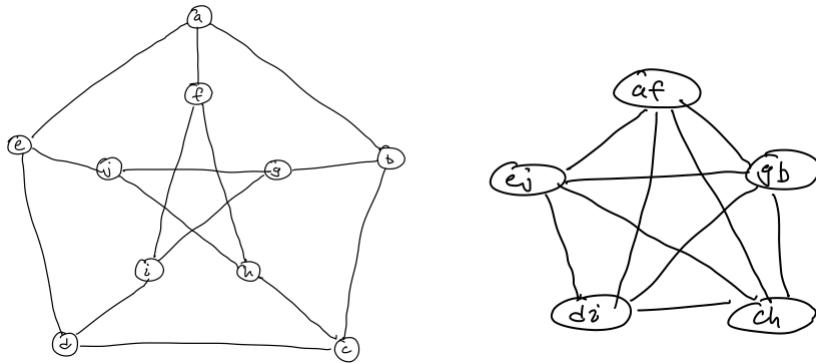


Figure 15.4: The Peterson graph appears on the left. On the right is a minor of the Peterson graph that is isomorphic to  $K_5$ , proving that the Peterson graph is not planar.

We will use one other important fact about planar graphs, whose utility in this context was observed by Jim Geelen.

**Lemma 15.2.3.** *Let  $(a, b)$  be an edge of a 3-connected planar graph and let  $S_1$  and  $S_2$  be the sets of vertices on the two faces containing  $(a, b)$ . Let  $P$  be a path in  $G$  that starts at a vertex of  $S_1 - \{a, b\}$ ,*

ends at a vertex of  $S_2 - \{a, b\}$ , and that does not intersect  $a$  or  $b$ . Then, every path in  $G$  from  $a$  to  $b$  either intersects a vertex of  $P$  or the edge  $(a, b)$ .

*Proof.* Let  $s_1$  and  $s_2$  be the vertices at the ends of the path  $P$ . Consider a planar drawing of  $G$  and the closed curve in the plane that follows the path  $P$  from  $s_1$  to  $s_2$ , and then connects  $s_1$  to  $s_2$  by moving inside the faces  $S_1$  and  $S_2$ , where the path only intersects the curve for edge  $(a, b)$ . This curve separates vertex  $a$  from vertex  $b$ . Thus, every path in  $G$  that connects  $a$  to  $b$  must intersect this curve. This means that it must either consist of just edge  $(a, b)$ , or it must intersect a vertex of  $P$ . See Figure 15.2.  $\square$

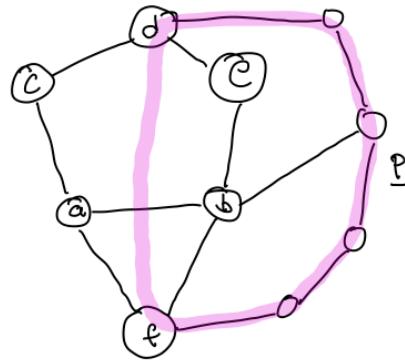


Figure 15.5: A depiction of Lemma 15.2.3.  $S_1 = abcde$ ,  $S_2 = abf$ , and the path  $P$  starts at  $d$ , ends at  $f$ , and contains the other unlabeled vertices.

### 15.3 Strictly Convex Polygons

This is a good time to remind you what exactly a convex polygon is. A subset  $C \subseteq \mathbb{R}^2$  is convex if for every two points  $x$  and  $y$  in  $C$ , the line segment between  $x$  and  $y$  is also in  $C$ . A convex polygon is a convex region of  $\mathbb{R}^2$  whose boundary is comprised of a finite number of straight lines. It is *strictly* convex if in addition the angle at every corner is less than  $\pi$ . We will always assume that the corners of a strictly convex polygon are distinct. Two corners form an edge of the polygon if the interior of the polygon is entirely on one side of the line through those corners. This leads to another definition of a strictly convex polygon: a convex polygon is strictly convex if for every edge, all of the corners of the polygon other than those two defining the edge lie entirely on one side of the polygon. In particular, none of the other corners lie on the line.

**Definition 15.3.1.** Let  $G = (V, E)$  be a 3-connected planar graph. We say that  $z : V \rightarrow \mathbb{R}^2$  is a Tutte embedding if

- a. There is a face  $F$  of  $G$  such that  $z$  maps the vertices of  $F$  to the corners of a strictly convex polygon so that every edge of the face joins consecutive corners of the polygon;

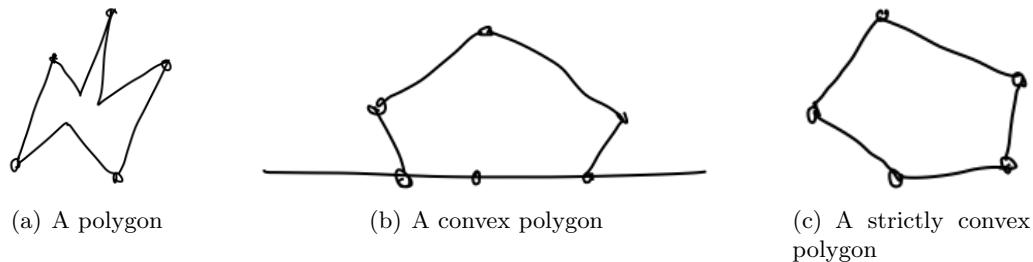


Figure 15.6: Polygons

b. Every vertex not in  $F$  lies at the center of gravity of its neighbors.

We will prove Tutte's theorem by proving that every face of  $G$  is embedded as a strictly convex polygon. In fact, we will not use the fact that every non-boundary vertex is exactly the average of its neighbors. We will only use the fact that every non-boundary vertex is inside the convex hull of its neighbors. This corresponds to allowing arbitrary spring constants in the embedding.

**Theorem 15.3.2.** Let  $G = (V, E)$  be a 3-connected planar graph, and let  $z$  be a Tutte embedding of  $G$ . If we represent every edge of  $G$  as the straight line between the embedding of its endpoints, then we obtain a planar drawing of  $G$ .

Note that if the graph were not 3-connected, then the embedding could be rather degenerate. If there are two vertices  $a$  and  $b$  whose removal disconnects the graph into two components, then all of the vertices in one of those components will embed on the line segment from  $a$  to  $b$ .

Henceforth,  $G$  will always be a 3-connected planar graph and  $z$  will always be a Tutte embedding.

## 15.4 Possible Degeneracies

The proof of Theorem 15.3.2 will be easy once we rule out certain degeneracies. There are two types of degeneracies that we must show can not happen. The most obvious is that we can not have  $z(a) = z(b)$  for any edge  $(a, b)$ . The fact that this degeneracy can not happen will be a consequence of Lemma 15.5.1.

The other type of degeneracy is when there is a vertex  $a$  such that all of its neighbors lie on one line in  $\mathbb{R}^2$ . We will rule out such degeneracies in this section.

We first observe two simple consequences of the fact that every vertex must lie at the average of its neighbors.

**Claim 15.4.1.** Let  $a$  be a vertex and let  $\ell$  be any line in  $\mathbb{R}^2$  through  $z(a)$ . If  $a$  has a neighbor that lies on one side of  $\ell$ , then it has a neighbor that lies on the other.

**Claim 15.4.2.** All vertices not in  $F$  must lie strictly inside the convex hull of the polygon of which the vertices in  $F$  are the corners.

*Proof.* For every vertex  $a$  not in  $F$ , we can show that the position of  $a$  is a weighted average of the positions of vertices in  $F$  by eliminating every vertex not in  $F \cup \{a\}$ . As we learned in Lecture 13, this results in a graph in which all the neighbors of  $a$  are in  $F$ , and thus the position of  $a$  is some weighted average of the position of the vertices in  $F$ . As the graph is 3-connected, we can show that this average must assign nonzero weights to at least 3 of the vertices in  $F$ .  $\square$

Note that it is also possible to prove Claim 15.4.2 by showing that one could reduce the potential energy by moving vertices inside the polygon. See Claim 8.8.1 from my lecture notes from 2015.

**Lemma 15.4.3.** *Let  $H$  be a halfspace in  $\mathbb{R}^2$  (that is, everything on one side of some line). Then the subgraph of  $G$  induced on the vertices  $a$  such that  $\mathbf{z}(a) \in H$  is connected.*

*Proof.* Let  $\mathbf{t}$  be a vector so that we can write the line  $\ell$  in the form  $\mathbf{t}^T \mathbf{x} = \mu$ , with the halfspace consisting of those points  $\mathbf{x}$  for which  $\mathbf{t}^T \mathbf{x} \geq \mu$ . Let  $a$  be a vertex such that  $\mathbf{z}(a) \in H$  and let  $b$  be a vertex that maximizes  $\mathbf{t}^T \mathbf{z}(b)$ . So,  $\mathbf{z}(b)$  is as far from the line defining the halfspace as possible. By Claim 15.4.2,  $b$  must be on the outside face,  $F$ .

For every vertex  $c$ , define  $t(c) = \mathbf{t}^T \mathbf{z}(c)$ . We will see that there is a path in  $G$  from  $a$  to  $b$  along which the function  $t$  never decreases, and thus all the vertices along the path lie in the halfspace. We first consider the case in which  $t(a) = t(b)$ . In this case, we also know that  $a \in F$ . As the vertices in  $F$  embed to a strictly convex polygon, this implies that  $(a, b)$  is an edge of that polygon, and thus the path from  $a$  to  $b$ .

If  $t(a) < t(b)$ , it suffices to show that there is a path from  $a$  to some other vertex  $c$  for which  $t(c) > t(a)$  and along which  $t$  never decreases: we can then proceed from  $c$  to obtain a path to  $b$ . Let  $U$  be the set of all vertices  $u$  reachable from  $a$  for which  $t(u) = t(a)$ . As the graph is connected, there must be a vertex  $u \in U$  that has a neighbor  $c \notin U$ . By Claim 15.4.1  $u$  must have a neighbor  $c$  for which  $t(c) > t(u)$ . Thus, the a path from  $a$  through  $U$  to  $c$  suffices.  $\square$

**Lemma 15.4.4.** *No vertex is colinear with all of its neighbors.*

*Proof.* This is trivially true for vertices in  $F$ , as no three of them are colinear.

Assume by way of contradiction that there is a vertex  $a$  that is colinear with all of its neighbors. Let  $\ell$  be that line, and let  $S^+$  and  $S^-$  be all the vertices that lie above and below the line, respectively. Lemma 15.4.3 tells us that both sets  $S^+$  and  $S^-$  are connected. Let  $U$  be the set of vertices  $u$  reachable from  $a$  and such that all of  $u$ 's neighbors lie on  $\ell$ . The vertex  $a$  is in  $U$ . Let  $W$  be the set of nodes that lie on  $\ell$  that are neighbors of vertices in  $U$ , but which themselves are not in  $U$ . As vertices in  $W$  are not in  $U$ , Claim 15.4.1 implies that each vertex in  $W$  has neighbors in both  $S^+$  and  $S^-$ . As the graph is 3-connected, and removing the vertices in  $W$  would disconnect  $U$  from the rest of the graph, there are at least 3 vertices in  $W$ . Let  $w_1, w_2$  and  $w_3$  be three of the vertices in  $W$ .

We will now obtain a contradiction by showing that  $G$  has a minor isomorphic to  $K_{3,3}$ . The three vertices on one side are  $w_1, w_2$ , and  $w_3$ . The other three are obtained by contracting the vertex sets  $S^+$ ,  $S^-$ , and  $U$ .  $\square$

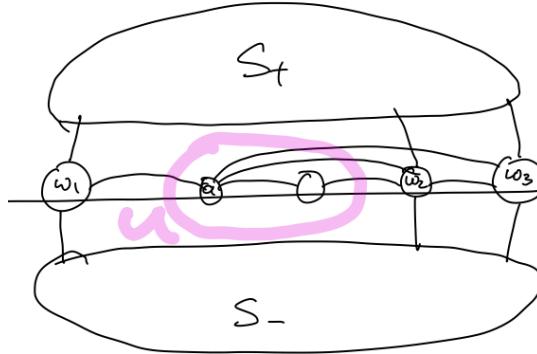


Figure 15.7: An illustration of the proof of Lemma 15.4.4.

## 15.5 All faces are convex

We now prove that every face of  $G$  embeds as a strictly convex polygon.

**Lemma 15.5.1.** *Let  $(a, b)$  be any non-boundary edge of the graph, and let  $\ell$  be a line through  $z(a)$  and  $z(b)$  (there is probably just one). Let  $F_0$  and  $F_1$  be the faces that border edge  $(a, b)$  and let  $S_0$  and  $S_1$  be the vertices on those faces, other than  $a$  and  $b$ . Then all the vertices of  $S_0$  and  $S_1$  lie on opposite sides of  $\ell$ , and none lie on  $\ell$ .*

Note: if  $z(a) = z(b)$ , then we can find a line passing through them and one of the vertices of  $S_0$ . This leads to a contradiction, and thus rules out this type of degeneracy.

*Proof.* Assume by way of contradiction that the lemma is false. Without loss of generality, we may then assume that there are vertices of both  $S_0$  and  $S_1$  on or below the line  $\ell$ . Let  $s_0$  and  $s_1$  be such vertices. By Lemma 15.4.4 and Claim 15.4.1, we know that both  $s_0$  and  $s_1$  have neighbors that lie strictly below the line  $\ell$ . By Lemma 15.4.3, we know that there is a path  $P$  that connects  $s_0$  and  $s_1$  on which all vertices other than  $s_0$  and  $s_1$  lie strictly below  $\ell$ .

On the other hand, we can similarly show that both  $a$  and  $b$  have neighbors above the line  $\ell$ , and that they are joined by a path that lies strictly above  $\ell$ . Thus, this path cannot consist of the edge  $(a, b)$  and must be disjoint from  $P$ . This contradicts Lemma 15.2.3.  $\square$

So, we now know that the embedding  $z$  contains no degeneracies, that every face is embedded as a strictly convex polygon, and that the two faces bordering each edge embed on opposite sides of that edge. This is all we need to know to prove Tutte's Theorem. We finish the argument in the proof below.

*Proof of Theorem 15.3.2.* We say that a point of the plane is *generic* if it does not lie on any  $z(a)$  for on any segment of the plane corresponding to an edge  $(a, b)$ . We first prove that every generic point lies in exactly one face of  $G$ .

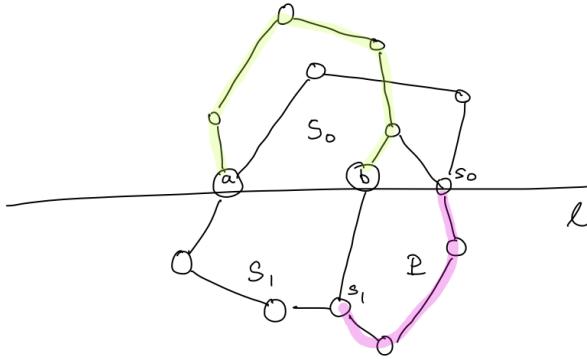


Figure 15.8: An illustration of the proof of Lemma 15.5.1.

Begin with a point that is outside the polygon on which  $F$  is drawn. Such a point lies only in the outside face. For any other generic point we can draw a curve between these points that never intersects a  $\alpha(a)$  and never crosses the intersection of the drawings of edges. That is, it only crosses drawings of edges in their interiors. By Lemma 15.5.1, when the curve does cross such an edge it moves from one face to another. So, at no point does it ever appear in two faces.

Now, assume by way of contradiction that the drawings of two edges cross. There must be some generic point near their intersection that lies in at least two faces. This would be a contradiction.  $\square$

## 15.6 Notes

This is the simplest proof of Tutte's theorem that I have seen. Over the years, I have taught many versions of Tutte's proof by building on expositions by Lovász [LV99] and Geelen [Gee12], and an alternative proof of Gortler, Gotsman and Thurston [GGT06].

## References

- [Gee12] Jim Geelen. On how to draw a graph. <https://www.math.uwaterloo.ca/~jfgeelen/Publications/tutte.pdf>. 2012.
- [GGT06] Steven J Gortler, Craig Gotsman, and Dylan Thurston. Discrete one-forms on meshes and applications to 3d mesh parameterization. *Computer Aided Geometric Design*, 23(2):83–112, 2006.
- [Kur30] Casimir Kuratowski. Sur le probleme des courbes gauches en topologie. *Fundamenta mathematicae*, 15(1):271–283, 1930.
- [LV99] László Lovász and Katalin Vesztergombi. Geometric representations of graphs. *Paul Erdos and his Mathematics*, 1999.

- [Tut63] W. T. Tutte. How to draw a graph. *Proc. London Mathematical Society*, 13:743–768, 1963.

## The Second Eigenvalue of Planar Graphs

Daniel A. Spielman

October 24, 2018

## 16.1 Overview

Spectral Graph theory first came to the attention of many because of the success of using the second Laplacian eigenvector to partition planar graphs and scientific meshes [DH72, DH73, Bar82, PSL90, Sim91].

In this lecture, we will attempt to explain this success by proving, at least for planar graphs, that the second smallest Laplacian eigenvalue is small. One can then use Cheeger's inequality to prove that the corresponding eigenvector provides a good cut.

This was already known for the model case of a 2-dimensional grid. If the grid is of size  $\sqrt{n}$ -by- $\sqrt{n}$ , then it has  $\lambda_2 \approx c/n$ . Cheeger's inequality then tells us that it has a cut of conductance  $c/\sqrt{n}$ . And, this is in fact the cut that goes right across the middle of one of the axes, which is the cut of minimum conductance.

**Theorem 16.1.1** ([ST07]). *Let  $G$  be a planar graph with  $n$  vertices of maximum degree  $d$ , and let  $\lambda_2$  be the second-smallest eigenvalue of its Laplacian. Then,*

$$\lambda_2 \leq \frac{8d}{n}.$$

The proof will involve almost no calculation, but will use some special properties of planar graphs. However, this proof has been generalized to many planar-like graphs, including the graphs of well-shaped 3d meshes.

## 16.2 Geometric Embeddings

We typically upper bound  $\lambda_2$  by evidencing a test vector. Here, we will upper bound  $\lambda_2$  by evidencing a test embedding. The bound we apply is:

**Lemma 16.2.1.** *For any  $d \geq 1$ ,*

$$\lambda_2 = \min_{\mathbf{v}_1, \dots, \mathbf{v}_n \in \mathbb{R}^d : \sum \mathbf{v}_i = \mathbf{0}} \frac{\sum_{(i,j) \in E} \|\mathbf{v}_i - \mathbf{v}_j\|^2}{\sum_i \|\mathbf{v}_i\|^2}. \quad (16.1)$$

*Proof.* Let  $\mathbf{v}_i = (x_i, y_i, \dots, z_i)$ . We note that

$$\sum_{(i,j) \in E} \|\mathbf{v}_i - \mathbf{v}_j\|^2 = \sum_{(i,j) \in E} (x_i - x_j)^2 + \sum_{(i,j) \in E} (y_i - y_j)^2 + \dots + \sum_{(i,j) \in E} (z_i - z_j)^2.$$

Similarly,

$$\sum_i \|\mathbf{v}_i\|^2 = \sum_i x_i^2 + \sum_i y_i^2 + \cdots + \sum_i z_i^2.$$

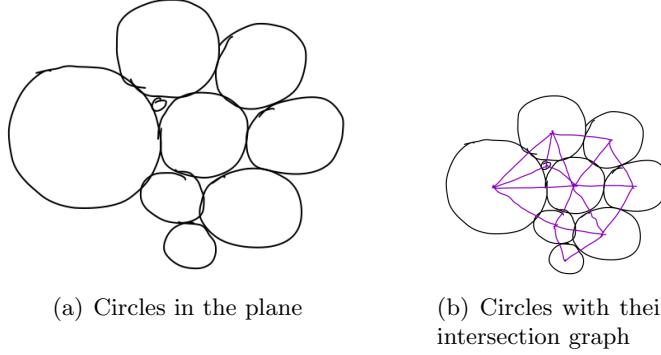
It is now trivial to show that  $\lambda_2 \geq RHS$ : just let  $x_i = y_i = \cdots = z_i$  be given by an eigenvector of  $\lambda_2$ . To show that  $\lambda_2 \leq RHS$ , we apply my favorite inequality:  $\frac{A+B+\cdots+C}{A'+B'+\cdots+C'} \geq \min\left(\frac{A}{A'}, \frac{B}{B'}, \dots, \frac{C}{C'}\right)$ , and then recall that  $\sum x_i = 0$  implies

$$\frac{\sum_{(i,j) \in E} (x_i - x_j)^2}{\sum_i x_i^2} \geq \lambda_2.$$

□

For an example, consider the natural embedding of the square with corners  $(\pm 1, \pm 1)$ .

The key to applying this embedding lemma is to obtain the right embedding of a planar graph. Usually, the right embedding of a planar graph is given by Koebe's embedding theorem, which I will now explain. I begin by considering one way of generating planar graphs. Consider a set of circles  $\{C_1, \dots, C_n\}$  in the plane such that no pair of circles intersects in their interiors. Associate a vertex with each circle, and create an edge between each pair of circles that meet at a boundary. See Figure 16.2. The resulting graph is clearly planar. Koebe's embedding theorem says that *every planar graph results from such an embedding*.



**Theorem 16.2.2** (Koebe). *Let  $G = (V, E)$  be a planar graph. Then there exists a set of circles  $\{C_1, \dots, C_n\}$  in  $\mathbb{R}^2$  that are interior-disjoint such that circle  $C_i$  touches circle  $C_j$  if and only if  $(i, j) \in E$ .*

This is an amazing theorem, which I won't prove today. You can find a beautiful proof in the book "Combinatorial Geometry" by Agarwal and Pach.

Such an embedding is often called a *kissing disk* embedding of the graph. From a kissing disk embedding, we obtain a natural choice of  $\mathbf{v}_i$ : the center of disk  $C_i$ . Let  $r_i$  denote the radius of this disk. We now have an easy upper bound on the numerator of (16.1):  $\|\mathbf{v}_i - \mathbf{v}_j\|^2 = (r_i + r_j)^2 \leq 2r_i^2 + 2r_j^2$ . On the other hand, it is trickier to obtain a lower bound on  $\sum \|\mathbf{v}_i\|^2$ . In fact, there are graphs whose kissing disk embeddings result in

$$(16.1) = \Theta(1).$$

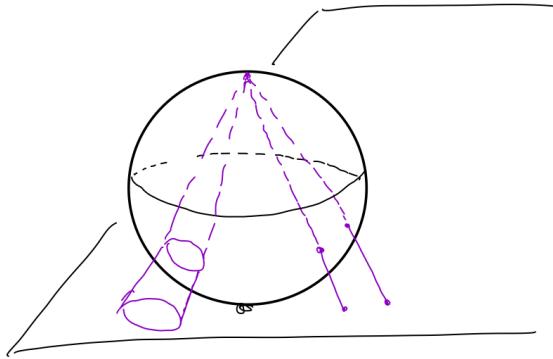
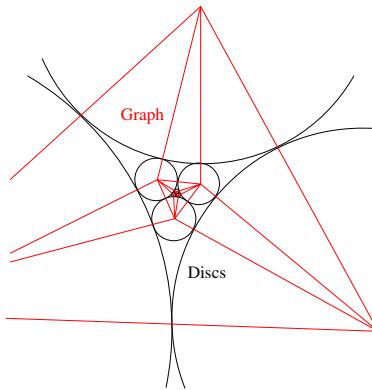


Figure 16.1: Stereographic Projection.

These graphs come from triangles inside triangles... Such a graph is depicted below:



We will fix this problem by lifting the planar embeddings to the sphere by stereographic projection. Given a plane,  $\mathbb{R}^2$ , and a sphere  $S$  tangent to the plane, we can define the stereographic projection map,  $\Pi$ , from the plane to the sphere as follows: let  $s$  denote the point where the sphere touches the plane, and let  $n$  denote the opposite point on the sphere. For any point  $x$  on the plane, consider the line from  $x$  to  $n$ . It will intersect the sphere somewhere. We let this point of intersection be  $\Pi(x)$ .

The fundamental fact that we will exploit about stereographic projection is that *it maps circles to circles!* So, by applying stereographic projection to a kissing disk embedding of a graph in the plane, we obtain a kissing disk embedding of that graph on the sphere. Let  $D_i = \Pi(C_i)$  denote the image of circle  $C_i$  on the sphere. We will now let  $v_i$  denote the center of  $D_i$ , on the sphere.

If we had  $\sum_i v_i = \mathbf{0}$ , the rest of the computation would be easy. For each  $i$ ,  $\|v_i\| = 1$ , so the denominator of (16.1) is  $n$ . Let  $r_i$  denote the straight-line distance from  $v_i$  to the boundary of  $D_i$ . We then have (see Figure 16.2)

$$\|v_i - v_j\|^2 \leq (r_i + r_j)^2 \leq 2r_i^2 + 2r_j^2.$$

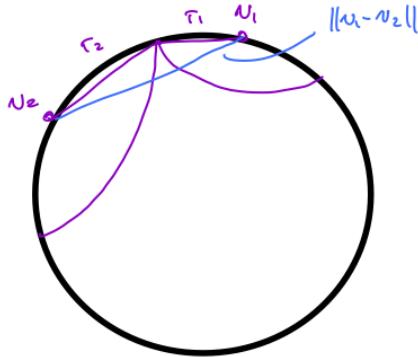


Figure 16.2: Stereographic Projection.

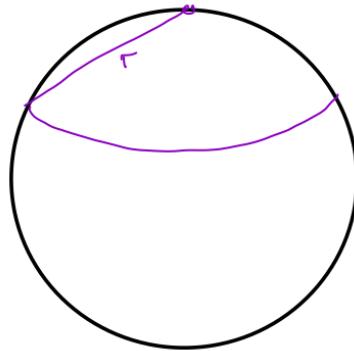


Figure 16.3: A Spherical Cap.

So, the denominator of (16.1) is at most  $2d \sum_i r_i^2$ . On the other hand, a theorem of Archimedes tells us that the area of the cap encircled by  $D_i$  is at exactly  $\pi r_i^2$ . Rather than proving it, I will convince you that it has to be true because it is true when  $r_i$  is small, it is true when the cap is a hemisphere and  $r_i = \sqrt{2}$ , and it is true when the cap is the whole sphere and  $r_i = 2$ .

As the caps are disjoint, we have

$$\sum_i \pi r_i^2 \leq 4\pi,$$

which implies that the denominator of (16.1) is at most

$$\sum_{(a,b) \in E} \|\mathbf{v}_a - \mathbf{v}_b\|^2 \leq 2r_a^2 + 2r_b^2 \leq 2d \sum_a r_a^2 \leq 8d.$$

Putting these inequalities together, we see that

$$\min_{\mathbf{v}_1, \dots, \mathbf{v}_n \in \mathbb{R}^d : \sum \mathbf{v}_i = \mathbf{0}} \frac{\sum_{(i,j) \in E} \|\mathbf{v}_i - \mathbf{v}_j\|^2}{\sum_i \|\mathbf{v}_i\|^2} \leq \frac{8d}{n}.$$

Thus, we merely need to verify that we can ensure that

$$\sum_i \mathbf{v}_i = \mathbf{0}. \quad (16.2)$$

Note that there is enough freedom in our construction to believe that we could prove such a thing: we can put the sphere anywhere on the plane, and we could even scale the image in the plane before placing the sphere. By carefully combining these two operations, it is clear that we can place the center of gravity of the  $\mathbf{v}_i$ s close to any point on the boundary of the sphere. It turns out that this is sufficient to prove that we can place it at the origin.

### 16.3 The center of gravity

We need a nice family of maps that transform our kissing disk embedding on the sphere. It is particularly convenient to parameterize these by a point  $\omega$  inside the sphere. For any point  $\alpha$  on the surface of the unit sphere, I will let  $\Pi_\alpha$  denote the stereographic projection from the plane tangent to the sphere at  $\alpha$ .

I will also define  $\Pi_\alpha^{-1}$ . To handle the point  $-\alpha$ , I let  $\Pi_\alpha^{-1}(-\alpha) = \infty$ , and  $\Pi_\alpha(\infty) = -\alpha$ . We also define the map that dilates the plane tangent to the sphere at  $\alpha$  by a factor  $a$ :  $D_\alpha^a$ . We then define the following map from the sphere to itself

$$f_\omega(\mathbf{x}) \stackrel{\text{def}}{=} \Pi_{\omega/\|\omega\|} \left( D_{\omega/\|\omega\|}^{1-\|\omega\|} \left( \Pi_{\omega/\|\omega\|}^{-1}(\mathbf{x}) \right) \right).$$

For  $\alpha \in S$  and  $\omega = a\alpha$ , this map pushes everything on the sphere to a point close to  $\alpha$ . As  $a$  approaches 1, the mass gets pushed closer and closer to  $\alpha$ .

Instead of proving that we can achieve (16.2), I will prove a slightly simpler theorem. The proof of the theorem we really want is similar, but about just a few minutes too long for class. We will prove

**Theorem 16.3.1.** *Let  $\mathbf{v}_1, \dots, \mathbf{v}_n$  be points on the unit-sphere. Then, there exists an  $\omega$  such that  $\sum_i f_\omega(\mathbf{v}_i) = \mathbf{0}$ .*

The reason that this theorem is different from the one that we want to prove is that if we apply a circle-preserving map from the sphere to itself, the center of the circle might not map to the center of the image circle.

To show that we can achieve  $\sum_i \mathbf{v}_i = \mathbf{0}$ , we will use the following topological lemma, which follows immediately from Brouwer's fixed point theorem. In the following, we let  $B$  denote the ball of points of norm less than 1, and  $S$  the sphere of points of norm 1.

**Lemma 16.3.2.** *If  $\phi : B \rightarrow B$  be a continuous map that is the identity on  $S$ . Then, there exists an  $\omega \in B$  such that*

$$\phi(\omega) = \mathbf{0}.$$

We will prove this lemma using Brouwer's fixed point theorem:

**Theorem 16.3.3** (Brouwer). *If  $g : B \rightarrow B$  is continuous, then there exists an  $\alpha \in B$  such that  $g(\alpha) = \alpha$ .*

*Proof of Lemma 16.3.2.* Let  $b$  be the map that sends  $\mathbf{z} \in B$  to  $\mathbf{z}/\|\mathbf{z}\|$ . The map  $b$  is continuous at every point other than  $\mathbf{0}$ . Now, assume by way of contradiction that  $\mathbf{0}$  is not in the image of  $\phi$ , and let  $g(\mathbf{z}) = -b(\phi(\mathbf{z}))$ . By our assumption,  $g$  is continuous and maps  $B$  to  $B$ . However, it is clear that  $g$  has no fixed point, contradicting Brouwer's fixed point theorem.  $\square$

Lemma 16.3.2, was our motivation for defining the maps  $f_\omega$  in terms of  $\omega \in B$ . Now consider setting

$$\phi(\omega) = \frac{1}{n} \sum_i f_\omega(\mathbf{v}_i).$$

The only thing that stops us from applying Lemma 16.3.2 at this point is that  $\phi$  is not defined on  $S$ , because  $f_\omega$  was not defined for  $\omega \in S$ . To fix this, we define for  $\alpha \in S$

$$f_\alpha(\mathbf{z}) = \begin{cases} \alpha & \text{if } \mathbf{z} \neq -\alpha \\ -\alpha & \text{otherwise.} \end{cases}$$

We then encounter the problem that  $f_\alpha(\mathbf{z})$  is not a continuous function of  $\alpha$  because it is discontinuous at  $\alpha = -\mathbf{v}_i$ . But, this shouldn't be a problem because the point  $\omega$  at which  $\phi(\omega) = 0$  won't be on or near the boundary. The following argument makes this intuition formal.

We set

$$h_\omega(\mathbf{z}) = \begin{cases} 1 & \text{if } \text{dist}(\omega, \mathbf{z}) < 2 - \epsilon, \text{ and} \\ (2 - \text{dist}(\omega, \mathbf{z}))/\epsilon & \text{otherwise.} \end{cases}$$

Now, the function  $f_\alpha(\mathbf{z})h_\omega(\mathbf{z})$  is continuous on all of  $B$ . So, we may set

$$\phi(\omega) \stackrel{\text{def}}{=} \frac{\sum_i f_\omega(\mathbf{v}_i)h_\omega(\mathbf{v}_i)}{\sum_i h_\omega(\mathbf{v}_i)},$$

which is now continuous and is the identity map on  $S$ .

So, for any  $\epsilon > 0$ , we may now apply Lemma 16.3.2 to find an  $\omega$  for which

$$\phi(\omega) = \mathbf{0}.$$

To finish the proof, we need to get rid of this  $\epsilon$ . That is, we wish to show that  $\omega$  is bounded away from  $S$ , say by  $\mu$ , for all sufficiently small  $\epsilon$ . If that is the case, then we will have  $\text{dist}(\omega, \mathbf{v}_i) \geq \mu > 0$  for all sufficiently small  $\epsilon$ . So, for  $\epsilon < \mu$  and sufficiently small,  $h_\omega(\mathbf{v}_i) = 1$  for all  $i$ , and we recover the  $\epsilon = 0$  case.

One can verify that this holds provided that the points  $\mathbf{v}_i$  are distinct and there are at least 3 of them.

Finally, recall that this is not exactly the theorem we wanted to prove: this theorem deals with  $\mathbf{v}_i$ , and not the centers of caps. The difficulty with centers of caps is that they move as the caps move. However, this can be overcome by observing that the centers remain inside the caps, and move continuously with  $\omega$ . For a complete proof, see [ST07, Theorem 4.2]

## 16.4 Further progress

This result has been improved in many ways. Jonathan Kelner [Kel06] generalized this result to graphs of bounded genus. Kelner, Lee, Price and Teng [KLPT09] obtained analogous bounds for  $\lambda_k$  for  $k \geq 2$ . Biswal, Lee and Rao [BLR10] developed an entirely new set of techniques to prove these results. Their techniques improve these bounds, and extend them to graphs that do not have  $K_h$  minors for any constant  $h$ .

## References

- [Bar82] Earl R. Barnes. An algorithm for partitioning the nodes of a graph. *SIAM Journal on Algebraic and Discrete Methods*, 3(4):541–550, 1982.
- [BLR10] P. Biswal, J. Lee, and S. Rao. Eigenvalue bounds, spectral partitioning, and metrical deformations via flows. *Journal of the ACM*, 2010. to appear.
- [DH72] W. E. Donath and A. J. Hoffman. Algorithms for partitioning graphs and computer logic based on eigenvectors of connection matrices. *IBM Technical Disclosure Bulletin*, 15(3):938–944, 1972.
- [DH73] W. E. Donath and A. J. Hoffman. Lower bounds for the partitioning of graphs. *IBM Journal of Research and Development*, 17(5):420–425, September 1973.
- [Kel06] Jonathan A. Kelner. Spectral partitioning, eigenvalue bounds, and circle packings for graphs of bounded genus. *SIAM J. Comput.*, 35(4):882–902, 2006.
- [KLPT09] Jonathan A. Kelner, James Lee, Gregory Price, and Shang-Hua Teng. Higher eigenvalues of graphs. In *Proceedings of the 50th IEEE Symposium on Foundations of Computer Science*, 2009.
- [PSL90] A. Pothen, H. D. Simon, and K.-P. Liou. Partitioning sparse matrices with eigenvectors of graphs. *SIAM J. Matrix Anal. Appl.*, 11:430–452, 1990.
- [Sim91] Horst D. Simon. Partitioning of unstructured problems for parallel processing. *Computing Systems in Engineering*, 2:135–148, 1991.
- [ST07] Daniel A. Spielman and Shang-Hua Teng. Spectral partitioning works: Planar graphs and finite element meshes. *Linear Algebra and its Applications*, 421:284–305, 2007.

## Properties of Expander Graphs

*Daniel A. Spielman*

October 29, 2018

## 17.1 Overview

We say that a  $d$ -regular graph is a good expander if all of its adjacency matrix eigenvalues are small. To quantify this, we set a threshold  $\epsilon > 0$ , and require that each adjacency matrix eigenvalue, other than  $d$ , has absolute value at most  $\epsilon d$ . This is equivalent to requiring all non-zero eigenvalues of the Laplacian to be within  $\epsilon d$  of  $d$ .

In this lecture, we will:

1. Show that this condition is equivalent to approximating the complete graph.
2. Prove that this condition implies that the number of edges between sets of vertices in the graph is approximately the same as in a  $d$ -regular random graph.
3. Prove Tanner's Theorem: that small sets of vertices have many neighbors.
4. Derive the Alon-Boppana bound, which says that  $\epsilon$  cannot be asymptotically smaller than  $2\sqrt{d-1}/d$ . This will tell us that the asymptotically best expanders are the Ramanujan graphs.

Random  $d$ -regular graphs are expander graphs. Explicitly constructed expander graphs have proved useful in a large number of algorithms and theorems. We will see some applications of them next week.

## 17.2 Expanders as Approximations of the Complete Graph

One way of measuring how well two matrices  $\mathbf{A}$  and  $\mathbf{B}$  approximate each other is to measure the operator norm of their difference:  $\|\mathbf{A} - \mathbf{B}\|$ . Since I consider the operator norm by default, I will just refer to it as the norm. Recall that the norm of a matrix  $\mathbf{M}$  is defined to be its largest singular value:

$$\|\mathbf{M}\| = \max_{\mathbf{x}} \frac{\|\mathbf{M}\mathbf{x}\|}{\|\mathbf{x}\|},$$

where the norms in the fraction are the standard Euclidean vector norms. The norm of a symmetric matrix is just the largest absolute value of one of its eigenvalues. It can be very different for a non-symmetric matrix.

For this lecture, we define an  $\epsilon$ -expander to be a  $d$ -regular graph whose adjacency matrix eigenvalues satisfy  $|\mu_i| \leq \epsilon d$  for  $\mu_i \geq 2$ . As the Laplacian matrix eigenvalues are given by  $\lambda_i = d - \mu_i$ , this is equivalent to  $|d - \lambda_i| \leq \epsilon d$  for  $i \geq 2$ . It is also equivalent to

$$\|\mathbf{L}_G - (d/n)\mathbf{L}_{K_n}\| \leq \epsilon d.$$

For this lecture, I define a graph  $G$  to be an  $\epsilon$ -approximation of a graph  $H$  if

$$(1 - \epsilon)H \preccurlyeq G \preccurlyeq (1 + \epsilon)H,$$

where I recall that I say  $H \preccurlyeq G$  if for all  $\mathbf{x}$

$$\mathbf{x}^T \mathbf{L}_H \mathbf{x} \leq \mathbf{x}^T \mathbf{L}_G \mathbf{x}.$$

I warn you that this definition is not symmetric. When I require a symmetric definition, I usually use the condition  $(1 + \epsilon)^{-1}H \preccurlyeq G$  instead of  $(1 - \epsilon)H \preccurlyeq G$ .

If  $G$  is an  $\epsilon$ -expander, then for all  $\mathbf{x} \in \mathbb{R}^V$  that are orthogonal to the constant vectors,

$$(1 - \epsilon)d\mathbf{x}^T \mathbf{x} \leq \mathbf{x}^T \mathbf{L}_G \mathbf{x} \leq (1 + \epsilon)d\mathbf{x}^T \mathbf{x}.$$

On the other hand, for the complete graph  $K_n$ , we know that all  $\mathbf{x}$  orthogonal to the constant vectors satisfy

$$\mathbf{x}^T \mathbf{L}_{K_n} \mathbf{x} = n\mathbf{x}^T \mathbf{x}.$$

Let  $H$  be the graph

$$H = \frac{d}{n}K_n,$$

so

$$\mathbf{x}^T \mathbf{L}_H \mathbf{x} = d\mathbf{x}^T \mathbf{x}.$$

So,  $G$  is an  $\epsilon$ -approximation of  $H$ .

This tells us that  $\mathbf{L}_G - \mathbf{L}_H$  is a matrix of small norm. Observe that

$$(1 - \epsilon)\mathbf{L}_H \preccurlyeq \mathbf{L}_G \preccurlyeq (1 + \epsilon)\mathbf{L}_H \quad \text{implies} \quad -\epsilon\mathbf{L}_H \preccurlyeq \mathbf{L}_G - \mathbf{L}_H \preccurlyeq \epsilon\mathbf{L}_H.$$

As  $\mathbf{L}_G$  and  $\mathbf{L}_H$  are symmetric, and all eigenvalues of  $\mathbf{L}_H$  are 0 or  $d$ , we may infer

$$\|\mathbf{L}_G - \mathbf{L}_H\| \leq \epsilon d. \tag{17.1}$$

### 17.3 Quasi-Random Properties of Expanders

There are many ways in which expander graphs act like random graphs. Conversely, one can prove that a random  $d$ -regular graph is an expander graph with reasonably high probability [Fri08].

We will see that all sets of vertices in an expander graph act like random sets of vertices. To make this precise, imagine creating a random set  $S \subset V$  by including each vertex in  $S$  independently

with probability  $\alpha$ . How many edges do we expect to find between vertices in  $S$ ? Well, for every edge  $(u, v)$ , the probability that  $u \in S$  is  $\alpha$  and the probability that  $v \in S$  is  $\alpha$ , so the probability that both endpoints are in  $S$  is  $\alpha^2$ . So, we expect an  $\alpha^2$  fraction of the edges to go between vertices in  $S$ . We will show that this is true for all sufficiently large sets  $S$  in an expander.

In fact, we will prove a stronger version of this statement for two sets  $S$  and  $T$ . Imagine including each vertex in  $S$  independently with probability  $\alpha$  and each vertex in  $T$  with probability  $\beta$ . We allow vertices to belong to both  $S$  and  $T$ . For how many ordered pairs  $(u, v) \in E$  do we expect to have  $u \in S$  and  $v \in T$ ? Obviously, it should hold for an  $\alpha\beta$  fraction of the pairs.

For a graph  $G = (V, E)$ , define

$$\vec{E}(S, T) = \{(u, v) : u \in S, v \in T, (u, v) \in E\}.$$

We have put the arrow above the  $E$  in the definition, because we are considering ordered pairs of vertices. When  $S$  and  $T$  are disjoint

$$|\vec{E}(S, T)|$$

is precisely the number of edges between  $S$  and  $T$ , while

$$|\vec{E}(S, S)|$$

counts every edge inside  $S$  twice.

The following bound is a slight extension by Beigel, Margulis and Spielman [BMS93] of a bound originally proved by Alon and Chung [AC88].

**Theorem 17.3.1.** *Let  $G = (V, E)$  be a  $d$ -regular graph that  $\epsilon$ -approximates  $\frac{d}{n}K_n$ . Then, for every  $S \subseteq V$  and  $T \subseteq V$ ,*

$$|\vec{E}(S, T)| - \alpha\beta dn \leq \epsilon dn \sqrt{(\alpha - \alpha^2)(\beta - \beta^2)},$$

where  $|S| = \alpha n$  and  $|T| = \beta n$ .

Observe that when  $\alpha$  and  $\beta$  are greater than  $\epsilon$ , the term on the right is less than  $\alpha\beta dn$ .

In class, we will just prove this in the case that  $S$  and  $T$  are disjoint.

*Proof.* The first step towards the proof is to observe

$$\chi_S^T \mathbf{L}_G \chi_T = d |S \cap T| - |\vec{E}(S, T)|.$$

Let  $H = \frac{d}{n}K_n$ . As  $G$  is a good approximation of  $H$ , let's compute

$$\chi_S^T \mathbf{L}_H \chi_T = \chi_S^T \left( dI - \frac{d}{n}J \right) \chi_T = d |S \cap T| - \frac{d}{n} |S| |T| = d |S \cap T| - \alpha\beta dn.$$

So,

$$|\vec{E}(S, T)| - \alpha\beta dn = |\chi_S^T \mathbf{L}_G \chi_T - \chi_S^T \mathbf{L}_H \chi_T|.$$

As

$$\|\mathbf{L}_G - \mathbf{L}_H\| \leq \epsilon d,$$

$$\begin{aligned} \chi_S^T \mathbf{L}_H \chi_T - \chi_S^T \mathbf{L}_G \chi_T &= \chi_S^T (\mathbf{L}_H - \mathbf{L}_G) \chi_T \\ &\leq \|\chi_S\| \|(\mathbf{L}_H - \mathbf{L}_G) \chi_T\| \\ &\leq \|\chi_S\| \|\mathbf{L}_H - \mathbf{L}_G\| \|\chi_T\| \\ &\leq \epsilon d \|\chi_S\| \|\chi_T\| \\ &= \epsilon d n \sqrt{\alpha \beta}. \end{aligned}$$

This is almost as good as the bound we are trying to prove. To prove the claimed bound, recall that  $\mathbf{L}_H \mathbf{x} = \mathbf{L}_H(\mathbf{x} + c\mathbf{1})$  for all  $c$ . So, let  $\mathbf{x}_S$  and  $\mathbf{x}_T$  be the result of orthogonalizing  $\chi_S$  and  $\chi_T$  with respect to the constant vectors. By Claim 2.4.2 (from Lecture 2),  $\|\mathbf{x}_S\| = n(\alpha - \alpha^2)$ . So, we obtain the improved bound

$$\mathbf{x}_S^T (\mathbf{L}_H - \mathbf{L}_G) \mathbf{x}_T = \chi_S^T (\mathbf{L}_H - \mathbf{L}_G) \chi_T,$$

while

$$\|\mathbf{x}_S\| \|\mathbf{x}_T\| = n \sqrt{(\alpha - \alpha^2)(\beta - \beta^2)}.$$

So, we may conclude

$$\left| \vec{E}(S, T) \right| - \alpha \beta d n \leq \epsilon d n \sqrt{(\alpha - \alpha^2)(\beta - \beta^2)}.$$

□

We remark that when  $S$  and  $T$  are disjoint, the same proof goes through even if  $G$  is irregular and weighted if we replace  $\vec{E}(S, T)$  with

$$w(S, T) = \sum_{(u,v) \in E, u \in S, v \in T} w(u, v).$$

We only need the fact that  $G$   $\epsilon$ -approximates  $\frac{d}{n} K_n$ . See [BSS12] for details.

## 17.4 Vertex Expansion

The reason for the name *expander graph* is that small sets of vertices in expander graphs have unusually large numbers of neighbors. For  $S \subset V$ , let  $N(S)$  denote the set of vertices that are neighbors of vertices in  $S$ . The following theorem, called Tanner's Theorem, provides a lower bound on the size of  $N(S)$ .

**Theorem 17.4.1** ([Tan84]). *Let  $G = (V, E)$  be a  $d$ -regular graph on  $n$  vertices that  $\epsilon$ -approximates  $\frac{d}{n} K_n$ . Then, for all  $S \subseteq V$ ,*

$$|N(S)| \geq \frac{|S|}{\epsilon^2(1 - \alpha) + \alpha},$$

where  $|S| = \alpha n$ .

Note that when  $\alpha$  is much less than  $\epsilon^2$ , the term on the right is approximately  $|S|/\epsilon^2$ , which can be much larger than  $|S|$ . We will derive Tanner's theorem from Theorem 17.3.1.

*Proof.* Let  $R = N(S)$  and let  $T = V - R$ . Then, there are no edges between  $S$  and  $T$ . Let  $|T| = \beta n$  and  $|R| = \gamma n$ , so  $\gamma = 1 - \beta$ . By Theorem 17.3.1, it must be the case that

$$\alpha\beta dn \leq \epsilon dn \sqrt{(\alpha - \alpha^2)(\beta - \beta^2)}.$$

The lower bound on  $\gamma$  now follows by re-arranging terms. Dividing through by  $dn$  and squaring both sides gives

$$\begin{aligned} \alpha^2\beta^2 &\leq \epsilon^2(\alpha - \alpha^2)(\beta - \beta^2) && \iff \\ \alpha\beta &\leq \epsilon^2(1 - \alpha)(1 - \beta) && \iff \\ \frac{\beta}{1 - \beta} &\leq \frac{\epsilon^2(1 - \alpha)}{\alpha} && \iff \\ \frac{1 - \gamma}{\gamma} &\leq \frac{\epsilon^2(1 - \alpha)}{\alpha} && \iff \\ \frac{1}{\gamma} &\leq \frac{\epsilon^2(1 - \alpha) + \alpha}{\alpha} && \iff \\ \gamma &\geq \frac{\alpha}{\epsilon^2(1 - \alpha) + \alpha}. \end{aligned}$$

□

If instead of  $N(S)$  we consider  $N(S) - S$ , then  $T$  and  $S$  are disjoint, so the same proof goes through for weighted, irregular graphs that  $\epsilon$ -approximate  $\frac{d}{n}K_n$ .

## 17.5 How well can a graph approximate the complete graph?

Consider applying Tanner's Theorem with  $S = \{v\}$  for some vertex  $v$ . As  $v$  has exactly  $d$  neighbors, we find

$$\epsilon^2(1 - 1/n) + 1/n \geq 1/d,$$

from which we see that  $\epsilon$  must be at least  $1/\sqrt{d + d^2/n}$ , which is essentially  $1/\sqrt{d}$ . But, how small can it be?

The Ramanujan graphs, constructed by Margulis [Mar88] and Lubotzky, Phillips and Sarnak [LPS88] achieve

$$\epsilon \leq \frac{2\sqrt{d-1}}{d}.$$

We will see that if we keep  $d$  fixed while we let  $n$  grow,  $\epsilon$  cannot exceed this bound in the limit. We will prove an upper bound on  $\epsilon$  by constructing a suitable test function.

As a first step, choose two vertices  $v$  and  $u$  in  $V$  whose neighborhoods do not overlap. Consider the vector  $\mathbf{x}$  defined by

$$\mathbf{x}(i) = \begin{cases} 1 & \text{if } i = u, \\ 1/\sqrt{d} & \text{if } i \in N(u), \\ -1 & \text{if } i = v, \\ -1/\sqrt{d} & \text{if } i \in N(v), \\ 0 & \text{otherwise.} \end{cases}$$

Now, compute the Rayleigh quotient with respect to  $\mathbf{x}$ . The numerator is the sum over all edges of the squares of differences across the edges. This gives  $(1 - 1/\sqrt{d})^2$  for the edges attached to  $u$  and  $v$ , and  $1/d$  for the edges attached to  $N(u)$  and  $N(v)$  but not to  $u$  or  $v$ , for a total of

$$2d(1 - 1/\sqrt{d})^2 + 2d(d-1)/d = 2(d - 2\sqrt{d} + 1 + (d-1)) = 2(2d - 2\sqrt{d}).$$

On the other hand, the denominator is 4, so we find

$$\frac{\mathbf{x}^T L \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = d - \sqrt{d}.$$

If we use instead the vector

$$\mathbf{y}(i) = \begin{cases} 1 & \text{if } i = u, \\ -1/\sqrt{d} & \text{if } i \in N(u), \\ -1 & \text{if } i = v, \\ 1/\sqrt{d} & \text{if } i \in N(v), \\ 0 & \text{otherwise,} \end{cases}$$

we find

$$\frac{\mathbf{y}^T L \mathbf{y}}{\mathbf{y}^T \mathbf{y}} = d + \sqrt{d}.$$

This is not so impressive, as it merely tells us that  $\epsilon \geq 1/\sqrt{d}$ , which we already knew. But, we can improve this argument by pushing it further. We do this by modifying it in two ways. First, we extend  $\mathbf{x}$  to neighborhoods of neighborhoods of  $u$  and  $v$ . Second, instead of basing the construction at vertices  $u$  and  $v$ , we base it at two edges. This way, each vertex has  $d-1$  edges to those that are farther away from the centers of the construction.

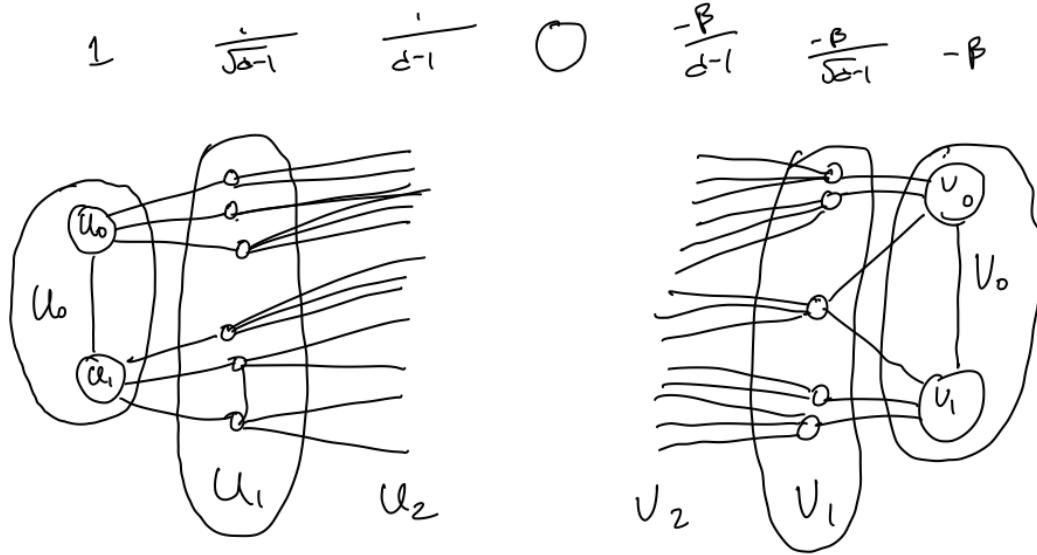
The following theorem is attributed to A. Nilli [Nil91], but we suspect it was written by N. Alon.

**Theorem 17.5.1.** *Let  $G$  be a  $d$ -regular graph containing two edges  $(u_0, u_1)$  and  $(v_0, v_1)$  that are at distance at least  $2k+2$ . Then*

$$\lambda_2 \leq d - 2\sqrt{d-1} + \frac{2\sqrt{d-1}-1}{k+1}.$$

*Proof.* Define the following neighborhoods.

$$\begin{aligned} U_0 &= \{u_0, u_1\} \\ U_i &= N(U_{i-1}) - \cup_{j < i} U_j, \quad \text{for } 0 < i \leq k, \\ V_0 &= \{v_0, v_1\} \\ V_i &= N(V_{i-1}) - \cup_{j < i} V_j, \quad \text{for } 0 < i \leq k. \end{aligned}$$

Figure 17.1: The construction of  $\mathbf{x}$ .

That is,  $U_i$  consists of exactly those vertices at distance  $i$  from  $U_0$ . Note that there are no edges between any vertices in any  $U_i$  and any  $V_j$ .

Our test vector for  $\lambda_2$  will be given by

$$\mathbf{x}(a) = \begin{cases} \frac{1}{(d-1)^{-i/2}} & \text{for } a \in U_i \\ -\frac{\beta}{(d-1)^{-i/2}} & \text{for } a \in V_i \\ 0 & \text{otherwise.} \end{cases}$$

We choose  $\beta$  so that  $\mathbf{x}$  is orthogonal to  $\mathbf{1}$ .

We now find that the Rayleigh quotient of  $\mathbf{x}$  with respect to  $\mathbf{L}$  is at most

$$\frac{X_0 + \beta^2 Y_0}{X_1 + \beta^2 Y_1},$$

where

$$X_0 = \sum_{i=0}^{k-1} |U_i| (d-1) \left( \frac{1 - 1/\sqrt{d-1}}{(d-1)^{-i/2}} \right)^2 + |U_k| (d-1)^{-k+1}, \text{ and } X_1 = \sum_{i=0}^k |U_i| (d-1)^{-i}$$

and

$$Y_0 = \sum_{i=0}^{k-1} |V_i| (d-1) \left( \frac{1 - 1/\sqrt{d-1}}{(d-1)^{-i/2}} \right)^2 + |V_k| (d-1)^{-k+1}, \text{ and } Y_1 = \sum_{i=0}^k |V_i| (d-1)^{-i}.$$

By my favorite inequality, it suffices to prove upper bounds on  $X_0/X_1$  and  $Y_0/Y_1$ . So, consider

$$\frac{\sum_{i=0}^{k-1} |U_i| (d-1) \left( \frac{1-1/\sqrt{d-1}}{(d-1)^{-i/2}} \right)^2 + |U_k| (d-1)^{-k+1}}{\sum_{i=0}^k |U_i| (d-1)^{-i}}.$$

For now, let's focus on the numerator,

$$\begin{aligned} & \sum_{i=0}^{k-1} |U_i| (d-1) \left( \frac{1-1/\sqrt{d-1}}{(d-1)^{-i/2}} \right)^2 + |U_k| (d-1)(d-1)^{-k} \\ &= \sum_{i=0}^{k-1} \frac{|U_i|}{(d-1)^i} (d - 2\sqrt{d-1}) + \frac{|U_k|}{(d-1)^k} (d-1) \\ &= \sum_{i=0}^{k-1} \frac{|U_i|}{(d-1)^i} (d - 2\sqrt{d-1}) + \frac{|U_k|}{(d-1)^k} (d - 2\sqrt{d-1}) + \frac{|U_k|}{(d-1)^k} (2\sqrt{d-1} - 1) \\ &= \sum_{i=0}^k \frac{|U_i|}{(d-1)^i} (d - 2\sqrt{d-1}) + \frac{|U_k|}{(d-1)^k} (2\sqrt{d-1} - 1). \end{aligned}$$

To upper bound the Rayleigh quotient, we observe that the left-most of these terms contributes

$$\frac{\sum_{i=0}^k \frac{|U_i|}{(d-1)^i} (d - 2\sqrt{d-1})}{\sum_{i=0}^k |U_i| (d-1)^{-i}} = d - 2\sqrt{d-1}.$$

To bound the impact of the remaining term,

$$\frac{|U_k|}{(d-1)^k} (2\sqrt{d-1} - 1),$$

note that

$$|U_k| \leq (d-1)^{k-i} |U_i|.$$

So, we have

$$\frac{|U_k|}{(d-1)^k} \leq \frac{1}{k+1} \sum_{i=0}^k \frac{|U_i|}{(d-1)^i}.$$

Thus, the last term contributes at most

$$\frac{2\sqrt{d-1}}{k+1}$$

to the Rayleigh quotient.

□

## 17.6 Open Problems

What can we say about  $\lambda_n$ ? In a previous iteration of this course, I falsely asserted that the same proof tells us that

$$\lambda_n \geq d + 2\sqrt{d-1} - \frac{2\sqrt{d-1} - 1}{k+1}.$$

But, the proof did not work.

Another question is how well a graph of average degree  $d$  can approximate the complete graph. That is, let  $G$  be a graph with  $dn/2$  edges, but let  $G$  be irregular. While I doubt that irregularity helps one approximate the complete graph, I do not know how to prove it.

We can generalize this question further. Let  $G = (V, E, w)$  be a weighted graph with  $dn/2$  edges. Can we prove that  $G$  cannot approximate a complete graph any better than the Ramanujan graphs do? I conjecture that for every  $d$  and every  $\beta > 0$  there is an  $n_0$  so that for every graph of average degree  $d$  on  $n \geq n_0$  vertices,

$$\frac{\lambda_2}{\lambda_n} \leq \frac{d - 2\sqrt{d-1}}{d + 2\sqrt{d-1}} + \beta.$$

## References

- [AC88] Noga Alon and Fan Chung. Explicit construction of linear sized tolerant networks. *Discrete Mathematics*, 72:15–19, 1988.
- [BMS93] Richard Beigel, Grigorii Margulis, and Daniel A. Spielman. Fault diagnosis in a small constant number of parallel testing rounds. In *SPAA '93: Proceedings of the fifth annual ACM symposium on Parallel algorithms and architectures*, pages 21–29, New York, NY, USA, 1993. ACM.
- [BSS12] Joshua Batson, Daniel A Spielman, and Nikhil Srivastava. Twice-Ramanujan sparsifiers. *SIAM Journal on Computing*, 41(6):1704–1721, 2012.
- [Fri08] Joel Friedman. *A Proof of Alon’s Second Eigenvalue Conjecture and Related Problems*. Number 910 in Memoirs of the American Mathematical Society. American Mathematical Society, 2008.
- [LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.
- [Mar88] G. A. Margulis. Explicit group theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators. *Problems of Information Transmission*, 24(1):39–46, July 1988.
- [Nil91] A. Nilli. On the second eigenvalue of a graph. *Discrete Math*, 91:207–210, 1991.
- [Tan84] R. Michael Tanner. Explicit concentrators from generalized  $n$ -gons. *SIAM Journal Alg. Disc. Meth.*, 5(3):287–293, September 1984.

## A simple construction of expander graphs

*Daniel A. Spielman*

October 31, 2018

## 18.1 Overview

Our goal is to prove that for every  $\epsilon > 0$  there is a  $d$  for which we can efficiently construct an infinite family of  $d$ -regular  $\epsilon$ -expanders. I recall that these are graphs whose adjacency matrix eigenvalues satisfy  $|\mu_i| \leq \epsilon d$  and whose Laplacian matrix eigenvalues satisfy  $|d - \lambda_i| \leq \epsilon d$ . Viewed as a function of  $\epsilon$ , the  $d$  that we obtain in this construction is rather large. But, it is a constant. The challenge here is to construct infinite families with fixed  $d$  and  $\epsilon$ .

Before we begin, I remind you that in Lecture 5 we showed that random generalized hypercubes were  $\epsilon$  expanders of degree  $f(\epsilon) \log n$ , for some function  $f$ . The reason they do not solve today's problem is that their degrees depend on the number of vertices. However, today's construction will require some small expander graph, and these graphs or graphs like them can serve in that role. So that we can obtain a construction for every number of vertices  $n$ , we will exploit random generalized ring graphs. Their analysis is similar to that of random generalized hypercubes.

**Claim 18.1.1.** *There exists a function  $f(\epsilon)$  so that for every  $\epsilon > 0$  and every sufficiently large  $n$  the Cayley graph with group  $\mathbb{Z}/n$  and a random set of at least  $f(\epsilon) \log n$  generators is an  $\epsilon$ -expander with high probability.*

I am going to present the simplest construction of expanders that I have been able to find. By “simplest”, I mean optimizing the tradeoff of simplicity of construction with simplicity of analysis. It is inspired by the Zig-Zag product and replacement product constructions presented by Reingold, Vadhan and Wigderson [RVW02].

For those who want the quick description, here it is. Begin with an expander. Take its line graph. Observe that the line graph is a union of cliques. So, replace each clique by a small expander. We need to improve the expansion slightly, so square the graph. Square one more time. Repeat.

The analysis will be simple because all of the important parts are equalities, which I find easier to understand than inequalities.

While this construction requires the choice of two expanders of constant size, it is explicit in the sense that we can obtain a simple *implicit* representation of the graph: if the name of a vertex in the graph is written using  $b$  bits, then we can compute its neighbors in time polynomial in  $b$ .

## 18.2 Squaring Graphs

We will first show that we can obtain a family of  $\epsilon$  expanders from a family of  $\beta$ -expanders for any  $\beta < 1$ . The reason is that squaring a graph makes it a better expander, although at the cost of increasing its degree.

Given a graph  $G$ , we define the graph  $G^2$  to be the graph in which vertices  $u$  and  $v$  are connected if they are at distance 2 in  $G$ . Formally,  $G^2$  should be a weighted graph in which the weight of an edge is the number of such paths. When first thinking about this, I suggest that you ignore the issue. When you want to think about it, I suggest treating such weighted edges as multiedges.

We may form the adjacency matrix of  $G^2$  from the adjacency matrix of  $G$ . Let  $\mathbf{M}$  be the adjacency matrix of  $G$ . Then  $\mathbf{M}^2(u, v)$  is the number of paths of length 2 between  $u$  and  $v$  in  $G$ , and  $\mathbf{M}^2(v, v)$  is always  $d$ . We will eliminate those self-loops. So,

$$\mathbf{M}_{G^2} = \mathbf{M}_G^2 - dI_n.$$

If  $G$  has no cycles of length up to 4, then all of the edges in its square will have weight 1. The following claim is immediate from this definition.

**Claim 18.2.1.** *The adjacency matrix eigenvalues of  $G^2$  are precisely*

$$\mu_i^2 - d,$$

where  $\mu_1, \dots, \mu_n$  are the adjacency matrix eigenvalues of  $G$ .

**Lemma 18.2.2.** *If  $\{G_i\}_i$  is an infinite family of  $d$ -regular  $\beta$ -expanders for  $\beta \geq 1/\sqrt{d-1}$ , then  $\{G_i^2\}_i$  is an infinite family of  $d(d-1)$ -regular  $\beta^2$  expanders.*

We remark that the case of  $\beta > 1/\sqrt{d-1}$ , or even larger, is the case of interest. We are not expecting to work with graphs that beat the Ramanujan bound,  $2\sqrt{d-1}/d$ .

*Proof.* For  $\mu$  an adjacency matrix eigenvalue of  $G_i$  other than  $d$ , we have

$$\frac{\mu^2 - d}{d(d-1)} = \frac{\mu^2 - d}{d^2 - d} \leq \frac{\mu^2}{d^2} \leq \beta^2.$$

On the other hand, every adjacency eigenvalue of  $G_i^2$  is at least  $-d$ , which is at least  $-\beta^2 d(d-1)$ .  $\square$

So, by squaring enough times, we can convert a family of  $\beta$  expanders for any  $\beta < 1$  into a family of  $\epsilon$  expanders.

## 18.3 The Relative Spectral Gap

To measure the qualities of the graphs that appear in our construction, we define a quantity that we will call the *relative spectral gap* of a  $d$ -regular graph:

$$r(G) \stackrel{\text{def}}{=} \min \left( \frac{\lambda_2(G)}{d}, \frac{2d - \lambda_n}{d} \right).$$

The graphs with larger relative spectral gaps are better expanders. An  $\epsilon$ -expander has relative spectral gap at least  $1 - \epsilon$ , and vice versa. Because we can square graphs, we know that it suffices to find an infinite family of graphs with relative spectral gap strictly greater than 0.

We now state exactly how squaring impacts the relative spectral gap of a graph.

**Corollary 18.3.1.** *If  $G$  has relative spectral gap  $\beta$ , then  $G^2$  has relative spectral gap at least*

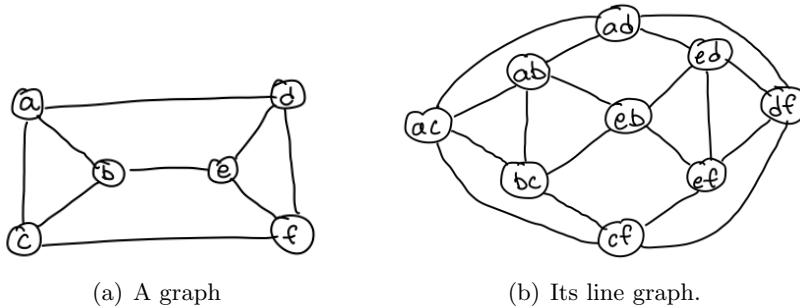
$$2\beta - \beta^2.$$

Note that when  $\beta$  is small, this gap is approximately  $2\beta$ .

## 18.4 Line Graphs

Our construction will leverage small expanders to make bigger expanders. To begin, we need a way to make a graph bigger and still say something about its spectrum.

We use the *line graph* of a graph. Let  $G = (V, E)$  be a graph. The line graph of  $G$  is the graph whose vertices are the edges of  $G$  in which two are connected if they share an endpoint in  $G$ . That is,  $((u, v), (w, z))$  is an edge of the line graph if one of  $\{u, v\}$  is the same as one of  $\{w, z\}$ . The line graph is often written  $L(G)$ , but we won't do that in this class so that we can avoid confusion with the Laplacian.



Let  $G$  be a  $d$ -regular graph with  $n$  vertices, and let  $H$  be its line graph<sup>1</sup>. As  $G$  has  $dn/2$  edges,  $H$  has  $dn/2$  vertices. Each vertex of  $H$ , say  $(u, v)$ , has degree  $2(d-1)$ :  $d-1$  neighbors for the other edges attached to  $u$  and  $d-1$  for  $v$ . In fact, if we just consider one vertex  $u$  in  $V$ , then all vertices in  $H$  of form  $(u, v)$  of  $G$  will be connected. That is,  $H$  contains a  $d$ -clique for every vertex in  $V$ . We see that each vertex of  $H$  is contained in exactly two of these cliques.

Here is the great fact about the spectrum of the line graph.

**Lemma 18.4.1.** *Let  $G$  be a  $d$ -regular graph with  $n$  vertices, and let  $H$  be its line graph. Then the spectrum of the Laplacian of  $H$  is the same as the spectrum of the Laplacian of  $G$ , except that it has  $dn/2 - n$  extra eigenvalues of  $2d$ .*

---

<sup>1</sup>If  $G$  has multiedges, which is how we interpret integer weights, then we include a vertex in the line graph for each of those multiedges. These will be connected to each other by edges of weight two—one for each vertex that they share. All of the following statements then work out.

Before we prove this lemma, we need to recall the factorization of a Laplacian as the product of the signed edge-vertex adjacency matrix times its transpose. We reserved the letter  $\mathbf{U}$  for this matrix, and defined it by

$$\mathbf{U}((a, b), c) = \begin{cases} 1 & \text{if } a = c \\ -1 & \text{if } b = c \\ 0 & \text{otherwise.} \end{cases}$$

For an unweighted graph, we have

$$\mathbf{L}_G = \mathbf{U}^T \mathbf{U}.$$

Recall that each edge indexes one column, and that we made an arbitrary choice when we ordered the edge  $(a, b)$  rather than  $(b, a)$ . But, this arbitrary choice factors out when we multiply by  $\mathbf{U}^T$ .

## 18.5 The Spectrum of the Line Graph

Define the matrix  $|\mathbf{U}|$  to be the matrix obtained by replacing every entry of  $\mathbf{U}$  by its absolute value. Now, consider  $|\mathbf{U}|^T |\mathbf{U}|$ . It looks just like the Laplacian, except that all of its off-diagonal entries are 1 instead of  $-1$ . So,

$$|\mathbf{U}|^T |\mathbf{U}| = \mathbf{D}_G + \mathbf{M}_G = d\mathbf{I} + \mathbf{M}_G,$$

as  $G$  is  $d$ -regular. We will also consider the matrix  $|\mathbf{U}| |\mathbf{U}|^T$ . This is a matrix with  $nd/2$  rows and  $nd/2$  columns, indexed by edges of  $G$ . The entry at the intersection of row  $(u, v)$  and column  $(w, z)$  is

$$(\delta_u + \delta_v)^T (\delta_w + \delta_z).$$

So, it is 2 if these are the same edge, 1 if they share a vertex, and 0 otherwise. That is

$$|\mathbf{U}| |\mathbf{U}|^T = 2I_{nd/2} + \mathbf{M}_H.$$

Moreover,  $|\mathbf{U}| |\mathbf{U}|^T$  and  $|\mathbf{U}|^T |\mathbf{U}|$  have the same eigenvalues, except that the later matrix has  $nd/2 - n$  extra eigenvalues of 0.

*Proof of Lemma 18.4.1.* First, let  $\lambda_i$  be an eigenvalue of  $\mathbf{L}_G$ . We see that

$$\begin{aligned} \lambda_i \text{ is an eigenvalue of } \mathbf{D}_G - \mathbf{M}_G &\implies \\ d - \lambda_i \text{ is an eigenvalue of } \mathbf{M}_G &\implies \\ 2d - \lambda_i \text{ is an eigenvalue of } \mathbf{D}_G + \mathbf{M}_G &\implies \\ 2d - \lambda_i \text{ is an eigenvalue of } 2I_{nd/2} + \mathbf{M}_H &\implies \\ 2(d-1) - \lambda_i \text{ is an eigenvalue of } \mathbf{M}_H &\implies \\ \lambda_i \text{ is an eigenvalue of } \mathbf{D}_H - \mathbf{M}_H. \end{aligned}$$

Of course, this last matrix is the Laplacian matrix of  $H$ . We can similarly show that the extra  $dn/2 - n$  zero eigenvalues of  $2I_{nd/2} + \mathbf{M}_H$  become  $2d$  in  $\mathbf{L}_H$ .  $\square$

While the line graph operation preserves  $\lambda_2$ , it causes the degree of the graph to grow. So, we are going to need to do more than just take line graphs to construct expanders.

**Proposition 18.5.1.** *Let  $G$  be a  $d$ -regular graph with  $d \geq 7$  and let  $H$  be its line graph. Then,*

$$r(H) = \frac{\lambda_2(G)}{2(d-1)} \geq r(G)/2.$$

*Proof.* For  $G$  a  $d$ -regular graph other than  $K_{d+1}$ ,  $\lambda_2(G) \leq d+1$ . By the Perron-Frobenius theorem (Lemma 6.A.1)  $\lambda_{\max}(G) \leq 2d$  (with equality if and only  $G$  is bipartite). So,  $\lambda_{\max}(H) = 2d$  and  $\lambda_2(H) = \lambda_2(G) \leq d$ . So, the term in the definition of the relative spectral gap corresponding to the largest eigenvalue of  $H$  satisfies

$$\frac{2(2d-2) - \lambda_{\max}(H)}{2d-2} = \frac{2(2d-2) - 2d}{2d-2} = 1 - \frac{2}{d} \geq 5/7,$$

as  $d \geq 7$ . On the other hand,

$$\frac{\lambda_2(H)}{2d-2} \leq \frac{d}{2d-2} \leq 2/3.$$

As  $2/3 < 5/7$ ,

$$\min\left(\frac{\lambda_2(H)}{2d-2}, \frac{2(2d-2) - \lambda_{\max}(H)}{2d-2}\right) = \frac{\lambda_2(H)}{2d-2} = \frac{\lambda_2(G)}{2d-2} \geq r(G/2).$$

□

While the line graph of  $G$  has more vertices, its degree is higher and its relative spectral gap is approximately half that of  $G$ . We can improve the relative spectral gap by squaring. In the next section, we show how to lower the degree.

## 18.6 Approximations of Line Graphs

Our next step will be to construct approximations of line graphs. We already know how to approximate complete graphs: we use expanders. As line graphs are sums of complete graphs, we will approximate them by sums of expanders. That is, we replace each clique in the line graph by an expander on  $d$  vertices. Since  $d$  will be a constant in our construction, we will be able to get these small expanders from known constructions, like the random generalized ring graphs.

Let  $G$  be a  $d$ -regular graph and let  $Z$  be a graph on  $d$  vertices of degree  $k$  (we will use a low-degree expander). We define the graph

$$G \circledR Z$$

to be the graph obtained by forming the edge graph of  $G$ ,  $H$ , and then replacing every  $d$ -clique in  $H$  by a copy of  $Z$ . Actually, this does not uniquely define  $G \circledR Z$ , as there are many ways to replace a  $d$ -clique by a copy of  $Z$ . But, any choice will work. Note that every vertex of  $G \circledR Z$  has degree  $2k$ .

**Lemma 18.6.1.** *Let  $G$  be a  $d$ -regular graph, let  $H$  be the line graph of  $G$ , and let  $Z$  be a  $k$ -regular  $\alpha$ -expander. Then,*

$$(1 - \alpha) \frac{k}{d} H \preceq G \circledR Z \preceq (1 + \alpha) \frac{k}{d} H$$

*Proof.* As  $H$  is a sum of  $d$ -cliques, let  $H_1, \dots, H_n$  be those  $d$ -cliques. So,

$$\mathbf{L}_H = \sum_{i=1}^n \mathbf{L}_{H_i}.$$

Let  $Z_i$  be the graph obtained by replacing  $H_i$  with a copy of  $Z$ , on the same set of vertices. To prove the lower bound, we compute

$$\mathbf{L}_{G \circledR Z} = \sum_{i=1}^n \mathbf{L}_{Z_i} \succcurlyeq (1 - \alpha) \frac{k}{d} \sum_{i=1}^n \mathbf{L}_{H_i} = (1 - \alpha) \frac{k}{d} \mathbf{L}_H.$$

The upper bound is proved similarly.  $\square$

**Corollary 18.6.2.** *Under the conditions of Lemma 18.6.1,*

$$r(G \circledR Z) \geq \frac{1 - \alpha}{2} r(G).$$

*Proof.* The proof is similar to the proof of Proposition 18.5.1. We have

$$\lambda_2(G \circledR Z) \geq (1 - \alpha) \frac{k \lambda_2(G)}{d},$$

and

$$\lambda_{\max}(G \circledR Z) \leq (1 + \alpha) 2k.$$

So,

$$\min(\lambda_2(G \circledR Z), 2(2k) - \lambda_{\max}(G \circledR Z)) \geq \min\left((1 - \alpha) \frac{k \lambda_2(G)}{d}, (1 - \alpha) 2k\right) = (1 - \alpha) \frac{k \lambda_2(G)}{d},$$

as  $\lambda_2(G) \leq d$ . So,

$$r(G \circledR Z) \geq \frac{1}{2k} (1 - \alpha) k r(G) = \frac{1 - \alpha}{2} r(G).$$

$\square$

So, the relative spectral gap of  $G \circledR Z$  is a little less than half that of  $G$ . But, the degree of  $G \circledR Z$  is  $2k$ , which we will arrange to be much less than the degree of  $G$ ,  $d$ .

We will choose  $k$  and  $d$  so that squaring this graph improves its relative spectral gap, but still leaves its degree less than  $d$ . If  $G$  has relative spectral gap  $\beta$ , then  $G^2$  has relative spectral gap at least

$$2\beta - \beta^2.$$

It is easy to see that when  $\beta$  is small, this gap is approximately  $2\beta$ . This is not quite enough to compensate for the loss of  $(1 - \epsilon)/2$  in the corollary above, so we will have to square the graph once more.

## 18.7 The whole construction

To begin, we need a “small”  $k$ -regular expander graph  $Z$  on

$$d \stackrel{\text{def}}{=} (2k(2k-1))^2 - 2k(2k-1)$$

vertices. It should be an  $\epsilon$ -expander for some small  $\epsilon$ . I believe that  $\epsilon = 1/6$  would suffice. The other graph we will need to begin our construction will be a small  $d$ -regular expander graph  $G_0$ . We use Claim 18.1.1 to establish the existence of both of these. Let  $\beta$  be the relative spectral gap of  $G_0$ . We will assume that  $\beta$  is small, but greater than 0. I believe that  $\beta = 1/5$  will work. Of course, it does not hurt to start with a graph of larger relative spectral gap.

We then construct  $G_0 \circledR Z$ . The degree of this graph is  $2k$ , and its relative spectral gap is a little less than  $\beta/2$ . So, we square the resulting graph, to obtain

$$(G_0 \circledR Z)^2.$$

It has degree approximately  $4k^2$ , and relative spectral gap slightly less than  $\beta$ . But, for induction, we need it to be more than  $\beta$ . So, we square one more time, to get a relative spectral gap a little less than  $2\beta$ . We now set

$$G_1 = \left( (G_0 \circledR Z)^2 \right)^2.$$

The graph  $G_1$  is at least as good an approximation of a complete graph as  $G_0$ , and it has degree approximately  $16k^4$ . In general, we set

$$G_{i+1} = \left( (G_i \circledR Z)^2 \right)^2.$$

To make the inductive construction work, we need for  $Z$  to be a graph of degree  $k$  whose number of vertices equals the degree of  $G$ . This is approximately  $16k^4$ , and is exactly

$$(2k(2k-1))^2 - 2k(2k-1).$$

I'll now carry out the computation of relative spectral gaps with more care. Let's assume that  $G_0$  has a relative spectral gap of  $\beta \geq 4/5$ , and assume, by way of induction, that  $\rho(G_i) \geq 4/5$ . Also assume that  $Z$  is a  $1/6$ -expander. We then find

$$r(G_i \circledR Z) \geq (1 - \epsilon)(4/5)/2 = 1/3.$$

So,  $G_i \circledR Z$  is a  $2/3$ -expander. Our analysis of graph squares then tells us that  $G_{i+1}$  is a  $(2/3)^4$ -expander. So,

$$r(G_{i+1}) \geq 1 - (2/3)^4 = 65/81 > 4/5.$$

By induction, we conclude that every  $G_i$  has relative spectral gap at least  $4/5$ .

To improve their relative spectral gaps of the graphs we produce, we can just square them a few times.

## 18.8 Better Constructions

There is a better construction technique, called the Zig-Zag product [RVW02]. The Zig-Zag construction is a little trickier to understand, but it achieves better expansion. I chose to present the line-graph based construction because its analysis is very closely related to an analysis of the Zig-Zag product.

## References

- [RVW02] Omer Reingold, Salil Vadhan, and Avi Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders. *Annals of Mathematics*, 155(1):157–187, 2002.

## 19.1 Overview

There has been a lot of work on the design of Pseudo-Random Number Generators (PSRGs) that can be proved to work for particular applications. In this class, we will see one of the foundational results in this area, namely Impagliazzo and Zuckerman's [IZ89] use of random walks on expanders to run the same algorithm many times. We are going to perform a very crude analysis that is easy to present in class. Rest assured that much tighter analyses are possible and much better PSRGs have been constructed since.

## 19.2 Why Study PSRGs?

Pseudo-random number generators take a seed which is presumably random (or which has a lot of randomness in it), and then generate a long string of random bits that are supposed to act random. We should first discuss why we would actually want such a thing. I can think of two reasons.

1. Random bits are scarce. This might be surprising. After all, if you look at the last few bits of the time that I last hit a key, it is pretty random. Similarly, the low-order bits of the temperature of the processor in my computer seem pretty random. While these bits are pretty random, there are not too many of them.

Many randomized algorithms need *a lot* of random bits. Sources such as these just do not produce random bits with a frequency sufficient for many applications.

2. If you want to re-run an algorithm, say to de-bug it, it is very convenient to be able to use the same set of random bits by re-running the PSRG with the same seed. If you use truly random bits, you can't do this.

You may also wonder how good the standard pseudo-random number generators are. The first answer is that the default ones, such as `rand` in C, are usually terrible. There are many applications, such as those in my thesis, for which these generators produce behavior that is very different from what one would expect from truly random bits (yes, this is personal). On the other hand, one can use cryptographic functions to create bits that will act random for most purposes, unless one can break the underlying cryptography [HILL99]. But, the resulting generators are usually much slower than the fastest pseudo-random generators. Fundamentally, it comes down to a time-versus-quality tradeoff. The longer you are willing to wait, the better the pseudo-random bits you can get.

### 19.3 Expander Graphs

In today's lecture we will require an infinite family of  $d$ -regular  $1/10$ -expander graphs. We require that  $d$  be a constant, that the graphs have  $2^r$  vertices for all sufficiently large  $r$ , and that we can construct the neighbors of a vertex in time polynomial in  $r$ . That is, we need the graphs to have a simple explicit description. One can construct expanders families of this form using the techniques from last lecture. For today's purposes, the best expanders are the Ramanujan graphs produced by Margulis [Mar88] and Lubotzky, Phillips and Sarnak [LPS88]. Ramanujan graphs of degree  $d = 400$  are  $1/10$ -expanders. See also the work of Alon, Bruck, Naor, Naor and Roth [ABN<sup>+</sup>92] for even more explicit constructions.

While the explicit Ramanujan graphs only exist in certain sizes, none of which do have exactly  $2^r$  vertices, some of them have just a little more than  $2^r$  vertices. It is possible to trim these to make them work, say by ignoring all steps in which the vertex does not correspond to  $r$  bits.

### 19.4 Today's Application : repeating an experiment

Imagine you are given a black box that takes  $r$  bits as input and then outputs either 0 or 1. Moreover, let's assume that the black box is very consistent: we know that it returns the same answer at least 99% of the time. If it almost always returns 0, we will call it a 0-box and if it almost always returns 1, we will call it a 1-box. Our job is to determine whether a given box is a 0 or 1 box. We assume that  $r$  is big, so we don't have time to test the box on all  $2^r$  settings of  $r$  bits. Instead, we could pick  $r$  bits at random, and check what the box returns. If it says "1", then it is probably a 1-box. But, what if we want more than 99% confidence? We could check the box on many choices of  $r$  random bits, and report the majority value returned by the box.<sup>1</sup>. But, this seems to require a new set of random bits for each run. In this lecture, we will prove that 9 new bits per run suffice. Note that the result would be interesting for any constant other than 9.

Since we will not make any assumptions about the black box, we will use truly random bits the first time we test it. But, we will show that we only need 9 new random bits for each successive test. In particular, we will show that if we use our PSRG to generate bits for  $t + 1$  test, then the probability that majority answer is wrong decreases exponentially in  $t$ .

You are probably wondering why we would want to do such a thing. The reason is to increase the accuracy of randomized algorithms. There are many randomized algorithms that provide weak guarantees, such as being correct 99% or 51% of the time. To obtain accurate answers from such algorithms, we run them many times with fresh random bits. You can view such an algorithm has having two inputs: the problem to be solved and its random bits. The black box is the behavior of the algorithm when the problem to be solved is fixed, so it is just working on the random bits.

---

<sup>1</sup>Check for yourself that running it twice doesn't help

## 19.5 The Random Walk Generator

Let  $r$  be the number of bits that our black box takes as input. So, the space of random bits is  $\{0, 1\}^r$ . Let  $X \subset \{0, 1\}^r$  be the settings of the random bits on which the box gives the minority answer, and let  $Y$  be the settings on which it gives the majority answer.

Our pseudo-random generator will use a random walk on a 1/10-expander graph whose vertex set is  $\{0, 1\}^r$ . Recall that we can use  $d = 400$ . For the first input we feed to the black box, we will require  $r$  truly random bits. We treat these bits as a vertex of our graph. For each successive test, we choose a random neighbor of the present vertex, and feed the corresponding bits to the box. That is, we choose a random  $i$  between 1 and 400, and move to the  $i$ th neighbor of the present vertex. Note that we only need  $\log_2 400 \approx 9$  random bits to choose the next vertex. So, we will only need 9 new bits to generate each input we feed to the box after the first.

## 19.6 Formalizing the problem

Assume that we are going to test the box  $t + 1$  times. Our pseudo-random generator will begin at a truly random vertex  $v$ , and then take  $t$  random steps. Recall that we defined  $X$  to be the set of vertices on which the box outputs the minority answer, and we assume that  $|X| \leq 2^r/100$ . If we report the majority of the outcomes of the  $t + 1$  outputs of the box, we will return the correct answer as long as the random walk is inside  $X$  less than half the time. To analyze this, let  $v_0$  be the initial random vertex, and let  $v_1, \dots, v_t$  be the vertices produced by the  $t$  steps of the random walk. Let  $T = \{0, \dots, t\}$  be the time steps, and let  $S = \{i : v_i \in X\}$ . We will prove

$$\Pr [|S| > t/2] \leq \left( \frac{2}{\sqrt{5}} \right)^{t+1}.$$

To begin our analysis, recall that the initial distribution of our random walk is  $\mathbf{p}_0 = \mathbf{1}/n$ . Let  $\chi_X$  and  $\chi_Y$  be the characteristic vectors of  $X$  and  $Y$ , respectively, and let  $\mathbf{D}_X = \text{diag}(\chi_X)$  and  $\mathbf{D}_Y = \text{diag}(\chi_Y)$ . Let

$$\mathbf{W} = \frac{1}{d} \mathbf{M} \tag{19.1}$$

be the transition matrix of the ordinary random walk on  $G$ . We are not using the lazy random walk: it would be silly to use the lazy random walk for this problem, as there is no benefit to re-running the experiment with the same random bits as before. Let  $\omega_1, \dots, \omega_n$  be the eigenvalues of  $\mathbf{W}$ . As the graph is a 1/10-expander,  $|\omega_i| \leq 1/10$  for all  $i \geq 2$ .

Let's see how we can use these matrices to understand the probabilities under consideration. For a probability vector  $\mathbf{p}$  on vertices, the probability that a vertex chosen according to  $\mathbf{p}$  is in  $X$  may be expressed

$$\chi_X^T \mathbf{p} = \mathbf{1}^T \mathbf{D}_X \mathbf{p}.$$

The second form will be more useful, as

$$\mathbf{D}_X \mathbf{p}$$

is the vector obtained by zeroing out the events in which the vertices are not in  $X$ . If we then want to take a step in the graph  $G$ , we multiply by  $\mathbf{W}$ . That is, the probability that the walk starts at vertex in  $X$ , and then goes to a vertex  $i$  is  $\mathbf{q}(i)$  where

$$\mathbf{q} = \mathbf{W} \mathbf{D}_X \mathbf{p}_0.$$

Continuing this way, we see that the probability that the walk is in  $X$  at precisely the times  $i \in R$  is

$$\mathbf{1}^T \mathbf{D}_{Z_t} \mathbf{W} \mathbf{D}_{Z_{t-1}} \mathbf{W} \cdots \mathbf{D}_{Z_1} \mathbf{W} \mathbf{D}_{Z_0} \mathbf{p}_0,$$

where

$$Z_i = \begin{cases} X & \text{if } i \in R \\ Y & \text{otherwise.} \end{cases}$$

We will prove that this probability is at most  $(1/5)^{|R|}$ . It will then follow that

$$\begin{aligned} \Pr [|S| > t/2] &\leq \sum_{|R|>t/2} \Pr [\text{the walk is in } X \text{ at precisely the times in } R] \\ &\leq \sum_{|R|>t/2} \left(\frac{1}{5}\right)^{|R|} \\ &\leq 2^{t+1} \left(\frac{1}{5}\right)^{(t+1)/2} \\ &= \left(\frac{2}{\sqrt{5}}\right)^{t+1}. \end{aligned}$$

## 19.7 Matrix Norms

Recall that the operator norm of a matrix  $M$  (also called the 2-norm) is defined by

$$\|M\| = \max_{\mathbf{v}} \frac{\|M\mathbf{v}\|}{\|\mathbf{v}\|}.$$

The matrix norm measures how much a vector can increase in size when it is multiplied by  $M$ . When  $M$  is symmetric, the 2-norm is just the largest absolute value of an eigenvalue of  $M$  (prove this for yourself). It is also immediate that

$$\|M_1 M_2\| \leq \|M_1\| \|M_2\|.$$

You should also verify this yourself. As  $\mathbf{D}_X$ ,  $\mathbf{D}_Y$  and  $\mathbf{W}$  are symmetric, they each have norm 1.

**Warning 19.7.1.** While the largest eigenvalue of a walk matrix is 1, the norm of an asymmetric walk matrix can be larger than 1. For instance, consider the walk matrix of the path on 3 vertices. Verify that it has norm  $\sqrt{2}$ .

Our analysis rests upon the following bound on the norm of  $\mathbf{D}_X W$ .

**Lemma 19.7.2.**

$$\|\mathbf{D}_X \mathbf{W}\| \leq 1/5.$$

Let's see why this implies the theorem. For any set  $R$ , let  $Z_i$  be as defined above. As  $\mathbf{p}_0 = \mathbf{W} \mathbf{p}_0$ , we have

$$\mathbf{1}^T \mathbf{D}_{Z_t} \mathbf{W} \mathbf{D}_{Z_{t-1}} \mathbf{W} \cdots \mathbf{D}_{Z_1} \mathbf{W} \mathbf{D}_{Z_0} \mathbf{p}_0 = \mathbf{1}^T (\mathbf{D}_{Z_t} \mathbf{W}) (\mathbf{D}_{Z_{t-1}} \mathbf{W}) \cdots (\mathbf{D}_{Z_0} \mathbf{W}) \mathbf{p}_0.$$

Now,

$$\|\mathbf{D}_{Z_{t-1}} \mathbf{W}\| \leq \begin{cases} 1/5 & \text{for } i \in R, \text{ and} \\ 1 & \text{for } i \notin R. \end{cases}$$

So,

$$\|(\mathbf{D}_{Z_t} \mathbf{W}) (\mathbf{D}_{Z_{t-1}} \mathbf{W}) \cdots (\mathbf{D}_{Z_0} \mathbf{W})\| \leq (1/5)^{|R|}.$$

As  $\|\mathbf{p}_0\| = 1/\sqrt{n}$  and  $\|\mathbf{1}\| = \sqrt{n}$ , we may conclude

$$\begin{aligned} \mathbf{1}^T (\mathbf{D}_{Z_t} \mathbf{W}) (\mathbf{D}_{Z_{t-1}} \mathbf{W}) \cdots (\mathbf{D}_{Z_0} \mathbf{W}) \mathbf{p}_0 &\leq \|\mathbf{1}^T\| \|(\mathbf{D}_{Z_t} \mathbf{W}) (\mathbf{D}_{Z_{t-1}} \mathbf{W}) \cdots (\mathbf{D}_{Z_0} \mathbf{W}) \mathbf{p}_0\| \\ &\leq \|\mathbf{1}^T\| (1/5)^{|R|} \|\mathbf{p}_0\| \\ &= (1/5)^{|R|}. \end{aligned}$$

## 19.8 The norm of $\mathbf{D}_X \mathbf{W}$

*Proof of Lemma 19.7.2.* Let  $\mathbf{x}$  be any non-zero vector, and write

$$\mathbf{x} = c\mathbf{1} + \mathbf{y},$$

where  $\mathbf{1}^T \mathbf{y} = 0$ . We will show that  $\|\mathbf{D}_X \mathbf{W} \mathbf{x}\| \leq \|\mathbf{x}\|/5$ .

We know that the constant vectors are eigenvectors of  $\mathbf{W}$ . So,  $\mathbf{W}\mathbf{1} = \mathbf{1}$  and

$$\mathbf{D}_X \mathbf{W} \mathbf{1} = \chi_X.$$

This implies

$$\|\mathbf{D}_X \mathbf{W} c\mathbf{1}\| = c \|\chi_X\| = c\sqrt{|X|} \leq c\sqrt{n}/10.$$

We will now show that  $\|\mathbf{W}\mathbf{y}\| \leq \|\mathbf{y}\|/10$ . The easiest way to see this is to consider the matrix

$$\mathbf{W} - \mathbf{J}/n,$$

where we recall that  $\mathbf{J}$  is the all-1 matrix. This matrix is symmetric and all of its eigenvalues have absolute value at most 1/10. So, it has norm at most 1/10. Moreover,  $(\mathbf{W} - \mathbf{J}/n)\mathbf{y} = \mathbf{W}\mathbf{y}$ , which implies  $\|\mathbf{W}\mathbf{y}\| \leq \|\mathbf{y}\|/10$ . Another way to prove this is to expand  $\mathbf{y}$  in the eigenbasis of  $\mathbf{W}$ , as in the proof of Lemma 2.1.3.

Finally, as  $\mathbf{1}$  is orthogonal to  $\mathbf{y}$ ,

$$\|\mathbf{x}\| = \sqrt{c^2 n + \|\mathbf{y}\|^2}.$$

So,

$$\|\mathbf{D}_X \mathbf{W} \mathbf{x}\| \leq \|\mathbf{D}_X \mathbf{W} c\mathbf{1}\| + \|\mathbf{D}_X \mathbf{W} \mathbf{y}\| \leq c\sqrt{n}/10 + \|\mathbf{y}\|/10 \leq \|\mathbf{x}\|/10 + \|\mathbf{x}\|/10 \leq \|\mathbf{x}\|/5.$$

□

## 19.9 Conclusion

We finished lecture by observing that this is a very strange proof. When considering probabilities, it seems that it would be much more natural to sum them. But, here we consider 2-norms of probability vectors.

## References

- [ABN<sup>+</sup>92] Noga Alon, Jehoshua Bruck, Joseph Naor, Moni Naor, and Ron M. Roth. Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs. *IEEE Transactions on Information Theory*, 38(2):509–516, March 1992.
- [HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.
- [IZ89] R. Impagliazzo and D. Zuckerman. How to recycle random bits. In *30th annual IEEE Symposium on Foundations of Computer Science*, pages 248–253, 1989.
- [LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.
- [Mar88] G. A. Margulis. Explicit group theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators. *Problems of Information Transmission*, 24(1):39–46, July 1988.

## The Lovàsz - Simonovits Approach to Random Walks

Daniel A. Spielman

November 7, 2018

## 20.1 Introduction

**These notes are still very rough, and will be finished later.**

For a vector  $\mathbf{f}$  and an integer  $k$ , we define  $\mathbf{f}\{k\}$  to be the sum of the largest  $k$  entries of  $\mathbf{f}$ . For convenience, we define  $\mathbf{f}\{0\} = 0$ . Symbolically, you can define this by setting  $\pi$  to be a permutation for which

$$\mathbf{f}(\pi(1)) \geq \mathbf{f}(\pi(2)) \geq \dots \geq \mathbf{f}(\pi(n)),$$

and then setting

$$\mathbf{f}\{k\} = \sum_{i=1}^k \mathbf{f}(\pi(i)).$$

For real number  $x$  between 0 and  $n$ , we define  $\mathbf{f}\{x\}$  by making it be piece-wise linear between consecutive integers. This means that for  $x$  between integers  $k$  and  $k+1$ , the slope of  $\mathbf{f}\{\cdot\}$  at  $x$  is  $\mathbf{f}(\pi(k+1))$ . As these slopes are monotone nonincreasing, the function  $\mathbf{f}\{x\}$  is concave.

We will prove the following theorem of Lovàsz and Simonovits [LS90] on the behavior of  $\mathbf{Wf}$ .

**Theorem 20.1.1.** *Let  $\mathbf{W}$  be the transition matrix of the lazy random walk on a  $d$ -regular graph with conductance at least  $\phi$ . Let  $\mathbf{g} = \mathbf{Wf}$ . Then for all integers  $0 \leq k \leq n$*

$$\mathbf{g}\{k\} \leq \frac{1}{2} (\mathbf{f}\{k - \phi h\} + \mathbf{f}\{k + \phi h\}),$$

where  $h = \min(k, n - k)$ .

I remark that this theorem has a very clean extension to irregular, weighted graphs. I just present this version to simplify the exposition.

We can use this theorem to bound the rate of convergence of random walks in a graph. Let  $\mathbf{p}_t$  be the probability distribution of the walk after  $t$  steps, and plot the curves  $\mathbf{p}_t\{x\}$ . The theorem tells us that these curves lie beneath each other, and that each curve lies beneath a number of chords drawn across the previous. The walk is uniformly mixed when the curve reaches a straight line from  $(0, 0)$  to  $(n, 1)$ . This theorem tells us how quickly the walks approach the straight line.

Today, we will use the theorem to prove a variant of Cheeger's inequality.

## 20.2 Definitions and Elementary Observations

We believe that larger conductance should imply faster mixing. In the case of Theorem 20.1.1, it should imply lower curves. This is because wider chords lie beneath narrower ones.

**Claim 20.2.1.** *Let  $h(x)$  be a convex function, and let  $z > y > 0$ . Then,*

$$\frac{1}{2}(h(x-z) + h(x+z)) \leq \frac{1}{2}(h(x-y) + h(x+y)).$$

**Claim 20.2.2.** *Let  $\mathbf{f}$  be a vector, let  $k \in [0, n]$ , and let  $\alpha_1, \dots, \alpha_n$  be numbers between 0 and 1 such that*

$$\sum_i \alpha_i = k.$$

*Then,*

$$\sum_i \alpha_i \mathbf{f}(i) \leq \mathbf{f}\{k\}.$$

This should be obvious, and most of you proved something like this when solving problem 2 on homework 1. It is true because the way one would maximize this sum is by setting  $x$  to 1 for the largest values.

Throughout this lecture, we will only consider lazy random walks on regular graphs. For a set  $S$  and a vertex  $a$ , we define  $\gamma(a, S)$  to be the probability that a walk that is at vertex  $a$  moves to  $S$  in one step. If  $a$  is not in  $S$ , this equals one half the fraction of edges from  $a$  to  $S$ . It is one half because there is a one half probability that the walk stays at  $a$ . Similarly, if  $a$  is in  $S$ , then  $\gamma(a, S)$  equals one half plus one half the fraction of edges of  $a$  that end in  $S$ .

## 20.3 Warm up

We warm up by proving that the curves must lie under each other.

For a vector  $\mathbf{f}$  and a set  $S$ , we define

$$\mathbf{f}(S) = \sum_{a \in S} \mathbf{f}(a).$$

For every  $k$  there is at least one set  $S$  for which

$$\mathbf{f}(S) = \mathbf{f}\{k\}.$$

If the values of  $\mathbf{f}$  are distinct, then the set  $S$  is unique.

**Lemma 20.3.1.** *Let  $\mathbf{f}$  be a vector and let  $\mathbf{g} = \mathbf{W}\mathbf{f}$ . Then for every  $x \in [0, n]$ ,*

$$\mathbf{g}\{x\} \leq \mathbf{f}\{x\}.$$

*Proof.* As the function  $\mathbf{g}\{x\}$  is piecewise linear between integers, it suffices to prove it at integers  $k$ . Let  $k$  be an integer and let  $S$  be a set of size  $k$  for which  $\mathbf{f}(S) = \mathbf{f}\{k\}$ . As  $\mathbf{g} = \mathbf{W}\mathbf{f}$ ,

$$\mathbf{g}(S) = \sum_{a \in V} \gamma(a, S) \mathbf{f}(a).$$

As the graph is regular,

$$\sum_{a \in V} \gamma(a, S) = k.$$

Thus, Claim 20.2.2 implies

$$\sum_{a \in V} \gamma(a, S) \mathbf{f}(a) \leq \mathbf{f}\{k\}.$$

□

## 20.4 The proof

Recall that the conductance of a subset of vertices  $S$  in a  $d$ -regular graph is defined to be

$$\phi(S) \stackrel{\text{def}}{=} \frac{|\partial(S)|}{d \min(|S|, n - |S|)}.$$

Our proof of the main theorem improves the previous argument by exploiting the conductance through the following lemma.

**Lemma 20.4.1.** *Let  $S$  be any set of  $k$  vertices. Then*

$$\sum_{a \notin S} \gamma(a, S) = (\phi(S)/2) \min(k, n - k).$$

*Proof.* For  $a \notin S$ ,  $\gamma(a, S)$  equals half the fraction of the edges from  $a$  that land in  $S$ . And, the number of edges leaving  $S$  equals  $d\phi(S) \min(k, n - k)$ . □

**Lemma 20.4.2.** *Let  $\mathbf{W}$  be the transition matrix of the lazy random walk on a  $d$ -regular graph, and let  $\mathbf{g} = \mathbf{W}\mathbf{f}$ . For every set  $S$  of size  $k$  with conductance at least  $\phi$ ,*

$$\mathbf{g}(S) \leq \frac{1}{2} (\mathbf{f}\{k - \phi h\} + \mathbf{f}\{k + \phi h\}),$$

where  $h = \min(k, n - k)$ .

*Proof.* To ease notation, define  $\gamma(a) = \gamma(a, S)$ . We prove the theorem by rearranging the formula

$$\mathbf{g}(S) = \sum_{a \in V} \gamma(a) \mathbf{f}(a).$$

Recall that  $\sum_{a \in V} \gamma(a) = k$ .

For every vertex  $a$  define

$$\alpha(a) = \begin{cases} \gamma(a) - 1/2 & \text{if } a \in S \\ 0 & \text{if } a \notin S \end{cases} \quad \text{and} \quad \beta(a) = \begin{cases} 1/2 & \text{if } a \in S \\ \gamma(a) & \text{if } a \notin S. \end{cases}$$

As  $\alpha(a) + \beta(a) = \gamma(a)$ ,

$$\mathbf{g}(S) = \sum_{a \in V} \alpha(a) \mathbf{f}(a) + \sum_{a \in V} \beta(a) \mathbf{f}(a).$$

We now come to the point in the argument where we exploit the laziness of the random walk, which manifests as the fact that  $\gamma(a) \geq 1/2$  for  $a \in S$ , and so  $0 \leq \alpha(a) \leq 1/2$  for all  $a$ . Similarly,  $0 \leq \beta(a) \leq 1/2$  for all  $a$ . So, we can write

$$\sum_{a \in V} \alpha(a) \mathbf{f}(a) = \frac{1}{2} \sum_{a \in V} (2\alpha(a)) \mathbf{f}(a), \quad \text{and} \quad \sum_{a \in V} \beta(a) \mathbf{f}(a) = \frac{1}{2} \sum_{a \in V} (2\beta(a)) \mathbf{f}(a)$$

with all coefficients  $2\alpha(a)$  and  $2\beta(a)$  between 0 and 1. As

$$\sum_{a \in V} \beta(a) = \frac{k}{2} + \sum_{a \notin S} \gamma(a),$$

we can set

$$z = \sum_{a \notin S} \gamma(a)$$

and write

$$\sum_{a \in V} (2\alpha(a)) = k - 2z \quad \text{and} \quad \sum_{a \in V} (2\beta(a)) = k + 2z.$$

Lemma 20.4.1 implies that

$$z \geq \phi h/2.$$

By Claim 20.2.2,

$$\mathbf{g}(S) \leq \frac{1}{2} (\mathbf{f}\{k-z\} + \mathbf{f}\{k+z\}).$$

So, Claim 20.2.1 implies

$$\mathbf{g}(S) \leq \frac{1}{2} (\mathbf{f}\{k-\phi h\} + \mathbf{f}\{k+\phi h\}).$$

□

Theorem 20.1.1 follows by applying Lemma 20.4.2 to sets  $S$  for which  $\mathbf{f}(S) = \mathbf{f}\{k\}$ , for each integer  $k$  between 0 and  $n$ .

## 20.5 Andersen's proof of Cheeger's inequality

Reid Andersen observed that the technique of Lovàsz and Simonovits can be used to give a new proof of Cheeger's inequality. I will state and prove the result for the special case of  $d$ -regular graphs that we consider in this lecture. But, one can of course generalize this to irregular, weighted graphs.

**Theorem 20.5.1.** Let  $G$  be a  $d$ -regular graph with lazy random walk matrix  $\mathbf{W}$ , and let  $\omega_2 = 1 - \lambda$  be the second-largest eigenvalue of  $\mathbf{W}$ . Then there is a subset of vertices  $S$  for which

$$\phi(S) \leq \sqrt{8\lambda}.$$

*Proof.* Let  $\psi$  be the eigenvector corresponding to  $\omega_2$ . As  $\psi$  is orthogonal to the constant vectors,  $\psi\{n\} = 0$ . Define

$$k = \arg \max_{0 \leq k \leq n} \frac{\psi\{k\}}{\sqrt{\min(k, n-k)}}.$$

Then, set  $\gamma$  to be the maximum value obtained:

$$\gamma = \frac{\psi\{k\}}{\sqrt{\min(k, n-k)}}.$$

We will assume without loss of generality that  $k \leq n/2$ : if it is not then we replace  $\psi$  by  $-\psi$  to make it so and obtain the same  $\gamma$ . Now,  $\psi\{k\} = \gamma\sqrt{k}$ .

We let  $S$  be a set (there is probably only one) for which

$$\psi(S) = \psi\{k\}.$$

As  $\psi$  is an eigenvector with positive eigenvalue, we also know that

$$(\mathbf{W}\psi)(S) = \mathbf{W}\psi\{k\}.$$

We also know that

$$(\mathbf{W}\psi)(S) = (1 - \lambda)\psi(S) = (1 - \lambda)\gamma\sqrt{k}.$$

Let  $\phi$  be the conductance of  $S$ . Lemma 20.4.2 tells us that

$$(\mathbf{W}\psi)(S) \leq \frac{1}{2} (\psi\{k - \phi k\} + \psi\{k + \phi k\}).$$

By the construction of  $k$  and  $\gamma$  at the start of the proof, we know this quantity is at most

$$\frac{1}{2} \left( \gamma\sqrt{k - \phi k} + \gamma\sqrt{k + \phi k} \right) = \gamma\sqrt{k} \frac{1}{2} \left( \sqrt{1 - \phi} + \sqrt{1 + \phi} \right).$$

Combining the inequalities derived so far yields

$$(1 - \lambda) \leq \frac{1}{2} \left( \sqrt{1 - \phi} + \gamma\sqrt{1 + \phi} \right).$$

An examination of the Taylor series for the last terms reveals that

$$\frac{1}{2} \left( \sqrt{1 - \phi} + \gamma\sqrt{1 + \phi} \right) \leq 1 - \phi^2/8.$$

This implies  $\lambda \geq \phi^2/8$ , and thus  $\phi(S) \leq \sqrt{8\lambda}$ . □

## References

- [LS90] L. Lovàsz and M. Simonovits. The mixing rate of Markov chains, an isoperimetric inequality, and computing the volume. In IEEE, editor, *Proceedings: 31st Annual Symposium on Foundations of Computer Science: October 22–24, 1990, St. Louis, Missouri*, volume 1, pages 346–354, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 1990. IEEE Computer Society Press.

## Sparsification by Random Sampling

*Daniel A. Spielman*

November 12, 2018

## 21.1 Overview

Two weeks ago, we learned that expander graphs are sparse approximations of the complete graph. This week we will learn that every graph can be approximated by a sparse graph. Today, we will see how a sparse approximation can be obtained by careful random sampling: every graph on  $n$  vertices has an  $\epsilon$ -approximation with only  $O(\epsilon^{-2}n \log n)$  edges (a result of myself and Srivastava [SS11]). We will prove this using a matrix Chernoff bound due to Tropp [Tro12].

We originally proved this theorem using a concentration bound of Rudelson [Rud99]. This required an argument that used sampling with replacement. When I taught this result in 2012, I asked if one could avoid sampling with replacement. Nick Harvey pointed out to me the argument that avoids replacement that I am presenting today.

In the next lecture, we will see that the  $\log n$  term is unnecessary. In fact, almost every graph can be approximated by a sparse graph almost as well as the Ramanujan graphs approximate complete graphs.

## 21.2 Sparsification

For this lecture, I define a graph  $H$  to be an  $\epsilon$ -approximation of a graph  $G$  if

$$(1 - \epsilon)\mathbf{L}_G \preccurlyeq \mathbf{L}_H \preccurlyeq (1 + \epsilon)\mathbf{L}_G.$$

We will show that every graph  $G$  has a good approximation by a sparse graph. This is a very strong statement, as graphs that approximate each other have a lot in common. For example,

1. the effective resistance between all pairs of vertices are similar in the two graphs,
2. the eigenvalues of the graphs are similar,
3. the boundaries of all sets are similar, as these are given by  $\chi_S^T \mathbf{L}_G \chi_S$ , and
4. the solutions of linear equations in the two matrices are similar.

We will prove this by using a very simple random construction. We first carefully<sup>1</sup> choose a probability  $p_{a,b}$  for each edge  $(a, b)$ . We then include each edge  $(a, b)$  with probability  $p_{a,b}$ , independently.

---

<sup>1</sup>For those who can't stand the suspense, we reveal that we will choose the probabilities to be proportional to leverage scores of the edges.

If we do include edge  $(a, b)$ , we give it weight  $w_{a,b}/p_{a,b}$ . We will show that our choice of probabilities ensures that the resulting graph  $H$  has at most  $4n \ln n/\epsilon^2$  edges and is an  $\epsilon$  approximation of  $G$  with high probability.

The reason we employ this sort of sampling—blowing up the weight of an edge by dividing by the probability that we choose it—is that it preserves the matrix in expectation. Let  $\mathbf{L}_{a,b}$  denote the elementary Laplacian on edge  $(a, b)$  with weight 1, so that

$$\mathbf{L}_G = \sum_{(a,b) \in E} w_{a,b} \mathbf{L}_{a,b}.$$

We then have that

$$\mathbb{E} \mathbf{L}_H = \sum_{(a,b) \in E} p_{a,b} (w_{a,b}/p_{a,b}) \mathbf{L}_{a,b} = \mathbf{L}_G.$$

### 21.3 Matrix Chernoff Bounds

The main tool that we will use in our analysis is a theorem about the concentration of random matrices. These may be viewed as matrix analogs of the Chernoff bound that we saw in Lecture 5. These are a surprisingly recent development, with the first ones appearing in the work of Rudelson and Vershynin [Rud99, RV07] and Ahlswede and Winter [AW02]. The best present source for these bounds is Tropp [Tro12], in which the following result appears as Corollary 5.2.

**Theorem 21.3.1.** *Let  $\mathbf{X}_1, \dots, \mathbf{X}_m$  be independent random  $n$ -dimensional symmetric positive semidefinite matrices so that  $\|\mathbf{X}_i\| \leq R$  almost surely. Let  $\mathbf{X} = \sum_i \mathbf{X}_i$  and let  $\mu_{\min}$  and  $\mu_{\max}$  be the minimum and maximum eigenvalues of*

$$\mathbb{E} [\mathbf{X}] = \sum_i \mathbb{E} [\mathbf{X}_i].$$

Then,

$$\begin{aligned} \Pr \left[ \lambda_{\min} \left( \sum_i \mathbf{X}_i \right) \leq (1 - \epsilon) \mu_{\min} \right] &\leq n \left( \frac{e^{-\epsilon}}{(1 - \epsilon)^{1-\epsilon}} \right)^{\mu_{\min}/R}, & \text{for } 0 < \epsilon < 1, \text{ and} \\ \Pr \left[ \lambda_{\max} \left( \sum_i \mathbf{X}_i \right) \geq (1 + \epsilon) \mu_{\max} \right] &\leq n \left( \frac{e^{\epsilon}}{(1 + \epsilon)^{1+\epsilon}} \right)^{\mu_{\max}/R}, & \text{for } 0 < \epsilon. \end{aligned}$$

It is important to note that the matrices  $\mathbf{X}_1, \dots, \mathbf{X}_m$  can have different distributions. Also note that as the norms of these matrices get bigger, the bounds above become weaker. As the expressions above are not particularly easy to work with, we often use the following approximations.

$$\begin{aligned} \left( \frac{e^{-\epsilon}}{(1 - \epsilon)^{1-\epsilon}} \right) &\leq e^{-\epsilon^2/2}, & \text{for } 0 < \epsilon < 1, \text{ and} \\ \left( \frac{e^{\epsilon}}{(1 + \epsilon)^{1+\epsilon}} \right) &\leq e^{-\epsilon^2/3}, & \text{for } 0 < \epsilon < 1. \end{aligned}$$

Chernoff (and Hoeffding and Bernstein) bounds rarely come in exactly the form you want. Sometimes you can massage them into the needed form. Sometimes you need to prove your own. For this reason, you may some day want to spend a lot of time reading how these are proved.

## 21.4 The key transformation

Before applying the matrix Chernoff bound, we make a transformation that will cause  $\mu_{\min} = \mu_{\max} = 1$ .

For positive definite matrices  $\mathbf{A}$  and  $\mathbf{B}$ , we have

$$\mathbf{A} \preccurlyeq (1 + \epsilon)\mathbf{B} \iff \mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2} \preccurlyeq (1 + \epsilon)\mathbf{I}.$$

The same things holds for singular semidefinite matrices that have the same nullspace:

$$\mathbf{L}_H \preccurlyeq (1 + \epsilon)\mathbf{L}_G \iff \mathbf{L}_G^{+1/2}\mathbf{L}_H\mathbf{L}_G^{+1/2} \preccurlyeq (1 + \epsilon)\mathbf{L}_G^{+1/2}\mathbf{L}_G\mathbf{L}_G^{+1/2},$$

where  $\mathbf{L}_G^{+1/2}$  is the square root of the pseudo-inverse of  $\mathbf{L}_G$ . Let

$$\boldsymbol{\Pi} = \mathbf{L}_G^{+1/2}\mathbf{L}_G\mathbf{L}_G^{+1/2},$$

which is the projection onto the range of  $\mathbf{L}_G$ . We now know that  $\mathbf{L}_G$  is an  $\epsilon$ -approximation of  $\mathbf{L}_H$  if and only if  $\mathbf{L}_G^{+1/2}\mathbf{L}_H\mathbf{L}_G^{+1/2}$  is an  $\epsilon$ -approximation of  $\boldsymbol{\Pi}$ .

As multiplication by a fixed matrix is a linear operation and expectation commutes with linear operations,

$$\mathbb{E}\mathbf{L}_G^{+1/2}\mathbf{L}_H\mathbf{L}_G^{+1/2} = \mathbf{L}_G^{+1/2}(\mathbb{E}\mathbf{L}_H)\mathbf{L}_G^{+1/2} = \mathbb{E}\mathbf{L}_G^{+1/2}\mathbf{L}_G\mathbf{L}_G^{+1/2} = \boldsymbol{\Pi}.$$

So, we really just need to show that this random matrix is probably close to its expectation,  $\boldsymbol{\Pi}$ . It would probably help to pretend that  $\boldsymbol{\Pi}$  is in fact the identity, as it will make it easier to understand the analysis. In fact, you don't have to pretend: you could project all the vectors and matrices onto the span of  $\boldsymbol{\Pi}$  and carry out the analysis there.

## 21.5 The probabilities

Let

$$\mathbf{X}_{a,b} = \begin{cases} (w_{a,b}/p_{a,b})\mathbf{L}_G^{+1/2}\mathbf{L}_{(a,b)}\mathbf{L}_G^{+1/2} & \text{with probability } p_{a,b} \\ 0 & \text{otherwise,} \end{cases}$$

so that

$$\mathbf{L}_G^{+1/2}\mathbf{L}_H\mathbf{L}_G^{+1/2} = \sum_{(a,b) \in E} \mathbf{X}_{a,b}.$$

We will choose the probabilities to be

$$p_{a,b} \stackrel{\text{def}}{=} \frac{1}{R}w_{a,b} \left\| \mathbf{L}_G^{+1/2}\mathbf{L}_{(a,b)}\mathbf{L}_G^{+1/2} \right\|,$$

for an  $R$  to be chosen later. Thus, when edge  $(a, b)$  is chosen,  $\|X_{a,b}\| = R$ . Making this value uniform for every edge optimizes one part of Theorem 21.3.1.

You may wonder what we should do if one of these probabilities  $p_{a,b}$  exceeds one. There are many ways of addressing this issue. For now, pretend that it does not happen. We will then explain how to deal with this at the end of lecture.

Recall that the leverage score of edge  $(a, b)$  written  $\ell_{a,b}$  was defined in Lecture 14 to be the weight of an edge times the effective resistance between its endpoints:

$$\ell_{a,b} = w_{a,b}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}_G^+ (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b).$$

To see the relation between the leverage score and  $p_{a,b}$ , compute

$$\begin{aligned} \left\| \mathbf{L}_G^{+/2} \mathbf{L}_{(a,b)} \mathbf{L}_G^{+/2} \right\| &= \left\| \mathbf{L}_G^{+/2} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}_G^{+/2} \right\| \\ &= \left\| (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}_G^{+/2} \mathbf{L}_G^{+/2} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \right\| \\ &= (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}_G^+ (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \\ &= \text{R}_{\text{eff}}(a, b). \end{aligned}$$

As we can quickly approximate the effective resistance of every edge, we can quickly compute sufficient probabilities.

Recall that the leverage score of an edge equals the probability that the edge appears in a random spanning tree. As every spanning tree has  $n - 1$  edges, this means that the sum of the leverage scores is  $n - 1$ , and thus

$$\sum_{(a,b) \in E} p_{a,b} = \frac{n-1}{R} \leq \frac{n}{R}.$$

This is a very clean bound on the expected number of edges in  $H$ . One can use a Chernoff bound (on real variables rather than matrices) to prove that it is exponentially unlikely that the number of edges in  $H$  is more than any small multiple of this.

For your convenience, I recall another proof that the sum of the leverage scores is  $n - 1$ :

$$\begin{aligned}
\sum_{(a,b) \in E} \ell_{a,b} &= \sum_{(a,b) \in E} w_{a,b} R_{\text{eff}}(a, b) \\
&= \sum_{(a,b) \in E} w_{a,b} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \mathbf{L}_G^+ (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \\
&= \sum_{(a,b) \in E} w_{a,b} \text{Tr} (\mathbf{L}_G^+ (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T) \\
&= \text{Tr} \left( \sum_{(a,b) \in E} \mathbf{L}_G^+ w_{a,b} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \right) \\
&= \text{Tr} \left( \mathbf{L}_G^+ \sum_{(a,b) \in E} w_{a,b} \mathbf{L}_{a,b} \right) \\
&= \text{Tr} (\mathbf{L}_G^+ \mathbf{L}_G) \\
&= \text{Tr} (\mathbf{\Pi}) \\
&= n - 1.
\end{aligned}$$

## 21.6 The analysis

We will choose

$$R = \frac{\epsilon^2}{3.5 \ln n}.$$

Thus, the number of edges in  $H$  will be at most  $4(\ln n)\epsilon^{-2}$  with high probability.

We have

$$\sum_{(a,b) \in E} \mathbb{E} \mathbf{X}_{a,b} = \mathbf{\Pi}.$$

It remains to show that it is unlikely to deviate from this by too much.

We first consider the case in which  $p_{(a,b)} \leq 1$  for all edges  $(a, b)$ . If this is the case, then Theorem 21.3.1 tells us that

$$\Pr \left[ \sum_{a,b} \mathbf{X}_{a,b} \geq (1 + \epsilon) \mathbf{\Pi} \right] \leq n \exp(-\epsilon^2 / 3R) = n \exp(-(3.5/3) \ln n) = n^{-1/6}.$$

For the lower bound, we need to remember that we can just work orthogonal to the all-1s vector, and so treat the smallest eigenvalue of  $\mathbf{\Pi}$  as 1. We then find that

$$\Pr \left[ \sum_{a,b} \mathbf{X}_{a,b} \leq (1 - \epsilon) \mathbf{\Pi} \right] \leq n \exp(-\epsilon^2 / 2R) = n \exp(-(3.5/2) \ln n) = n^{-3/2},$$

We finally return to deal with the fact that there might be some edges for which  $p_{a,b} \geq 1$  and so definitely appear in  $H$ . There are two natural ways to deal with these—one that is easiest algorithmically and one that simplifies the proof. The algorithmically natural way to handle these is to simply include these edges in  $H$ , and remove them from the analysis above. This requires a small adjustment to the application of the Matrix Chernoff bound, but it does go through.

From the perspective of the proof, the simplest way to deal with these is to split each such  $\mathbf{X}_{a,b}$  into many independent random edges:  $k = \lfloor \ell_{a,b}/R \rfloor$  that appear with probability exactly 1, and one more that appears with probability  $\ell_{a,b}/R - k$ . This does not change the expectation of their sum, or the expected number of edges once we remember to add together the weights of edges that appear multiple times. The rest of the proof remains unchanged.

## 21.7 Open Problem

If I have time in class, I will sketch a way to quickly approximate the effective resistances of every edge in the graph. The basic idea, which can be found in [SS11] and which is carried out better in [KLP12], is that we can compute the effective resistance of an edge  $(a, b)$  from the solution to a logarithmic number of systems of random linear equations in  $\mathbf{L}_G$ . That is, after solving a logarithmic number of systems of linear equations in  $\mathbf{L}_G$ , we have information from which we can estimate all of the effective resistances.

In order to sparsify graphs, we do not actually need estimates of effective resistances that are always accurate. We just need a way to identify many edges of low effective resistance, without listing any that have high effective resistance. I believe that better algorithms for doing this remain to be found. Current fast algorithms that make progress in this direction and that exploit such estimates may be found in [KLP12, Kou14, CLM<sup>+</sup>15, LPS15]. These, however, rely on fast Laplacian equation solvers. It would be nice to be able to estimate effective resistances without these. A step in this direction was recently taken in the works [CGP<sup>+</sup>18, LSY18], which quickly decompose graphs into the union of short cycles plus a few edges.

## References

- [AW02] R. Ahlswede and A. Winter. Strong converse for identification via quantum channels. *Information Theory, IEEE Transactions on*, 48(3):569–579, 2002.
- [CGP<sup>+</sup>18] Timothy Chu, Yu Gao, Richard Peng, Sushant Sachdeva, Saurabh Sawlani, and Junxing Wang. Graph sparsification, spectral sketches, and faster resistance computation, via short cycle decompositions. *arXiv preprint arXiv:1805.12051*, 2018.
- [CLM<sup>+</sup>15] Michael B Cohen, Yin Tat Lee, Cameron Musco, Christopher Musco, Richard Peng, and Aaron Sidford. Uniform sampling for matrix approximation. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, pages 181–190. ACM, 2015.

- [KLP12] Ioannis Koutis, Alex Levin, and Richard Peng. Improved spectral sparsification and numerical algorithms for sdd matrices. In *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science)*, volume 14, pages 266–277. LIPIcs, 2012.
- [Kou14] Ioannis Koutis. Simple parallel and distributed algorithms for spectral graph sparsification. In *Proceedings of the 26th ACM Symposium on Parallelism in Algorithms and Architectures, SPAA '14*, pages 61–66, New York, NY, USA, 2014. ACM.
- [LPS15] Yin Tat Lee, Richard Peng, and Daniel A. Spielman. Sparsified cholesky solvers for SDD linear systems. *CoRR*, abs/1506.08204, 2015.
- [LSY18] Yang P Liu, Sushant Sachdeva, and Zejun Yu. Short cycles via low-diameter decompositions. *arXiv preprint arXiv:1810.05143*, 2018.
- [Rud99] M. Rudelson. Random vectors in the isotropic position,. *Journal of Functional Analysis*, 164(1):60 – 72, 1999.
- [RV07] Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *J. ACM*, 54(4):21, 2007.
- [SS11] D.A. Spielman and N. Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.
- [Tro12] Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.

## Linear Sized Sparsifiers

*Daniel A. Spielman*

November 14, 2018

## 22.1 Overview

In this lecture, we will prove a slight simplification of the main result of [BSS12, BSS14]. This will tell us that every graph with  $n$  vertices has an  $\epsilon$ -approximation with approximately  $4\epsilon^{-2}n$  edges. To translate this into a relation between approximation quality and average degree, note that such a graph has average degree  $d_{ave} = 8\epsilon^{-2}$ . So,

$$\epsilon \approx \frac{2\sqrt{2}}{\sqrt{d}},$$

which is about twice what you would get from a Ramanujan graph. Interestingly, this result even works for average degree just a little bit more than 1.

## 22.2 Turning edges into vectors

In the last lecture, we considered the Laplacian matrix of a graph  $G$  times the square root of the pseudoinverse on either side. That is,

$$\mathbf{L}_G^{+/-} \left( \sum_{(a,b) \in E} w_{a,b} \mathbf{L}_{(a,b)} \right) \mathbf{L}_G^{+/-}.$$

Today, it will be convenient to view this as a sum of outer products of vectors. Set

$$\mathbf{v}_{(a,b)} = \sqrt{w_{a,b}} \mathbf{L}_G^{+/-} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b).$$

Then,

$$\mathbf{L}_G^{+/-} \left( \sum_{(a,b) \in E} w_{a,b} \mathbf{L}_{(a,b)} \right) \mathbf{L}_G^{+/-} = \sum_{(a,b) \in E} \mathbf{v}_{(a,b)} \mathbf{v}_{(a,b)}^T = \mathbf{\Pi},$$

where we recall that  $\mathbf{\Pi} = \frac{1}{n} \mathbf{L}_{K_n}$  is the projection orthogonal to the constant vectors.

The problem of sparsification is then the problem of finding a small subset of these vectors,  $S \subseteq E$ , along with scaling factors,  $c : S \rightarrow \mathbb{R}$ , so that

$$(1 - \epsilon) \mathbf{\Pi} \preccurlyeq \sum_{(a,b) \in S} c_{a,b} \mathbf{v}_{(a,b)} \mathbf{v}_{(a,b)}^T \preccurlyeq (1 + \epsilon) \mathbf{\Pi}$$

If we project onto the span of the Laplacian, then the sum of the outer products of vectors  $\mathbf{v}_{(a,b)}$  becomes the identity, and our goal is to find a set  $S$  and scaling factors  $c_{a,b}$  so that

$$(1 - \epsilon)\mathbf{I} \preccurlyeq \sum_{(a,b) \in S} c_{a,b} \mathbf{v}_{(a,b)} \mathbf{v}_{(a,b)}^T \preccurlyeq (1 + \epsilon)\mathbf{I}.$$

That is, so that all the eigenvalues of the matrix in the middle lie between  $(1 - \epsilon)$  and  $(1 + \epsilon)$ .

## 22.3 The main theorem

**Theorem 22.3.1.** *Let  $\mathbf{v}_1, \dots, \mathbf{v}_m$  be vectors in  $\mathbb{R}^n$  so that*

$$\sum_i \mathbf{v}_i \mathbf{v}_i^T = \mathbf{I}.$$

*Then, for every  $\epsilon > 0$  there exists a set  $S$  along with scaling factors  $c_i$  so that*

$$(1 - \epsilon)^2 \mathbf{I} \preccurlyeq \sum_{i \in S} c_i \mathbf{v}_i \mathbf{v}_i^T \preccurlyeq (1 + \epsilon)^2 \mathbf{I},$$

*and*

$$|S| \leq \lceil n/\epsilon^2 \rceil.$$

The condition that the sum of the outer products of the vectors sums to the identity has a name, *isotropic position*. I now mention one important property of vectors in isotropic position

**Lemma 22.3.2.** *Let  $\mathbf{v}_1, \dots, \mathbf{v}_m$  be vectors in isotropic position. Then, for every matrix  $\mathbf{M}$ ,*

$$\sum_i \mathbf{v}_i^T \mathbf{M} \mathbf{v}_i = \text{Tr}(\mathbf{M}).$$

*Proof.* We have

$$\mathbf{v}^T \mathbf{M} \mathbf{v} = \text{Tr}(\mathbf{v} \mathbf{v}^T \mathbf{M}),$$

so

$$\sum_i \mathbf{v}_i^T \mathbf{M} \mathbf{v}_i = \sum_i \text{Tr}(\mathbf{v}_i \mathbf{v}_i^T \mathbf{M}) = \text{Tr}\left(\left(\sum_i \mathbf{v}_i \mathbf{v}_i^T\right) \mathbf{M}\right) = \text{Tr}(\mathbf{I} \mathbf{M}) = \text{Tr}(\mathbf{M}).$$

□

Today, we will prove that we can find a set of  $6n$  vectors for which all eigenvalues lie between  $1n$  and  $13n$ . If you divide all scaling factors by  $\sqrt{13}n$ , this puts the eigenvalues between  $1/\sqrt{13}$  and  $\sqrt{13}$ . You can tighten the argument to prove Theorem 22.3.1.

We will prove this theorem by an iterative argument in which we choose one vector at a time to add to the set  $S$ . We will set the scaling factor of a vector when we add it to  $S$ . It is possible that we will add a vector to  $S$  more than once, in which case we will increase its scaling factor each

time. Throughout the argument we will maintain the invariant that the eigenvalues of the scaled sum of outer products is in the interval  $[l, u]$ , where  $l$  and  $u$  are quantities that will change with each addition to  $S$ . At the start of the algorithm, when  $S$  is empty, we will have

$$l_0 = -n \quad \text{and} \quad u_0 = n.$$

Every time we add a vector to  $S$ , we increase  $l$  by  $\delta_L$  and  $u$  by  $\delta_U$ , where

$$\delta_L = 1/3 \quad \text{and} \quad \delta_U = 2.$$

After we have done this  $6n$  times, we will have  $l = n$  and  $u = 13n$ .

## 22.4 Rank-1 updates

We will need to understand what happens to a matrix when we add the outer product of a vector.

**Theorem 22.4.1** (Sherman-Morrison). *Let  $\mathbf{A}$  be a nonsingular symmetric matrix and let  $\mathbf{v}$  be a vector and let  $c$  be a real number. Then,*

$$(\mathbf{A} - c\mathbf{v}\mathbf{v}^T)^{-1} = \mathbf{A}^{-1} + c \frac{\mathbf{A}^{-1}\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1}}{1 - c\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}}.$$

*Proof.* The easiest way to prove this is to multiply it out, gathering  $\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}$  terms into scalars:

$$\begin{aligned} (\mathbf{A} - c\mathbf{v}\mathbf{v}^T) \left( \mathbf{A}^{-1} + c \frac{\mathbf{A}^{-1}\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1}}{1 - c\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}} \right) &= \mathbf{I} - c\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1} + c \frac{\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1}}{1 - c\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}} - c^2 \frac{\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1}}{1 - c\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}} \\ &= \mathbf{I} - c\mathbf{v}\mathbf{v}^T\mathbf{A}^{-1} \left( 1 - \frac{1}{1 - c\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}} + \frac{c\mathbf{v}^T\mathbf{A}\mathbf{v}}{1 - c\mathbf{v}^T\mathbf{A}^{-1}\mathbf{v}} \right) \\ &= \mathbf{I}. \end{aligned}$$

□

## 22.5 Barrier Function Arguments

To prove the main theorem we need a good way to measure progress. We would like to keep all the eigenvalues of the matrix we have constructed at any point to lie in a nice range. But, more than that, we need them to be nicely distributed within this range. To enforce this, we need to measure how close the eigenvalues are to the limits.

Let  $\mathbf{A}$  be a symmetric matrix with eigenvalues  $\lambda_1 \leq \dots \leq \lambda_n$ . If  $u$  is larger than all of the eigenvalues of  $\mathbf{A}$ , then we call  $u$  an upper bound on  $\mathbf{A}$ . To make this notion quantitative, we define the *upper barrier function*

$$\Phi^u(\mathbf{A}) = \sum_i \frac{1}{u - \lambda_i}.$$

This is positive for all upper bounds  $u$ , goes to infinity as  $u$  approaches the largest eigenvalue, decreases as  $u$  grows, and is convex for  $u > \lambda_n$ . In particular, we will use

$$\Phi^{u+\delta}(\mathbf{A}) < \Phi^u(\mathbf{A}), \quad \text{for } \delta > 0. \quad (22.1)$$

Also, observe that

$$\lambda_n \leq u - 1/\Phi^u(\mathbf{A}). \quad (22.2)$$

We will exploit the following formula for the upper barrier function:

$$\Phi^u(\mathbf{A}) = \text{Tr}((u\mathbf{I} - \mathbf{A})^{-1}).$$

For a lower bound on the eigenvalues  $l$ , we will define an analogous lower barrier function

$$\Phi_l(\mathbf{A}) = \sum_i \frac{1}{\lambda_i - l} = \text{Tr}((\mathbf{A} - l\mathbf{I})^{-1}).$$

This is positive whenever  $l$  is smaller than all the eigenvalues, goes to infinity as  $l$  approaches the smallest eigenvalue, and decreases as  $l$  becomes smaller. In particular,

$$l + 1/\Phi_l(\mathbf{A}) \leq \lambda_1. \quad (22.3)$$

The analog of (22.1) is the following.

**Claim 22.5.1.** *Let  $l$  be a lower bound on  $\mathbf{A}$  and let  $\delta < 1/\Phi_l(\mathbf{A})$ . Then,*

$$\Phi_{l+\delta}(\mathbf{A}) \leq \frac{1}{1/\Phi_l(\mathbf{A}) - \delta}.$$

Note that this inequality is an equality when  $A$  is one-dimensional. In that case,

$$\frac{1}{\lambda_1 - l - \delta} = \frac{1}{1/(1/\lambda_1 - l) - \delta}.$$

*Proof.* After rearranging terms, we see that the inequality is equivalent to

$$\Phi_{l+\delta}(\mathbf{A}) - \Phi_l(\mathbf{A}) \leq \delta \Phi_{l+\delta}(\mathbf{A}) \Phi_l(\mathbf{A}).$$

We then prove this by expanding in the eigenvalues, keeping in mind that all the terms  $\lambda_i - l - \delta$  are positive:

$$\begin{aligned} \Phi_{l+\delta}(\mathbf{A}) - \Phi_l(\mathbf{A}) &= \sum_i \frac{1}{\lambda_i - l - \delta} - \sum_i \frac{1}{\lambda_i - l} \\ &= \sum_i \frac{\delta}{(\lambda_i - l - \delta)(\lambda_i - l)} \\ &\leq \delta \left( \sum_i \frac{1}{(\lambda_i - l - \delta)} \right) \left( \sum_i \frac{1}{(\lambda_i - l)} \right). \end{aligned}$$

□

Initially, we will have

$$\Phi_{l_0}(0) = \Phi_{-n}(0) = 1 \quad \text{and} \quad \Phi^{u_0}(0) = \Phi^n(0) = 1.$$

## 22.6 Barrier Function Updates

The most important thing to understand about the barrier functions is how they change when we add a vector to  $S$ . The Sherman-Morrison theorem tells us that happens when we change  $\mathbf{A}$  to  $\mathbf{A} + c\mathbf{v}\mathbf{v}^T$ :

$$\begin{aligned}\Phi^u(\mathbf{A} + c\mathbf{v}\mathbf{v}^T) &= \text{Tr}((u\mathbf{I} - \mathbf{A} - c\mathbf{v}\mathbf{v}^T)^{-1}) \\ &= \text{Tr}((u\mathbf{I} - \mathbf{A})^{-1}) + c \frac{\text{Tr}((u\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}\mathbf{v}^T(u\mathbf{I} - \mathbf{A})^{-1})}{1 - c\mathbf{v}^T(u\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}} \\ &= \Phi^u(\mathbf{A}) + c \frac{\text{Tr}(\mathbf{v}^T(u\mathbf{I} - \mathbf{A})^{-1}(u\mathbf{I} - \mathbf{A})^{-1}\mathbf{v})}{1 - c\mathbf{v}^T(u\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}} \\ &= \Phi^u(\mathbf{A}) + c \frac{\mathbf{v}^T(u\mathbf{I} - \mathbf{A})^{-2}\mathbf{v}}{1 - c\mathbf{v}^T(u\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}}.\end{aligned}$$

This increases the upper barrier function, and we would like to counteract this increase by increasing  $u$  at the same time. If we advance  $u$  to  $\hat{u} = u + \delta_U$ , then we find

$$\begin{aligned}\Phi^{u+\delta_U}(\mathbf{A} + c\mathbf{v}\mathbf{v}^T) &= \Phi^{u+\delta_U}(\mathbf{A}) + c \frac{\mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-2}\mathbf{v}}{1 - c\mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}} \\ &= \Phi^u(\mathbf{A}) - (\Phi^u(\mathbf{A}) - \Phi^{u+\delta_U}(\mathbf{A})) + \frac{\mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-2}\mathbf{v}}{1/c - \mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}}.\end{aligned}$$

We would like for this to be less than  $\Phi^u(\mathbf{A})$ . If we commit to how much we are going to increase  $u$ , then this gives an upper bound on how large  $c$  can be. We want

$$(\Phi^u(\mathbf{A}) - \Phi^{u+\delta_U}(\mathbf{A})) \geq \frac{\mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-2}\mathbf{v}}{1/c - \mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}},$$

which is equivalent to

$$\frac{1}{c} \geq \frac{\mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-2}\mathbf{v}}{(\Phi^u(\mathbf{A}) - \Phi^{u+\delta_U}(\mathbf{A}))} + \mathbf{v}^T(\hat{u}\mathbf{I} - \mathbf{A})^{-1}\mathbf{v}.$$

Define

$$\mathbf{U}_A = \frac{((u + \delta_u)\mathbf{I} - \mathbf{A})^{-2}}{(\Phi^u(\mathbf{A}) - \Phi^{u+\delta_U}(\mathbf{A}))} + ((u + \delta_u)\mathbf{I} - \mathbf{A})^{-1}.$$

We have established a clean condition for when we can add  $c\mathbf{v}\mathbf{v}^T$  to  $S$  and increase  $u$  by  $\delta_U$  without increasing the upper barrier function.

**Lemma 22.6.1.** *If*

$$\frac{1}{c} \geq \mathbf{v}^T \mathbf{U}_A \mathbf{v},$$

*then*

$$\Phi^{u+\delta_U}(\mathbf{A} + c\mathbf{v}\mathbf{v}^T) \leq \Phi^u(\mathbf{A}).$$

The miracle in the above formula is that the condition in the lemma just involves the vector  $\mathbf{v}$  as the argument of a quadratic form.

We also require the following analog for the lower barrier function. The difference is that increasing  $l$  by setting  $\hat{l} = l + \delta_L$  increases the barrier function, and adding a vector decreases it.

**Lemma 22.6.2.** *Define*

$$\mathbf{L}_A = \frac{(\mathbf{A} - \hat{l}\mathbf{I})^{-2}}{(\Phi_{l+\delta_L}(\mathbf{A}) - \Phi_l(\mathbf{A}))} - (\mathbf{A} - \hat{l}\mathbf{I})^{-1}.$$

If

$$\frac{1}{c} \leq \mathbf{v}^T \mathbf{L}_A \mathbf{v},$$

then

$$\Phi_{l+\delta_L}(\mathbf{A} + c\mathbf{v}\mathbf{v}^T) \leq \Phi_l(\mathbf{A}).$$

If we fix the vector  $\mathbf{v}$  and an increment  $\delta_L$ , then this gives a lower bound on the scaling factor by which we need to multiply it for the lower barrier function not to increase.

## 22.7 The inductive argument

It remains to show that there exists a vector  $\mathbf{v}$  and a scaling factor  $c$  so that

$$\Phi^{u+\delta_U}(\mathbf{A} + c\mathbf{v}\mathbf{v}^T) \leq \Phi^u(\mathbf{A}) \quad \text{and} \quad \Phi_{l+\delta_L}(\mathbf{A} + c\mathbf{v}\mathbf{v}^T) \leq \Phi_l(\mathbf{A}).$$

That is, we need to show that there is a vector  $\mathbf{v}_i$  so that

$$\mathbf{v}_i^T \mathbf{U}_A \mathbf{v}_i \leq \mathbf{v}_i^T \mathbf{L}_A \mathbf{v}_i.$$

Once we know this, we can set  $c$  so that

$$\mathbf{v}_i^T \mathbf{U}_A \mathbf{v}_i \leq \frac{1}{c} \leq \mathbf{v}_i^T \mathbf{L}_A \mathbf{v}_i.$$

**Lemma 22.7.1.**

$$\sum_i \mathbf{v}_i^T \mathbf{U}_A \mathbf{v}_i \leq \frac{1}{\delta_U} + \Phi_u(\mathbf{A}).$$

*Proof.* By Lemma 22.3.2, we know

$$\sum_i \mathbf{v}_i^T \mathbf{U}_A \mathbf{v}_i = \text{Tr}(\mathbf{U}_A).$$

To bound this, we break it into two parts

$$\frac{\text{Tr}((\hat{u}\mathbf{I} - \mathbf{A})^{-2})}{(\Phi^u(\mathbf{A}) - \Phi^{u+\delta_U}(\mathbf{A}))}$$

and

$$\text{Tr}((\hat{u}\mathbf{I} - \mathbf{A})^{-1}).$$

The second term is easiest

$$\text{Tr}((\hat{u}\mathbf{I} - \mathbf{A})^{-1}) = \Phi^{u+\delta}(\mathbf{A}) \leq \Phi^u(\mathbf{A}).$$

To bound the first term, consider the derivative of the barrier function with respect to  $u$ :

$$\frac{\partial}{\partial u} \Phi^u(\mathbf{A}) = \frac{\partial}{\partial u} \sum_i \frac{1}{u - \lambda_i} = - \sum_i \left( \frac{1}{u - \lambda_i} \right)^2 = -\text{Tr}(\hat{u}\mathbf{I} - \mathbf{A})^{-2}.$$

As  $\Phi^u(\mathbf{A})$  is convex in  $u$ , we may conclude that

$$\Phi^u(\mathbf{A}) - \Phi^{u+\delta_U}(\mathbf{A}) \geq -\delta_U \frac{\partial}{\partial u} \Phi^{u+\delta_u}(\mathbf{A}) = \delta_U \text{Tr}(\hat{u}\mathbf{I} - \mathbf{A})^{-2}.$$

□

The analysis for the lower barrier is similar, but the second term is slightly more complicated.

**Lemma 22.7.2.**

$$\sum_i \mathbf{v}_i^T \mathbf{L}_{\mathbf{A}} \mathbf{v}_i \geq \frac{1}{\delta_L} - \frac{1}{1/\Phi_l(\mathbf{A}) - \delta_L}.$$

*Proof.* As before, we bound

$$\frac{\text{Tr}((\mathbf{A} - (l + \delta_L \mathbf{I}))^{-2})}{\Phi_{l+\delta_L}(\mathbf{A}) - \Phi_l(\mathbf{A})}$$

by recalling that

$$\frac{\partial}{\partial l} \Phi_l(\mathbf{A}) = \text{Tr}(\mathbf{A} - l\mathbf{I})^{-2}.$$

As  $\Phi_l(\mathbf{A})$  is convex in  $l$ , we have

$$\Phi_{l+\delta_L}(\mathbf{A}) - \Phi_l(\mathbf{A}) \leq \delta_L \frac{\partial}{\partial l} \Phi_{l+\delta_L}(\mathbf{A}) = \delta_L \text{Tr}(\mathbf{A} - (l + \delta_L) \mathbf{I})^{-2}.$$

To bound the other term, we use Claim 22.5.1 to prove

$$\text{Tr}((\mathbf{A} - (l + \delta_L \mathbf{I}))^{-1}) \leq \frac{1}{1/\Phi_l(\mathbf{A}) - \delta_L}.$$

□

So, for there to exist a  $\mathbf{v}_i$  that we can add to  $S$  with scale factor  $c$  so that neither barrier function increases, we just need that

$$\frac{1}{\delta_U} + \Phi^u(\mathbf{A}) \leq \frac{1}{\delta_L} - \frac{1}{1/\Phi_l(\mathbf{A}) - \delta_L}.$$

If this holds, then there is a  $\mathbf{v}_i$  so that

$$\mathbf{v}_i \mathbf{U}_A \mathbf{v}_i \leq \mathbf{v}_i \mathbf{L}_A \mathbf{v}_i.$$

We then set  $c$  so that

$$\mathbf{v}_i \mathbf{U}_A \mathbf{v}_i \leq \frac{1}{c} \leq \mathbf{v}_i \mathbf{L}_A \mathbf{v}_i.$$

We now finish the proof by checking that the numbers I gave earlier satisfy the necessary conditions. At the start both barrier functions are less than 1, and we need to show that this holds throughout the algorithm. At every step, we will have by induction

$$\frac{1}{\delta_U} + \Phi_u(\mathbf{A}) \leq \frac{1}{2} + 1 = \frac{3}{2},$$

and

$$\frac{1}{\delta_L} - \frac{1}{1/\Phi_l(\mathbf{A}) - \delta_L} \geq 3 - \frac{1}{1 - 1/3} = \frac{3}{2}.$$

So, there is always a  $\mathbf{v}_i$  that we can add to  $S$  and a scaling factor  $c$  so that both barrier function remain upper bounded by 1.

If we now do this for  $6n$  steps, we will have

$$l = -n + 6n/3 = n \quad \text{and} \quad u = n + 2 \cdot 6n = 13n.$$

The bound stated at the beginning of the lecture comes from tightening the analysis. In particular, it is possible to improve Lemma 22.7.2 so that it says

$$\sum_i \mathbf{v}_i^T \mathbf{L}_A \mathbf{v}_i \geq \frac{1}{\delta_L} - \frac{1}{1/\Phi_l(\mathbf{A})}.$$

I recommend the paper for details.

## 22.8 Progress and Open Problems

- It is possible to generalize this result to sums of positive semidefinite matrices, instead of outer products of vectors [dCSHS11].
- It is now possible to compute sparsifiers that are almost this good in something close to linear time. [AZLO15, LS15].
- Given last lecture, it seems natural to conjecture that the scaling factors of edges should be proportional to their weights times effective resistances. Similarly, one might conjecture that if all vectors  $\mathbf{v}_i$  have the same norm, then the scaling factors are unnecessary. This is true, but not obvious. In fact, it is essentially equivalent to the Kadison-Singer problem [MSS14, MSS15].

## References

- [AZLO15] Zeyuan Allen-Zhu, Zhenyu Liao, and Lorenzo Orecchia. Spectral sparsification and regret minimization beyond matrix multiplicative updates. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 237–245. ACM, 2015.
- [BSS12] Joshua Batson, Daniel A Spielman, and Nikhil Srivastava. Twice-Ramanujan sparsifiers. *SIAM Journal on Computing*, 41(6):1704–1721, 2012.
- [BSS14] Joshua Batson, Daniel A Spielman, and Nikhil Srivastava. Twice-ramanujan sparsifiers. *SIAM Review*, 56(2):315–334, 2014.
- [dCSHS11] Marcel K. de Carli Silva, Nicholas J. A. Harvey, and Cristiane M. Sato. Sparse sums of positive semidefinite matrices. *CoRR*, abs/1107.0088, 2011.
- [LS15] Yin Tat Lee and He Sun. Constructing linear-sized spectral sparsification in almost-linear time. *arXiv preprint arXiv:1508.03261*, 2015.
- [MSS14] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Ramanujan graphs and the solution of the Kadison-Singer problem. In *Proceedings of the International Congress of Mathematicians*, 2014.
- [MSS15] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Interlacing families II: Mixed characteristic polynomials and the Kadison-Singer problem. *Ann. of Math.*, 182-1:327–350, 2015.

## Iterative solvers for linear equations

*Daniel A. Spielman*

November 26, 2018

## 23.1 Overview

In this and the next lecture, I will discuss iterative algorithms for solving linear equations in positive semidefinite matrices, with an emphasis on Laplacians. Today's lecture will cover Richardson's first-order iterative method and the Chebyshev method.

## 23.2 Why iterative methods?

One is first taught to solve linear systems like

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

by *direct methods* such as Gaussian elimination, computing the inverse of  $\mathbf{A}$ , or the *LU* factorization. However, all of these algorithms can be very slow. This is especially true when  $\mathbf{A}$  is sparse. Just writing down the inverse takes  $O(n^2)$  space, and computing the inverse takes  $O(n^3)$  time if we do it naively. This might be OK if  $\mathbf{A}$  is dense. But, it is very wasteful if  $\mathbf{A}$  only has  $O(n)$  non-zero entries.

In general, we prefer algorithms whose running time is proportional to the number of non-zero entries in the matrix  $\mathbf{A}$ , and which do not require much more space than that used to store  $\mathbf{A}$ .

Iterative algorithms solve linear equations *while only performing multiplications* by  $\mathbf{A}$ , and performing a few vector operations. Unlike the *direct methods* which are based on elimination, the iterative algorithms do not find exact solutions. Rather, they get closer and closer to the solution the longer they work. The advantage of these methods is that they need to store very little, and are often much faster than the direct methods. When  $\mathbf{A}$  is symmetric, the running times of these methods are determined by the eigenvalues of  $\mathbf{A}$ .

Throughout this lecture we will assume that  $\mathbf{A}$  is positive definite or positive semidefinite.

## 23.3 First-Order Richardson Iteration

To get started, we will examine a simple, but sub-optimal, iterative method, Richardson's iteration. The idea of the method is to find an iterative process that has the solution to  $\mathbf{A}\mathbf{x} = \mathbf{b}$  as a fixed

point, and which converges. We observe that if  $\mathbf{A}\mathbf{x} = \mathbf{b}$ , then for any  $\alpha$ ,

$$\begin{aligned}\alpha\mathbf{A}\mathbf{x} &= \alpha\mathbf{b}, \quad \implies \\ \mathbf{x} + (\alpha\mathbf{A} - I)\mathbf{x} &= \alpha\mathbf{b}, \quad \implies \\ \mathbf{x} &= (I - \alpha\mathbf{A})\mathbf{x} + \alpha\mathbf{b}.\end{aligned}$$

This leads us to the following iterative process:

$$\mathbf{x}^t = (I - \alpha\mathbf{A})\mathbf{x}^{t-1} + \alpha\mathbf{b}, \quad (23.1)$$

where we will take  $\mathbf{x}^0 = \mathbf{0}$ . We will show that this converges if

$$I - \alpha\mathbf{A}$$

has norm less than 1, and that the convergence rate depends on how much the norm is less than 1. This is analogous to our analysis of random walks on graphs.

As we are assuming  $\mathbf{A}$  is symmetric,  $I - \alpha\mathbf{A}$  is symmetric as well, and so its norm is the maximum absolute value of its eigenvalues. Let  $0 < \lambda_1 \leq \lambda_2 \dots \leq \lambda_n$  be the eigenvalues of  $\mathbf{A}$ . Then, the eigenvalues of  $I - \alpha\mathbf{A}$  are

$$1 - \alpha\lambda_i,$$

and the norm of  $I - \alpha\mathbf{A}$  is

$$\max_i |1 - \alpha\lambda_i| = |\max(1 - \alpha\lambda_1, 1 - \alpha\lambda_n)|.$$

This is minimized by taking

$$\alpha = \frac{2}{\lambda_n + \lambda_1},$$

in which case the smallest and largest eigenvalues of  $I - \alpha\mathbf{A}$  become

$$\pm \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1},$$

and the norm of  $I - \alpha\mathbf{A}$  becomes

$$1 - \frac{2\lambda_1}{\lambda_n + \lambda_1}.$$

While we might not know  $\lambda_n + \lambda_1$ , a good guess is often sufficient. If we choose an  $\alpha < 2/(\lambda_n + \lambda_1)$ , then the norm of  $I - \alpha\mathbf{A}$  is at most

$$1 - \alpha\lambda_1.$$

To show that  $\mathbf{x}^t$  converges to the solution,  $\mathbf{x}$ , consider  $\mathbf{x} - \mathbf{x}^t$ . We have

$$\begin{aligned}\mathbf{x} - \mathbf{x}^t &= ((I - \alpha\mathbf{A})\mathbf{x} + \alpha\mathbf{b}) - ((I - \alpha\mathbf{A})\mathbf{x}^{t-1} + \alpha\mathbf{b}) \\ &= (I - \alpha\mathbf{A})(\mathbf{x} - \mathbf{x}^{t-1}).\end{aligned}$$

So,

$$\mathbf{x} - \mathbf{x}^t = (I - \alpha\mathbf{A})^t(\mathbf{x} - \mathbf{x}^0) = (I - \alpha\mathbf{A})^t\mathbf{x}.$$

and

$$\begin{aligned}\|\mathbf{x} - \mathbf{x}^t\| &= \|(I - \alpha \mathbf{A})^t \mathbf{x}\| \leq \|(I - \alpha \mathbf{A})^t\| \|\mathbf{x}\| \\ &= \|(I - \alpha \mathbf{A})\|^t \|\mathbf{x}\| \\ &\leq \left(1 - \frac{2\lambda_1}{\lambda_n + \lambda_1}\right)^t \|\mathbf{x}\|. \\ &\leq e^{-2\lambda_1 t / (\lambda_n + \lambda_1)} \|\mathbf{x}\|.\end{aligned}$$

So, if we want to get a solution  $\mathbf{x}^t$  with

$$\frac{\|\mathbf{x} - \mathbf{x}^t\|}{\|\mathbf{x}\|} \leq \epsilon,$$

it suffices to run for

$$\frac{\lambda_n + \lambda_1}{2\lambda_1} \ln(1/\epsilon) = \left(\frac{\lambda_n}{2\lambda_1} + \frac{1}{2}\right) \ln(1/\epsilon).$$

iterations. The term

$$\frac{\lambda_n}{\lambda_1}$$

is called the *condition number*<sup>1</sup> of the matrix  $\mathbf{A}$ , when  $\mathbf{A}$  is symmetric. It is often written  $\kappa(\mathbf{A})$ , and the running time of iterative algorithms is often stated in terms of this quantity. We see that if the condition number is small, then this algorithm quickly provides an approximate solution.

## 23.4 A polynomial approximation of the inverse

I am now going to give another interpretation of Richardson's iteration. It provides us with a polynomial in  $\mathbf{A}$  that approximates  $\mathbf{A}^{-1}$ . In particular, the  $t$ th iterate,  $\mathbf{x}^t$  can be expressed in the form

$$p^t(\mathbf{A})\mathbf{b},$$

where  $p^t$  is a polynomial of degree  $t$ .

We will view  $p^t(\mathbf{A})$  as a good approximation of  $\mathbf{A}^{-1}$  if

$$\|\mathbf{A}p^t(\mathbf{A}) - I\|$$

is small. From the formula defining Richardson's iteration (23.1), we find

$$\begin{aligned}\mathbf{x}^0 &= \mathbf{0}, \\ \mathbf{x}^1 &= \alpha \mathbf{b}, \\ \mathbf{x}^2 &= (\mathbf{I} - \alpha \mathbf{A})\alpha \mathbf{b} + \alpha \mathbf{b}, \\ \mathbf{x}^3 &= (\mathbf{I} - \alpha \mathbf{A})^2 \alpha \mathbf{b} + (\mathbf{I} - \alpha \mathbf{A})\alpha \mathbf{b} + \alpha \mathbf{b}, \text{ and} \\ \mathbf{x}^t &= \sum_{i=0}^t (\mathbf{I} - \alpha \mathbf{A})^i \alpha \mathbf{b}.\end{aligned}$$

---

<sup>1</sup>For general matrices, the condition number is defined to be the ratio of the largest to smallest singular value.

To get some idea of why this should be an approximation of  $\mathbf{A}^{-1}$ , consider what we get if we let the sum go to infinity. Assuming that the infinite sum converges, we have

$$\alpha \sum_{i=0}^{\infty} (\mathbf{I} - \alpha \mathbf{A})^i = \alpha (\mathbf{I} - (\mathbf{I} - \alpha \mathbf{A}))^{-1} = \alpha(\alpha \mathbf{A})^{-1} = \mathbf{A}^{-1}.$$

So, the Richardson iteration can be viewed as a truncation of this infinite summation.

In general, a polynomial  $p^t$  will enable us to compute a solution to precision  $\epsilon$  if

$$\|p^t(\mathbf{A})\mathbf{b} - \mathbf{x}\| \leq \epsilon \|\mathbf{x}\|.$$

As  $\mathbf{b} = \mathbf{A}\mathbf{x}$ , this is equivalent to

$$\|p^t(\mathbf{A})\mathbf{A}\mathbf{x} - \mathbf{x}\| \leq \epsilon \|\mathbf{x}\|,$$

which is equivalent to

$$\|\mathbf{A}p^t(\mathbf{A}) - \mathbf{I}\| \leq \epsilon$$

## 23.5 Better Polynomials

This leads us to the question of whether we can find better polynomial approximations to  $\mathbf{A}^{-1}$ . The reason I ask is that the answer is yes! As  $\mathbf{A}$ ,  $p^t(\mathbf{A})$  and  $\mathbf{I}$  all commute, the matrix

$$\mathbf{A}p^t(\mathbf{A}) - \mathbf{I}$$

is symmetric and its norm is the maximum absolute value of its eigenvalues. So, it suffices to find a polynomial  $p^t$  such that

$$|\lambda_i p^t(\lambda_i) - 1| \leq \epsilon,$$

for all eigenvalues  $\lambda_i$  of  $\mathbf{A}$ .

To reformulate this, define

$$q^t(x) = 1 - xp(x).$$

Then, it suffices to find a polynomial  $q^t$  of degree  $t + 1$  for which

$$\begin{aligned} q^t(0) &= 1, \text{ and} \\ |q^t(x)| &\leq \epsilon, \text{ for } \lambda_1 \leq x \leq \lambda_n. \end{aligned}$$

We will see that there are polynomials of degree

$$\ln(2/\epsilon) \left( \sqrt{\lambda_n/\lambda_1} + 1 \right) / 2$$

that satisfy these conditions and thus allow us to compute solutions of accuracy  $\epsilon$ . In terms of the condition number of  $\mathbf{A}$ , this is a quadratic improvement over Richardson's first-order method.

**Theorem 23.5.1.** *For every  $t \geq 1$ , and  $0 < \lambda_{\min} \leq \lambda_{\max}$ , there exists a polynomial  $q^t(x)$  such that*

1.  $|q^t(x)| \leq \epsilon$ , for  $\lambda_{\min} \leq x \leq \lambda_{\max}$ , and
2.  $q^t(0) = 1$ ,

for

$$\epsilon \leq 2(1 + 2/\sqrt{\kappa})^{-t} \leq 2e^{-2t/\sqrt{\kappa}},$$

where

$$\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}.$$

## 23.6 Chebyshev Polynomials

I'd now like to explain how we find these better polynomials. The key is to transform one of the most fundamental polynomials: the Chebyshev polynomials. These polynomials are as small as possible on  $[-1, 1]$ , and grow quickly outside this interval. We will translate the interval  $[-1, 1]$  to obtain the polynomials we need.

The  $t$ th Chebyshev polynomial,  $T_t(x)$  has degree  $t$ , and may be defined by setting

$$T_0(x) = 1, \quad T_1(x) = 2x - 1,$$

and for  $t \geq 2$

$$T_t(x) = 2xT_{t-1}(x) - T_{t-2}(x).$$

These polynomials are best understood by realizing that they are the polynomials for which

$$\cos(t\theta) = T_t(\cos(\theta)) \quad \text{and} \quad \cosh(t\theta) = T_t(\cosh(\theta)).$$

It might not be obvious that one can express  $\cos(t\theta)$  as a polynomial in  $\cos(\theta)$ . To see this, and the correctness of the above formulas, recall that

$$\cos(\theta) = \frac{1}{2} (e^{i\theta} + e^{-i\theta}), \quad \text{and} \quad \cosh(\theta) = \frac{1}{2} (e^\theta + e^{-\theta}).$$

To verify that these satisfy the stated recurrences, compute

$$\begin{aligned} & \frac{1}{2} (e^\theta + e^{-\theta}) (e^{(t-1)\theta} + e^{-(t-1)\theta}) - \frac{1}{2} (e^{(t-2)\theta} + e^{-(t-2)\theta}) \\ &= \frac{1}{2} (e^{(t\theta)} + e^{-t\theta}) + \frac{1}{2} (e^{(t-2)\theta} + e^{-(t-2)\theta}) - \frac{1}{2} (e^{(t-2)\theta} + e^{-(t-2)\theta}) \\ &= \frac{1}{2} (e^{(t\theta)} + e^{-t\theta}). \end{aligned}$$

**Claim 23.6.1.** *For  $x \in [-1, 1]$ ,  $|T_t(x)| \leq 1$ .*

*Proof.* For  $x \in [-1, 1]$ , there is a  $\theta$  so that  $\cos(\theta) = x$ . We then have  $T_t(x) = \cos(t\theta)$ , which must also be between  $-1$  and  $1$ .  $\square$

To compute the values of the Chebyshev polynomials outside  $[-1, 1]$ , we use the hyperbolic cosine function. Hyperbolic cosine maps the real line to  $[1, \infty]$  and is symmetric about the origin. So, the inverse of hyperbolic cosine may be viewed as a map from  $[1, \infty]$  to  $[0, \infty]$ , and satisfies

$$\operatorname{acosh}(x) = \ln \left( x + \sqrt{x^2 - 1} \right), \text{ for } x \geq 1.$$

**Claim 23.6.2.** For  $\gamma > 0$ ,

$$T_t(1 + \gamma) \geq (1 + \sqrt{2\gamma})^t / 2.$$

*Proof.* Setting  $x = 1 + \gamma$ , we compute

$$\begin{aligned} T_t(x) &= \frac{1}{2} \left( e^{t \operatorname{acosh}(x)} + e^{-t \operatorname{acosh}(x)} \right) \\ &\geq \frac{1}{2} \left( e^{t \operatorname{acosh}(x)} \right) \\ &= \frac{1}{2} (x + \sqrt{x^2 - 1})^t \\ &= \frac{1}{2} (1 + \gamma + \sqrt{(1 + \gamma)^2 - 1})^t \\ &= \frac{1}{2} (1 + \gamma + \sqrt{2\gamma + \gamma^2})^t \\ &\geq \frac{1}{2} (1 + \sqrt{2\gamma})^t. \end{aligned}$$

$\square$

## 23.7 Proof of Theorem 23.5.1

We will exploit the following properties of the Chebyshev polynomials:

1.  $T_t$  has degree  $t$ .
2.  $T_t(x) \in [-1, 1]$ , for  $x \in [-1, 1]$ .
3.  $T_t(x)$  is monotonically increasing for  $x \geq 1$ .
4.  $T_t(1 + \gamma) \geq (1 + \sqrt{2\gamma})^t / 2$ , for  $\gamma > 0$ .

To express  $q^t(x)$  in terms of a Chebyshev polynomial, we should map the range on which we want  $p$  to be small,  $[\lambda_{\min}, \lambda_{\max}]$  to  $[-1, 1]$ . We will accomplish this with the linear map:

$$l(x) \stackrel{\text{def}}{=} \frac{\lambda_{\max} + \lambda_{\min} - 2x}{\lambda_{\max} - \lambda_{\min}}.$$

Note that

$$l(x) = \begin{cases} -1 & \text{if } x = \lambda_{\max} \\ 1 & \text{if } x = \lambda_{\min} \\ \frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}} & \text{if } x = 0. \end{cases}$$

To guarantee that the constant coefficient in  $q^t(x)$  is one ( $q^t(0) = 1$ ), we should set

$$q^t(x) \stackrel{\text{def}}{=} \frac{T_t(l(x))}{T_t(l(0))}.$$

We know that  $|T_t(l(x))| \leq 1$  for  $x \in [\lambda_{\min}, \lambda_{\max}]$ . To find  $q(x)$  for  $x$  in this range, we must compute  $T_t(l(0))$ . We have

$$l(0) \geq 1 + 2/\kappa(\mathbf{A}),$$

and so by properties 3 and 4 of Chebyshev polynomials,

$$T_t(l(0)) \geq (1 + 2/\sqrt{\kappa})^t / 2.$$

Thus,

$$q(x) \leq 2(1 + 2/\sqrt{\kappa})^{-t},$$

for  $x \in [\lambda_{\min}, \lambda_{\max}]$ , and so all eigenvalues of  $q(\mathbf{A})$  will have absolute value at most  $2(1 + 2/\sqrt{\kappa})^{-t}$ .

## 23.8 Laplacian Systems

One might at first think that these techniques do not apply to Laplacian systems, as these are always singular. However, we can apply these techniques without change if  $\mathbf{b}$  is in the span of  $\mathbf{L}$ . That is, if  $\mathbf{b}$  is orthogonal to the all-1s vector and the graph is connected. In this case the eigenvalue  $\lambda_1 = 0$  has no role in the analysis, and it is replaced by  $\lambda_2$ . One way of understanding this is to just view  $\mathbf{L}$  as an operator acting on the space orthogonal to the all-1s vector.

By considering the example of the Laplacian of the path graph, one can show that it is impossible to do much better than the  $\sqrt{\kappa}$  iteration bound that I claimed at the end of the last section. To see this, first observe that when one multiplies a vector  $\mathbf{x}$  by  $\mathbf{L}$ , the entry  $(\mathbf{L}\mathbf{x})(i)$  just depends on  $\mathbf{x}(i-1), \mathbf{x}(i)$ , and  $\mathbf{x}(i+1)$ . So, if we apply a polynomial of degree at most  $t$ ,  $\mathbf{x}^t(i)$  will only depend on  $\mathbf{b}(j)$  with  $i-t \leq j \leq i+t$ . This tells us that we will need a polynomial of degree on the order of  $n$  to solve such a system.

On the other hand,  $\sqrt{\lambda_n/\lambda_2}$  is on the order of  $n$  as well. So, we should not be able to solve the system with a polynomial whose degree is significantly less than  $\sqrt{\lambda_n/\lambda_2}$ .

## 23.9 Warning

The polynomial-based approach that I have described here only works in infinite precision arithmetic. In finite precision arithmetic one has to be more careful about how one implements these

algorithms. This is why the descriptions of methods such as the Chebyshev method found in Numerical Linear Algebra textbooks are more complicated than that presented here. The algorithms that are actually used are mathematically identical in infinite precision, but they actually work. The problem with the naive implementations are the typical experience: in double-precision arithmetic the polynomial approach to Chebyshev will fail to solve linear systems in random positive definite matrices in 60 dimensions!

## References

## Preconditioning Laplacians

*Daniel A. Spielman*

November 28, 2018

A *preconditioner* for a positive semidefinite matrix  $\mathbf{A}$  is a positive semidefinite matrix  $\mathbf{B}$  such that it is easy to solve systems of linear equations in  $\mathbf{B}$  and the condition number of  $\mathbf{B}^{-1}\mathbf{A}$  is small. A good preconditioner allows one to quickly solve systems of equations in  $\mathbf{A}$ .

In this lecture, we will measure the quality of preconditioners in terms of the ratio

$$\kappa(\mathbf{A}, \mathbf{B}) \stackrel{\text{def}}{=} \beta/\alpha,$$

where  $\alpha$  is the largest number and  $\beta$  is the smallest such that

$$\alpha\mathbf{B} \preccurlyeq \mathbf{A} \preccurlyeq \beta\mathbf{B}.$$

**Lemma 24.0.1.** *Let  $\alpha$  and  $\beta$  be as defined above. Then,  $\alpha$  and  $\beta$  are the smallest and largest eigenvalues of  $\mathbf{B}^{-1}\mathbf{A}$ , excluding possible zero eigenvalues corresponding to a common nullspace of  $\mathbf{A}$  and  $\mathbf{B}$ .*

We need to exclude the common nullspace when  $\mathbf{A}$  and  $\mathbf{B}$  are the Laplacian matrices of connected graphs. If these matrices have different nullspaces  $\alpha = 0$  or  $\beta = \infty$  and the condition number  $\beta/\alpha$  is infinite.

*Proof of Lemma 24.0.1.* We just prove the statement for  $\beta$ , in the case where neither matrix is singular. We have

$$\begin{aligned} \lambda_{\max}(\mathbf{B}^{-1}\mathbf{A}) &= \lambda_{\max}(\mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2}) \\ &= \max_{\mathbf{x}} \frac{\mathbf{x}^T \mathbf{B}^{-1/2} \mathbf{A} \mathbf{B}^{-1/2} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \\ &= \max_{\mathbf{y}} \frac{\mathbf{y}^T \mathbf{A} \mathbf{y}}{\mathbf{y}^T \mathbf{B} \mathbf{y}}, \quad \text{setting } \mathbf{y} = \mathbf{B}^{-1/2} \mathbf{x}, \end{aligned}$$

which equals  $\beta$ . □

Recall that the eigenvalues of  $\mathbf{B}^{-1}\mathbf{A}$  are the same as those of  $\mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2}$  and  $\mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2}$ .

## 24.1 Approximate Solutions

Recall the  $\mathbf{A}$ -norm:

$$\|\mathbf{x}\|_{\mathbf{A}} = \sqrt{\mathbf{x}^T \mathbf{A} \mathbf{x}} = \left\| \mathbf{A}^{1/2} \mathbf{x} \right\|.$$

We say that  $\tilde{\mathbf{x}}$  is an  $\epsilon$ -approximate solution to the problem  $\mathbf{A}\mathbf{x} = \mathbf{b}$  if

$$\|\tilde{\mathbf{x}} - \mathbf{x}\|_{\mathbf{A}} \leq \epsilon \|\mathbf{x}\|_{\mathbf{A}}.$$

## 24.2 Iterative Refinement

We will now see how to use a *very* good preconditioner to solve a system of equations. Let's consider a preconditioner  $\mathbf{B}$  that satisfies

$$(1 - \epsilon)\mathbf{B} \preccurlyeq \mathbf{A} \preccurlyeq (1 + \epsilon)\mathbf{B}.$$

So, all of the eigenvalues of

$$\mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2} - \mathbf{I}$$

have absolute value at most  $\epsilon$ .

The vector  $\mathbf{B}^{-1}\mathbf{b}$  is a good approximation of  $\mathbf{x}$  in the  $\mathbf{A}$ -norm. We have

$$\begin{aligned} \|\mathbf{B}^{-1}\mathbf{b} - \mathbf{x}\|_{\mathbf{A}} &= \left\| \mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{b} - \mathbf{A}^{1/2}\mathbf{x} \right\| \\ &= \left\| \mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}\mathbf{x} - \mathbf{A}^{1/2}\mathbf{x} \right\| \\ &= \left\| \mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2}(\mathbf{A}^{1/2}\mathbf{x}) - \mathbf{A}^{1/2}\mathbf{x} \right\| \\ &\leq \left\| \mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2} - \mathbf{I} \right\| \left\| \mathbf{A}^{1/2}\mathbf{x} \right\| \\ &\leq \epsilon \left\| \mathbf{A}^{1/2}\mathbf{x} \right\| \\ &= \epsilon \left\| \mathbf{x} \right\|_{\mathbf{A}}. \end{aligned}$$

**Remark:** This result crucially depends upon the use of the  $\mathbf{A}$ -norm. It fails under the Euclidean norm.

If we want a better solution, we can just compute the residual and solve the problem in the residual. That is, we set

$$\mathbf{x}_1 = \mathbf{B}^{-1}\mathbf{b},$$

and compute

$$\mathbf{r}_1 = \mathbf{b} - \mathbf{A}\mathbf{x}_1 = \mathbf{A}(\mathbf{x} - \mathbf{x}_1).$$

We then use one solve in  $\mathbf{B}$  to compute a vector  $\mathbf{x}_2$  such that

$$\|(\mathbf{x} - \mathbf{x}_1) - \mathbf{x}_2\|_{\mathbf{A}} \leq \epsilon \|\mathbf{x} - \mathbf{x}_1\|_{\mathbf{A}} \leq \epsilon^2 \|\mathbf{x}\|_{\mathbf{A}}.$$

So,  $\mathbf{x}_1 + \mathbf{x}_2$ , our new estimate of  $\mathbf{x}$ , differs from  $\mathbf{x}$  by at most an  $\epsilon^2$  factor. Continuing in this way, we can find an  $\epsilon^k$  approximation of  $\mathbf{x}$  after solving  $k$  linear systems in  $\mathbf{B}$ . This procedure is called *iterative refinement*.

## 24.3 Iterative Methods in the Matrix Norm

The iterative methods we studied last class can also be shown to produce good approximate solutions in the matrix norm. Given a matrix  $\mathbf{A}$ , these produce  $\epsilon$ -approximation solutions after  $t$

iterations if there is a polynomial  $q$  of degree  $t$  for which  $q(0) = 1$  and  $|q(\lambda_i)| \leq \epsilon$  for all eigenvalues of  $\mathbf{A}$ . To see this, recall that we can define  $p(x)$  so that  $q(x) = 1 - xp(x)$ , and set

$$\tilde{\mathbf{x}} = p(\mathbf{A})\mathbf{b},$$

to get

$$\|\tilde{\mathbf{x}} - \mathbf{x}\|_{\mathbf{A}} = \|p(\mathbf{A})\mathbf{b} - \mathbf{x}\|_{\mathbf{A}} = \|p(\mathbf{A})\mathbf{A}\mathbf{x} - \mathbf{x}\|_{\mathbf{A}}.$$

As  $\mathbf{I}$ ,  $\mathbf{A}$ ,  $p(\mathbf{A})$  and  $\mathbf{A}^{1/2}$  all commute, this equals

$$\begin{aligned} \|\mathbf{A}^{1/2}p(\mathbf{A})\mathbf{A}\mathbf{x} - \mathbf{A}^{1/2}\mathbf{x}\| &= \|p(\mathbf{A})\mathbf{A}\mathbf{A}^{1/2}\mathbf{x} - \mathbf{A}^{1/2}\mathbf{x}\| \\ &\leq \|p(\mathbf{A})\mathbf{A} - \mathbf{I}\| \|\mathbf{A}^{1/2}\mathbf{x}\| \\ &\leq \epsilon \|\mathbf{x}\|_{\mathbf{A}}. \end{aligned}$$

## 24.4 Preconditioned Iterative Methods

Preconditioned iterative methods can be viewed as the extension of Iterative Refinement by algorithms like Chebyshev iteration and the Preconditioned Conjugate Gradient. These usually work with condition numbers much larger than 2.

In each iteration of a preconditioned method we will solve a system of equations in  $\mathbf{B}$ , multiply a vector by  $\mathbf{A}$ , and perform a constant number of other vector operations. For this to be worthwhile, the cost of solving equations in  $\mathbf{B}$  has to be low.

We begin by seeing how the analysis with polynomials translates. Let  $\lambda_i$  be the  $i$ th eigenvalue of  $\mathbf{B}^{-1}\mathbf{A}$ . If  $q_t(x) = 1 - xp_t(x)$  is a polynomial such that  $|q_t(\lambda_i)| \leq \epsilon$  for all  $i$ , then

$$\mathbf{x}_t \stackrel{\text{def}}{=} p_t(\mathbf{B}^{-1}\mathbf{A})\mathbf{B}^{-1}\mathbf{b}$$

will be an  $\epsilon$ -approximate solution to  $\mathbf{A}\mathbf{x} = \mathbf{b}$ :

$$\begin{aligned} \|\mathbf{x} - \mathbf{x}_t\|_{\mathbf{A}} &= \|\mathbf{A}^{1/2}\mathbf{x} - \mathbf{A}^{1/2}\mathbf{x}_t\| \\ &= \|\mathbf{A}^{1/2}\mathbf{x} - \mathbf{A}^{1/2}p_t(\mathbf{B}^{-1}\mathbf{A})\mathbf{B}^{-1}\mathbf{b}\| \\ &= \|\mathbf{A}^{1/2}\mathbf{x} - \mathbf{A}^{1/2}p_t(\mathbf{B}^{-1}\mathbf{A})\mathbf{B}^{-1}\mathbf{A}\mathbf{x}\| \\ &= \|\mathbf{A}^{1/2}\mathbf{x} - \mathbf{A}^{1/2}p_t(\mathbf{B}^{-1}\mathbf{A})\mathbf{B}^{-1}\mathbf{A}^{1/2}(\mathbf{A}^{1/2}\mathbf{x})\| \\ &\leq \|\mathbf{I} - \mathbf{A}^{1/2}p_t(\mathbf{B}^{-1}\mathbf{A})\mathbf{B}^{-1}\mathbf{A}^{1/2}\| \|\mathbf{A}^{1/2}\mathbf{x}\|. \end{aligned}$$

We now prod this matrix into a more useful form:

$$\mathbf{I} - \mathbf{A}^{1/2}p_t(\mathbf{B}^{-1}\mathbf{A})\mathbf{B}^{-1}\mathbf{A}^{1/2} = \mathbf{I} - p_t(\mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2})\mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2} = q_t(\mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2}).$$

So, we find

$$\|\mathbf{x} - \mathbf{x}_t\|_{\mathbf{A}} \leq \|q_t(\mathbf{A}^{1/2}\mathbf{B}^{-1}\mathbf{A}^{1/2})\| \|\mathbf{A}^{1/2}\mathbf{x}\| \leq \epsilon \|\mathbf{x}\|_{\mathbf{A}}.$$

The Preconditioned Conjugate Gradient (PCG) is a magical algorithm that after  $t$  steps (each of which involves solving a system in  $\mathbf{B}$ , multiplying a vector by  $\mathbf{A}$ , and performing a constant number of vector operations) produces the vector  $\mathbf{x}_t$  that minimizes

$$\|\mathbf{x}_t - \mathbf{x}\|_{\mathbf{A}}$$

over all vectors  $\mathbf{x}_t$  that can be written in the form  $p_t(\mathbf{b})$  for a polynomial of degree at most  $t$ . That is, the algorithm finds the best possible solution among all iterative methods of the form we have described. We first bound the quality of PCG by saying that it is at least as good as Preconditioned Chebyshev, but it has the advantage of not needing to know  $\alpha$  and  $\beta$ . We will then find an improved analysis.

## 24.5 Preconditioning by Trees

Vaidya [Vai90] had the remarkable idea of preconditioning the Laplacian matrix of a graph by the Laplacian matrix of a subgraph. If  $H$  is a subgraph of  $G$ , then

$$\mathbf{L}_H \preccurlyeq \mathbf{L}_G,$$

so all eigenvalues of  $\mathbf{L}_H^{-1} \mathbf{L}_G$  are at least 1. Thus, we only need to find a subgraph  $H$  such that  $\mathbf{L}_H$  is easy to invert and such that the largest eigenvalue of  $\mathbf{L}_H^{-1} \mathbf{L}_G$  is not too big.

It is relatively easy to show that linear equations in the Laplacian matrices of trees can be solved exactly in linear time. One can either do this by finding an  $LU$ -factorization with a linear number of non-zeros, or by viewing the process of solving the linear equation as a dynamic program that passes up once from the leaves of the tree to a root, and then back down.

We will now show that a special type of tree, called a *low-stretch spanning tree* provides a very good preconditioner. To begin, let  $T$  be a spanning tree of  $G$ . Write

$$\mathbf{L}_G = \sum_{(u,v) \in E} w_{u,v} \mathbf{L}_{u,v} = \sum_{(u,v) \in E} w_{u,v} (\chi_u - \chi_v)(\chi_u - \chi_v)^T.$$

We will actually consider the trace of  $\mathbf{L}_T^{-1} \mathbf{L}_G$ . As the trace is linear, we have

$$\begin{aligned} \text{Tr}(\mathbf{L}_T^{-1} \mathbf{L}_G) &= \sum_{(u,v) \in E} w_{u,v} \text{Tr}(\mathbf{L}_T^{-1} \mathbf{L}_{u,v}) \\ &= \sum_{(u,v) \in E} w_{u,v} \text{Tr}(\mathbf{L}_T^{-1} (\chi_u - \chi_v)(\chi_u - \chi_v)^T) \\ &= \sum_{(u,v) \in E} w_{u,v} \text{Tr}((\chi_u - \chi_v)^T \mathbf{L}_T^{-1} (\chi_u - \chi_v)) \\ &= \sum_{(u,v) \in E} w_{u,v} (\chi_u - \chi_v)^T \mathbf{L}_T^{-1} (\chi_u - \chi_v). \end{aligned}$$

To evaluate this last term, we need to know the value of  $(\chi_u - \chi_v)^T \mathbf{L}_T^{-1} (\chi_u - \chi_v)$ . You already know something about it: it is the effective resistance in  $T$  between  $u$  and  $v$ . In a tree, this equals

the distance in  $T$  between  $u$  and  $v$ , when we view the length of an edge as the reciprocal of its weight. This is because it is the resistance of a path of resistors in series. Let  $T(u, v)$  denote the path in  $T$  from  $u$  to  $v$ , and let  $w_1, \dots, w_k$  denote the weights of the edges on this path. As we view the weight of an edge as the reciprocal of its length,

$$(\chi_u - \chi_v)^T \mathbf{L}_T^{-1} (\chi_u - \chi_v) = \sum_{i=1}^k \frac{1}{w_i}. \quad (24.1)$$

Even better, the term (24.1) is something that has been well-studied. It was defined by Alon, Karp, Peleg and West [AKPW95] to be the *stretch* of the unweighted edge  $(u, v)$  with respect to the tree  $T$ . Moreover, the *stretch* of the edge  $(u, v)$  with weight  $w_{u,v}$  with respect to the tree  $T$  is defined to be exactly

$$w_{u,v} \sum_{i=1}^k \frac{1}{w_i},$$

where again  $w_1, \dots, w_k$  are the weights on the edges of the unique path in  $T$  from  $u$  to  $v$ . A sequence of works, beginning with [AKPW95], has shown that every graph  $G$  has a spanning tree in which the sum of the stretches of the edges is low. The best result so far is due to [AN12], who prove the following theorem.

**Theorem 24.5.1.** *Every weighted graph  $G$  has a spanning tree subgraph  $T$  such that the sum of the stretches of all edges of  $G$  with respect to  $T$  is at most*

$$O(m \log n \log \log n),$$

where  $m$  is the number of edges  $G$ . Moreover, one can compute this tree in time  $O(m \log n \log \log n)$ .

Thus, if we choose a low-stretch spanning tree  $T$ , we will ensure that

$$\text{Tr}(\mathbf{L}_T^{-1} \mathbf{L}_G) = \sum_{(u,v) \in E} w_{u,v} (\chi_u - \chi_v)^T \mathbf{L}_T^{-1} (\chi_u - \chi_v) \leq O(m \log n \log \log n).$$

In particular, this tells us that  $\lambda_{\max}(\mathbf{L}_T^{-1} \mathbf{L}_G)$  is at most  $O(m \log n \log \log n)$ , and so the Preconditioned Conjugate Gradient will require at most  $O(m^{1/2} \log n)$  iterations, each of which requires one multiplication by  $\mathbf{L}_G$  and one linear solve in  $\mathbf{L}_T$ . This gives an algorithm that runs in time  $O(m^{3/2} \log n \log 1/\epsilon)$ , which is much lower than the  $O(n^3)$  of Gaussian elimination when  $m$ , the number of edges in  $G$ , is small.

This result is due to Boman and Hendrickson [BH01].

## 24.6 Improving the Bound on the Running Time

We can show that the Preconditioned Conjugate Gradient will actually run in closer to  $O(m^{1/3})$  iterations. Since the trace is the sum of the eigenvalues, we know that for every  $\beta > 0$ ,  $\mathbf{L}_T^{-1} \mathbf{L}_G$  has at most

$$\text{Tr}(\mathbf{L}_T^{-1} \mathbf{L}_G) / \beta$$

eigenvalues that are larger than  $\beta$ .

To exploit this fact, we use the following lemma. It basically says that we can ignore the largest eigenvalues of  $\mathbf{B}^{-1}\mathbf{A}$  if we are willing to spend one iteration for each.

**Lemma 24.6.1.** *Let  $\lambda_1, \dots, \lambda_n$  be positive numbers such that all of them are at least  $\alpha$  and at most  $k$  of them are more than  $\beta$ . Then, for every  $t \geq k$ , there exists a polynomial  $p(X)$  of degree  $t$  such that  $p(0) = 1$  and*

$$|p(\lambda_i)| \leq 2 \left(1 + \frac{2}{\sqrt{\beta/\alpha}}\right)^{-(t-k)},$$

for all  $\lambda_i$ .

*Proof.* Let  $r(X)$  be the polynomial we constructed using Chebyshev polynomials of degree  $t - k$  for which

$$|r(X)| \leq 2 \left(1 + \frac{2}{\sqrt{\beta/\alpha}}\right)^{-(t-k)},$$

for all  $X$  between  $\alpha$  and  $\beta$ . Now, set

$$p(X) = r(X) \prod_{i:\lambda_i > \beta} (1 - X/\lambda_i).$$

This new polynomial is zero at every  $\lambda_i$  greater than  $\beta$ , and for  $X$  between  $\alpha$  and  $\beta$

$$|p(X)| = |r(X)| \prod_{i:\lambda_i > \beta} |(1 - X/\lambda_i)| \leq |r(X)|,$$

as we always have  $X < \lambda_i$  in the product.  $\square$

Applying this lemma to the analysis of the Preconditioned Conjugate Gradient, with  $\beta = \text{Tr}(\mathbf{L}_T^{-1}\mathbf{L}_G)^{2/3}$  and  $k = \text{Tr}(\mathbf{L}_T^{-1}\mathbf{L}_G)^{1/3}$ , we find that the algorithm produces  $\epsilon$ -approximate solutions within

$$O(\text{Tr}(\mathbf{L}_T^{-1}\mathbf{L}_G)^{1/3} \ln(1/\epsilon)) = O(m^{1/3} \log n \ln 1/\epsilon)$$

iterations.

This result is due to Spielman and Woo [SW09].

## 24.7 Further Improvements

We now have three families of algorithms for solving systems of equations in Laplacian matrices in nearly-linear time.

- By subgraph preconditioners. These basically work by adding back edges to the low-stretch trees. The resulting systems can no longer be solved directly in linear time. Instead, we use Gaussian elimination to eliminate the degree 1 and 2 vertices to reduce to a smaller system, and then solve that system recursively. The first nearly linear time algorithm of this form ran in time  $O(m \log^c n \log 1/\epsilon)$ , for some constant  $c$  [ST09]. An approach of this form was first made practical (and much simpler) by Koutis, Miller, and Peng [KMP11]. The asymptotically fastest method also works this way. It runs in time  $O(m \log^{1/2} m \log^c \log n \log 1/\epsilon)$ , [CKM<sup>+</sup>14] (Cohen, Kyng, Miller, Pachocki, Peng, Rao, Xu).
- By sparsification (see my notes from Lecture 19 from 2015). These algorithms work rather differently, and do not exploit low-stretch spanning trees. They appear in the papers [PS14, KLP<sup>+</sup>16].
- Accelerating Gaussian elimination by random sampling, by Kyng and Sachdeva [KS16]. This is the most elegant of the algorithms. While the running time of the algorithms,  $O(m \log^2 n \log 1/\epsilon)$  is not the asymptotically best, the algorithm is so simple that it is the best in practice. An optimized implementation appears in the package `Laplacian.jl`.

There are other algorithms that are often fast in practice, but for which we have no theoretical analysis. I suggest the Algebraic Multigrid of Livne and Brandt, and the Combinatorial Multigrid of Yiannis Koutis.

## 24.8 Questions

I conjecture that it is possible to construct spanning trees of even lower stretch. Does every graph have a spanning tree of average stretch  $2 \log_2 n$ ? I do not see any reason this should not be true. I also believe that this should be achievable by a practical algorithm. The best code that I know for computing low-stretch spanning trees, and which I implemented in `Laplacians.jl`, is a heuristic based on the algorithm of Alon, Karp, Peleg and West. However, I do not know an analysis of their algorithm that gives stretch better than  $O(m 2^{\sqrt{\log n}})$ . The theoretically better low-stretch trees of Abraham and Neiman are obtained by improving constructions of [EEST08, ABN08]. However, they seem too complicated to be practical.

The eigenvalues of  $\mathbf{L}_H^{-1} \mathbf{L}_G$  are called generalized eigenvalues. The relation between generalized eigenvalues and stretch is the first result of which I am aware that establishes a combinatorial interpretation of generalized eigenvalues. Can you find any others?

## References

- [ABN08] I. Abraham, Y. Bartal, and O. Neiman. Nearly tight low stretch spanning trees. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 781–790, Oct. 2008.

- [AKPW95] Noga Alon, Richard M. Karp, David Peleg, and Douglas West. A graph-theoretic game and its application to the  $k$ -server problem. *SIAM Journal on Computing*, 24(1):78–100, February 1995.
- [AN12] Ittai Abraham and Ofer Neiman. Using petal-decompositions to build a low stretch spanning tree. In *Proceedings of the 44th Annual ACM Symposium on the Theory of Computing (STOC '12)*, pages 395–406, 2012.
- [BH01] Erik Boman and B. Hendrickson. On spanning tree preconditioners. Manuscript, Sandia National Lab., 2001.
- [CKM<sup>+</sup>14] Michael B. Cohen, Rasmus Kyng, Gary L. Miller, Jakub W. Pachocki, Richard Peng, Anup B. Rao, and Shen Chen Xu. Solving sdd linear systems in nearly  $m\log 1/2n$  time. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 343–352, New York, NY, USA, 2014. ACM.
- [EEST08] Michael Elkin, Yuval Emek, Daniel A. Spielman, and Shang-Hua Teng. Lower-stretch spanning trees. *SIAM Journal on Computing*, 32(2):608–628, 2008.
- [KLP<sup>+</sup>16] Rasmus Kyng, Yin Tat Lee, Richard Peng, Sushant Sachdeva, and Daniel A. Spielman. Sparsified cholesky and multigrid solvers for connection laplacians. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 842–850. ACM, 2016.
- [KMP11] I. Koutis, G.L. Miller, and R. Peng. A nearly- $m\log n$  time solver for sdd linear systems. In *Foundations of Computer Science (FOCS), 2011 52nd Annual IEEE Symposium on*, pages 590–598, 2011.
- [KS16] Rasmus Kyng and Sushant Sachdeva. Approximate gaussian elimination for laplacians-fast, sparse, and simple. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 573–582. IEEE, 2016.
- [PS14] Richard Peng and Daniel A. Spielman. An efficient parallel solver for SDD linear systems. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 333–342, 2014.
- [ST09] Daniel A. Spielman and Shang-Hua Teng. Nearly-linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. *CoRR*, abs/cs/0607105, 2009. Available at <http://www.arxiv.org/abs/cs.NA/0607105>.
- [SW09] Daniel A. Spielman and Jaeho Woo. A note on preconditioning by low-stretch spanning trees. *CoRR*, abs/0903.2816, 2009. Available at <http://arxiv.org/abs/0903.2816>.
- [Vai90] Pravin M. Vaidya. Solving linear equations with symmetric diagonally dominant matrices by constructing good preconditioners. Unpublished manuscript UIUC 1990. A talk based on the manuscript was presented at the IMA Workshop on Graph Theory and Sparse Matrix Computation, October 1991, Minneapolis., 1990.

## Bipartite Ramanujan Graphs

*Daniel A. Spielman*

December 3, 2018

## 25.1 Overview

Margulis [Mar88] and Lubotzky, Phillips and Sarnak [LPS88] presented the first explicit constructions of infinite families of Ramanujan graphs. These had degrees  $p + 1$ , for primes  $p$ . There have been a few other explicit constructions, [Piz90, Chi92, JL97, Mor94], all of which produce graphs of degree  $q + 1$  for some prime power  $q$ . Over this lecture and the next we will prove the existence of infinite families of bipartite Ramanujan of every degree. While today's proof of existence does not lend itself to an explicit construction, it is easier to understand than the presently known explicit constructions.

We think that much stronger results should be true. There is good reason to think that random  $d$ -regular graphs should be Ramanujan [MNS08]. And, Friedman [Fri08] showed that a random  $d$ -regular graph is almost Ramanujan: for sufficiently large  $n$  such a graph is a  $2\sqrt{d-1} + \epsilon$  approximation of the complete graph with high probability, for every  $\epsilon > 0$ .

In today's lecture, we will use the method of interlacing families of polynomials to prove (half) a conjecture of Bilu and Linial [BL06] that every bipartite Ramanujan graph has a 2-lift that is also Ramanujan. This theorem comes from [MSS15a], but today's proof is informed by the techniques of [HPS15]. We will use theorems about the matching polynomials of graphs that we will prove next lecture.

In the same way that a Ramanujan graph approximates the complete graph, a bipartite Ramanujan graph approximates a complete bipartite graph. We say that a  $d$ -regular graph is a bipartite Ramanujan graph if all of its adjacency matrix eigenvalues, other than  $d$  and  $-d$ , have absolute value at most  $2\sqrt{d-1}$ . The eigenvalue of  $d$  is a consequence of being  $d$ -regular and the eigenvalue of  $-d$  is a consequence of being bipartite. In particular, recall that the adjacency matrix eigenvalues of a bipartite graph are symmetric about the origin. This is a special case of the following claim, which you can prove when you have a sparse moment.

**Claim 25.1.1.** *The eigenvalues of a symmetric matrix of the form*

$$\begin{pmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^T & \mathbf{0} \end{pmatrix}$$

*are symmetric about the origin.*

We remark that one can derive bipartite Ramanujan graphs from ordinary Ramanujan graphs—just take the double cover. However, we do not know any way to derive ordinary Ramanujan graphs from the bipartite ones.

As opposed to reasoning directly about eigenvalues, we will work with characteristic polynomials. For a matrix  $\mathbf{M}$ , we write its characteristic polynomial in the variable  $x$  as

$$\chi_x(\mathbf{M}) \stackrel{\text{def}}{=} \det(x\mathbf{I} - \mathbf{M}).$$

## 25.2 2-Lifts

We saw 2-lifts of graphs in Problem 3 from Problem Set 2:

We define a *signed adjacency matrix* of  $G$  to be a symmetric matrix  $\mathbf{S}$  with the same nonzero pattern as the adjacency matrix  $\mathbf{A}$ , but such that each nonzero entry is either 1 or  $-1$ .

We will use it to define a graph  $G^{\mathbf{S}}$ . Like the double-cover, the graph  $G^{\mathbf{S}}$  will have two vertices for every vertex of  $G$  and two edges for every edge of  $G$ . For each edge  $(u, v) \in E$ , if  $\mathbf{S}(u, v) = -1$  then  $G^{\mathbf{S}}$  has the two edges

$$(u_1, v_2) \quad \text{and} \quad (v_1, u_2),$$

just like the double-cover. If  $\mathbf{S}(u, v) = 1$ , then  $G^{\mathbf{S}}$  has the two edges

$$(u_1, v_1) \quad \text{and} \quad (v_2, u_2).$$

You should check that  $G^{-\mathbf{A}}$  is the double-cover of  $G$  and that  $G^{\mathbf{A}}$  consists of two disjoint copies of  $G$ .

Prove that the eigenvalues of the adjacency matrix of  $G^{\mathbf{S}}$  are the union of the eigenvalues of  $\mathbf{A}$  and the eigenvalues of  $\mathbf{S}$ .

The graphs  $G^{\mathbf{S}}$  that we form this way are called 2-lifts of  $G$ .

Bilu and Linial [BL06] conjectured that every  $d$ -regular graph  $G$  has a signed adjacency matrix  $\mathbf{S}$  so that  $\|\mathbf{S}\| \leq 2\sqrt{d-1}$ . This would give a simple procedure for constructing infinite families of Ramanujan graphs. We would begin with any small  $d$ -regular Ramanujan graph, such as the complete graph on  $d+1$  vertices. Then, given any  $d$ -regular Ramanujan graph we could construct a new Ramanujan graph on twice as many vertices by using  $G^{\mathbf{S}}$  where  $\|\mathbf{S}\| \leq 2\sqrt{d-1}$ .

We will prove something close to their conjecture.

**Theorem 25.2.1.** *Every  $d$ -regular graph  $G$  has a signed adjacency matrix  $\mathbf{S}$  for which the minimum eigenvalue of  $\mathbf{S}$  is at least  $-2\sqrt{d-1}$ .*

We can use this theorem to build infinite families of bipartite Ramanujan graphs, because their eigenvalues are symmetric about the origin. Thus, if  $\mu_n \geq -2\sqrt{d-1}$ , then we know that  $|\mu_i| \leq 2\sqrt{d-1}$  for all  $1 < i < n$ . Note that every 2-lift of a bipartite graph is also a bipartite graph.

### 25.3 Random 2-Lifts

We will prove Theorem 25.2.1 by considering a random 2-lift. In particular, we consider the expected characteristic polynomial of a random signed adjacency matrix  $\mathbf{S}$ :

$$\mathbb{E}_{\mathbf{S}} [\chi_x(\mathbf{S})]. \quad (25.1)$$

Godsil and Gutman [GG81] proved that this is equal to the matching polynomial of  $G$ ! We will learn more about the matching polynomial next lecture.

For now, we just need the following bound on its zeros which was proved by Heilmann and Lieb [HL72].

**Theorem 25.3.1.** *The eigenvalues of the matching polynomial of a graph of maximum degree at most  $d$  are real and have absolute value at most  $2\sqrt{d-1}$ .*

Now that we know that the smallest zero of (25.1) is at least  $-2\sqrt{d-1}$ , all we need to do is to show that there is some signed adjacency matrix whose smallest eigenvalue is at least this bound. This is not necessarily as easy as it sounds, because the smallest zero of the average of two polynomials is not necessarily related to the smallest zeros of those polynomials. We will show that, in this case, it is.

### 25.4 Laplacianized Polynomials

Instead of directly reasoning about the characteristic polynomials of signed adjacency matrices  $\mathbf{S}$ , we will work with characteristic polynomials of  $d\mathbf{I} - \mathbf{S}$ . It suffices for us to prove that there exists an  $\mathbf{S}$  for which the largest eigenvalue of  $d\mathbf{I} - \mathbf{S}$  is at most  $d + 2\sqrt{d-1}$ .

Fix an ordering on the  $m$  edges of the graph, associate each  $\mathbf{S}$  with a vector  $\sigma \in \{\pm 1\}^m$ , and define

$$p_\sigma(x) = \chi_x(d\mathbf{I} - \mathbf{S}).$$

The expected polynomial is the average of all these polynomials.

We define two vectors for each edge in the graph. If the  $i$ th edge is  $(a, b)$ , then we define

$$\mathbf{v}_{i,\sigma_i} = \delta_a - \sigma_i \delta_b.$$

For every  $\sigma \in \{\pm 1\}^m$ , we have

$$\sum_{i=1}^m \mathbf{v}_{i,\sigma_i} \mathbf{v}_{i,\sigma_i}^T = d\mathbf{I} - \mathbf{S},$$

where  $\mathbf{S}$  is the signed adjacency matrix corresponding to  $\sigma$ . So, for every  $\sigma \in \{\pm 1\}^m$ ,

$$p_\sigma(x) = \chi_x \left( \sum_{i=1}^m \mathbf{v}_{i,\sigma_i} \mathbf{v}_{i,\sigma_i}^T \right).$$

## 25.5 Interlacing Families of Polynomials

Here is the problem we face. We have a large family of polynomials, say  $p_1(x), \dots, p_m(x)$ , for which we know each  $p_i$  is real-rooted and that their sum is real rooted. We would like to show that there is some polynomial  $p_i$  whose largest zero is at most the largest zero of the sum. This is not true in general. But, it is true in our case because the polynomials form an *interlacing family*.

For a polynomial  $p(x) = \prod_{i=1}^n (x - \lambda_i)$  of degree  $n$  and a polynomial  $q(x) = \prod_{i=1}^{n-1} (x - \mu_i)$  of degree  $n-1$ , we say that  $q(x)$  *interlaces*  $p(x)$  if

$$\lambda_n \leq \mu_{n-1} \leq \lambda_{n-1} \leq \cdots \leq \lambda_2 \leq \mu_1 \leq \lambda_1.$$

If  $r(x) = \prod_{i=1}^n (x - \mu_i)$  has degree  $n$ , we write  $r(x) \rightarrow p(x)$  if

$$\mu_n \leq \lambda_n \leq \mu_{n-1} \leq \cdots \leq \lambda_2 \leq \mu_1 \leq \lambda_1.$$

That is, if the zeros of  $p$  and  $r$  interlace, with the zeros of  $p$  being larger. We also make these statements if they hold of positive multiples of  $p$ ,  $r$  and  $q$ .

The following lemma gives the examples of interlacing polynomials that motivate us.

**Lemma 25.5.1.** *Let  $\mathbf{A}$  be a symmetric matrix and let  $\mathbf{v}$  be a vector. For a real number  $t$  let*

$$p_t(x) = \chi_x(\mathbf{A} + t\mathbf{v}\mathbf{v}^T).$$

*Then, for  $t > 0$ ,  $p_0(x) \rightarrow p_t(x)$  and there is a monic<sup>1</sup> degree  $n-1$  polynomial  $q(x)$  so that for all  $t$*

$$p_t(x) = \chi_x(\mathbf{A}) - tq(x).$$

*Proof.* The fact that  $p_0(x) \rightarrow p_t(x)$  for  $t > 0$  follows from the Courant-Fischer Theorem.

We first establish the existence of  $q(x)$  in the case that  $\mathbf{v} = \boldsymbol{\delta}_1$ . As the matrix  $t\boldsymbol{\delta}_1\boldsymbol{\delta}_1^T$  is zeros everywhere except for the element  $t$  in the upper left entry and the determinant is linear in each entry of the matrix,

$$\chi_x(\mathbf{A} + t\boldsymbol{\delta}_1\boldsymbol{\delta}_1^T) = \det(x\mathbf{I} - \mathbf{A} - t\boldsymbol{\delta}_1\boldsymbol{\delta}_1^T) = \det(x\mathbf{I} - \mathbf{A}) - t \det(x\mathbf{I}^{(1)} - \mathbf{A}^{(1)}) = \chi_x(\mathbf{A}) - t\chi_x(\mathbf{A}^{(1)}),$$

where  $\mathbf{A}^{(1)}$  is the submatrix of  $\mathbf{A}$  obtained by removing its first row and column. The polynomial  $q(x) = \chi_x(\mathbf{A}^{(1)})$  has degree  $n-1$ .

For arbitrary,  $\mathbf{v}$ , let  $\mathbf{Q}$  be a rotation matrix for which  $\mathbf{Q}\mathbf{v} = \boldsymbol{\delta}_1$ . As determinants, and thus characteristic polynomials, are unchanged by multiplication by rotation matrices,

$$\begin{aligned} \chi_x(\mathbf{A} + t\mathbf{v}\mathbf{v}^T) &= \chi_x(\mathbf{Q}(\mathbf{A} + t\mathbf{v}\mathbf{v}^T)\mathbf{Q}^T) \\ &= \chi_x(\mathbf{Q}\mathbf{A}\mathbf{Q}^T + t\boldsymbol{\delta}_1\boldsymbol{\delta}_1^T)) = \chi_x(\mathbf{Q}\mathbf{A}\mathbf{Q}^T) - tq(x) = \chi_x(\mathbf{A}) - tq(x), \end{aligned}$$

for some  $q(x)$  of degree  $n-1$ . □

---

<sup>1</sup>A *monic* polynomial is one whose leading coefficient is 1.

For a polynomial  $p$ , let  $\lambda_{\max}(p)$  denote its largest zero. When polynomials interlace, we can relate the largest zero of their sum to the largest zero of at least one of them.

**Lemma 25.5.2.** *Let  $p_1(x)$ ,  $p_2(x)$  and  $r(x)$  be polynomials so that  $r(x) \rightarrow p_i(x)$ . Then,  $r(x) \rightarrow p_1(x) + p_2(x)$  and there is an  $i \in \{1, 2\}$  for which*

$$\lambda_{\max}(p_i) \leq \lambda_{\max}(p_1 + p_2).$$

*Proof.* Let  $\mu_1$  be the largest zero of  $r(x)$ . As each polynomial  $p_i(x)$  has a positive leading coefficient, each is eventually positive and so is their sum. As each has exactly one zero that is at least  $\mu_1$  each is nonpositive at  $\mu_1$ , and the same is also true of their sum. Let  $\lambda$  be the largest zero of  $p_1 + p_2$ . We have established that  $\lambda \geq \mu_1$ .

If  $p_i(\lambda) = 0$  for some  $i$ , then we are done. If not, there is an  $i$  for which  $p_i(\lambda) > 0$ . As  $p_i$  only has one zero larger than  $\mu_1$ , and it is eventually positive, the largest zero of  $p_i$  must be less than  $\lambda$ .  $\square$

If  $p_1, \dots, p_m$  are polynomials such that there exists an  $r(x)$  for which  $r(x) \rightarrow p_i(x)$  for all  $i$ , then these polynomials are said to have a *common interlacing*. Such polynomials satisfy the natural generalization of Lemma 25.5.2.

The polynomials  $p_\sigma(x)$  do not all have a common interlacing. However, they satisfy a property that is just as useful: they form an *interlacing family*. Rather than defining these in general, we will just explain the special case we need for today's theorem.

We define polynomials that correspond to fixing the signs of the first  $k$  edges and then choosing the rest at random. We indicate these by shorter sequences  $\sigma \in \{\pm 1\}^k$ . For  $k < m$  and  $\sigma \in \{\pm 1\}^k$  we define

$$p_\sigma(x) \stackrel{\text{def}}{=} \mathbb{E}_{\rho \in \{\pm 1\}^{n-k}} [p_{\sigma, \rho}(x)].$$

So,

$$p_\emptyset(x) = \mathbb{E}_{\sigma \in \{\pm 1\}^m} [p_\sigma(x)].$$

We view the strings  $\sigma$ , and thus the polynomials  $p_\sigma$ , as vertices in a complete binary tree. The nodes with  $\sigma$  of length  $m$  are the leaves, and  $\emptyset$  corresponds to the root. For  $\sigma$  of length less than  $n$ , the children of  $\sigma$  are  $(\sigma, 1)$  and  $(\sigma, -1)$ . We call such a pair of nodes *siblings*. We will eventually prove in Lemma 25.6.1 that all the polynomials  $p_\sigma(x)$  are real rooted and in Corollary 25.6.2 that every pair of siblings has a common interlacing.

But first, we show that this implies that there is a leaf indexed by  $\sigma \in \{\pm 1\}^m$  for which

$$\lambda_{\max}(p_\sigma) \leq \lambda_{\max}(p_\emptyset).$$

This implies Theorem 25.2.1, as we know from Theorem 25.3.1 that  $\lambda_{\max}(p_\emptyset) \leq d + 2\sqrt{d-1}$ .

**Lemma 25.5.3.** *There is a  $\sigma \in \{\pm 1\}^m$  for which*

$$\lambda_{\max}(p_\sigma) \leq \lambda_{\max}(p_\emptyset).$$

*Proof.* Corollary 25.6.2 and Lemma 25.5.2 imply that every non-leaf node in the tree has a child whose largest zero is at most the largest zero of that node. Starting at the root of the tree, we find a node whose largest zero is at most the largest zero of  $p_\emptyset$ . We then proceed down the tree until we reach a leaf, at each step finding a node labeled by a polynomial whose largest zero is at most the largest zero of the previous polynomial. The leaf we reach,  $\sigma$ , satisfies the desired inequality.  $\square$

## 25.6 Common Interlacings

We can now use Lemmas 25.5.1 and 25.5.2 to show that every  $\sigma \in \{\pm 1\}^{m-1}$  has a child  $(\sigma, s)$  for which  $\lambda_{\max}(p_{\sigma,s}) \leq \lambda_{\max}(p_\sigma)$ . Let

$$\mathbf{A} = \sum_{i=1}^{m-1} \mathbf{v}_{i,\sigma_i} \mathbf{v}_{i,\sigma_i}^T.$$

The children of  $\sigma$ ,  $(\sigma, 1)$  and  $(\sigma, -1)$  have polynomials  $p_{(\sigma,1)}$  and  $p_{(\sigma,-1)}$  that equal

$$\chi_x(\mathbf{A} + \mathbf{v}_{m,1} \mathbf{v}_{m,1}^T) \quad \text{and} \quad \chi_x(\mathbf{A} + \mathbf{v}_{m,-1} \mathbf{v}_{m,-1}^T).$$

By Lemma 25.5.1,  $\chi_x(\mathbf{A}) \rightarrow \chi_x(\mathbf{A} + \mathbf{v}_{m,s} \mathbf{v}_{m,s}^T)$  for  $s \in \{\pm 1\}$ , and Lemma 25.5.2 implies that there is an  $s$  for which the largest zero of  $p_{(\sigma,s)}$  is at most the largest zero of their average, which is  $p_\sigma$ .

To extend this argument to nodes higher up in the tree, we will prove the following statement.

**Lemma 25.6.1.** *Let  $\mathbf{A}$  be a symmetric matrix and let  $\mathbf{w}_{i,s}$  be vectors for  $1 \leq i \leq k$  and  $s \in \{0, 1\}$ . Then the polynomial*

$$\sum_{\rho \in \{0,1\}^k} \chi_x \left( \mathbf{A} + \sum_{i=1}^k \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T \right)$$

*is real rooted, and for each  $s \in \{0, 1\}$ ,*

$$\sum_{\rho \in \{0,1\}^k} \chi_x \left( \mathbf{A} + \sum_{i=1}^{k-1} \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T \right) \rightarrow \sum_{\rho \in \{0,1\}^k} \chi_x \left( \mathbf{A} + \sum_{i=1}^{k-1} \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T + \mathbf{w}_{k,s} \mathbf{w}_{k,s}^T \right).$$

**Corollary 25.6.2.** *For every  $k < n$  and  $\sigma \in \{\pm 1\}^k$ , the polynomials  $p_{\sigma,s}(x)$  for  $s \in \{\pm 1\}$  are real rooted and have a common interlacing.*

## 25.7 Real Rootedness

To prove Lemma 25.6.1, we use the following two lemmas which are known collectively as Obreschkoff's Theorem [Obr63].

**Lemma 25.7.1.** *Let  $p$  and  $q$  be polynomials of degree  $n$  and  $n-1$ , and let  $p_t(x) = p(x) - tq(x)$ . If  $p_t$  is real rooted for all  $t \in \mathbb{R}$ , then  $q$  interlaces  $p$ .*

*Proof Sketch.* Recall that the roots of a polynomial are continuous functions of its coefficients, and thus the roots of  $p_t$  are continuous functions of  $t$ . We will use this fact to obtain a contradiction.

For simplicity,<sup>2</sup> I just consider the case in which all of the roots of  $p$  and  $q$  are distinct. If they are not, one can prove this by dividing out their common divisors.

If  $p$  and  $q$  do not interlace, then  $p$  must have two roots that do not have a root of  $q$  between them. Let these roots of  $p$  be  $\lambda_{i+1}$  and  $\lambda_i$ . Assume, without loss of generality, that both  $p$  and  $q$  are positive between these roots. We now consider the behavior of  $p_t$  for positive  $t$ .

As we have assumed that the roots of  $p$  and  $q$  are distinct,  $q$  is positive at these roots, and so  $p_t$  is negative at  $\lambda_{i+1}$  and  $\lambda_i$ . If  $t$  is very small, then  $p_t$  will be close to  $p$  in value, and so there must be some small  $t_0$  for which  $p_{t_0}(x) > 0$  for some  $\lambda_{i+1} < x < \lambda_i$ . This means that  $p_{t_0}$  must have two roots between  $\lambda_{i+1}$  and  $\lambda_i$ .

As  $q$  is positive on the entire closed interval  $[\lambda_{i+1}, \lambda_i]$ , when  $t$  is large  $p_t$  will be negative on this entire interval, and thus have no roots inside. As we vary  $t$  between  $t_0$  and infinity, the two roots at  $t_0$  must vary continuously and cannot cross  $\lambda_{i+1}$  or  $\lambda_i$ . This means that they must become complex, contradicting our assumption that  $p_t$  is always real rooted.  $\square$

**Lemma 25.7.2.** *Let  $p$  and  $q$  be polynomials of degree  $n$  and  $n - 1$  that interlace and have positive leading coefficients. For every  $t > 0$ , define  $p_t(x) = p(x) - tq(x)$ . Then,  $p_t(x)$  is real rooted and*

$$p(x) \rightarrow p_t(x).$$

*Proof Sketch.* For simplicity, I consider the case in which all of the roots of  $p$  and  $q$  are distinct. One can prove the general case by dividing out the common repeated roots.

To see that the largest root of  $p_t$  is larger than  $\lambda_1$ , note that  $q(x)$  is positive for all  $x > \mu_1$ , and  $\lambda_1 > \mu_1$ . So,  $p_t(\lambda_1) = p(\lambda_1) - tq(\lambda_1) < 0$ . As  $p_t$  is monic, it is eventually positive and it must have a root larger than  $\lambda_1$ .

We will now show that for every  $i \geq 1$ ,  $p_t$  has a root between  $\lambda_{i+1}$  and  $\lambda_i$ . As this gives us  $d - 1$  more roots, it accounts for all  $d$  roots of  $p_t$ . For  $i$  odd, we know that  $q(\lambda_i) > 0$  and  $q(\lambda_{i+1}) < 0$ . As  $p$  is zero at both of these points,  $p_t(\lambda_i) > 0$  and  $p_t(\lambda_{i+1}) < 0$ , which means that  $p_t$  has a root between  $\lambda_i$  and  $\lambda_{i+1}$ . The case of even  $i$  is similar.  $\square$

**Lemma 25.7.3.** *Let  $p_0(x)$  and  $p_1(x)$  be degree  $n$  monic polynomials for which there is a third polynomial  $r(x)$  Such that*

$$r(x) \rightarrow p_0(x) \quad \text{and} \quad r(x) \rightarrow p_1(x).$$

*Then*

$$r(x) \rightarrow (1/2)p_0(x) + (1/2)p_1(x),$$

*and the latter is a real rooted polynomial.*

*Sketch.* Assume for simplicity that all the roots of  $r$  are distinct and different from the roots of  $p_0$  and  $p_1$ . Let  $\mu_n < \mu_{n-1} < \dots < \mu_1$  be the roots of  $r$ . Our assumptions imply that both  $p_0$  and  $p_1$

---

<sup>2</sup>I thank Sushant Sachdeva for helping me work out this particularly simple proof.

are negative at  $\mu_i$  for odd  $i$  and positive for even  $i$ . So, the same is true of their average. This tells us that their average must have at least  $n - 1$  real roots between  $\mu_n$  and  $\mu_1$ . As their average is monic, it must be eventually positive and so must have a root larger than  $\mu_1$ . That accounts for all  $n$  of its roots.  $\square$

*Proof of Lemma 25.6.1.* We prove this by induction on  $k$ . Assuming that we have proved it for  $k - 1$ , we now prove it for  $k$ . Let  $\mathbf{u}$  be any vector and let  $t \in \mathbb{R}$ . Define

$$p_t(x) = \sum_{\rho \in \{0,1\}^k} \chi_x \left( \mathbf{A} + \sum_{i=1}^{k-1} \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T + t \mathbf{u} \mathbf{u}^T \right).$$

By Lemma 25.5.1, we can express this polynomial in the form

$$p_t(x) = p_0(x) - tq(x),$$

where  $q$  has positive leading coefficient and degree  $n - 1$ . By absorbing  $t \mathbf{u} \mathbf{u}^T$  into  $\mathbf{A}$  we may use induction on  $k$  to show that  $p_t(x)$  is real rooted for all  $t$ . Thus, Lemma 25.7.1 implies that  $q(x)$  interlaces  $p_0(x)$ , and Lemma 25.7.2 tells us that for  $t > 0$

$$p_0(x) \rightarrow p_t(x).$$

So, we may conclude that for every  $s \in \{\pm 1\}$ ,

$$\sum_{\rho \in \{0,1\}^{k-1}} \chi_x \left( \mathbf{A} + \sum_{i=1}^{k-1} \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T \right) \rightarrow \sum_{\rho \in \{0,1\}^k} \chi_x \left( \mathbf{A} + \sum_{i=1}^{k-1} \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T + \mathbf{w}_{k,s} \mathbf{w}_{k,s}^T \right).$$

So, Lemma 25.7.3 implies that

$$\sum_{\rho \in \{0,1\}^{k-1}} \chi_x \left( \mathbf{A} + \sum_{i=1}^{k-1} \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T \right) \rightarrow \sum_{\rho \in \{0,1\}^k} \chi_x \left( \mathbf{A} + \sum_{i=1}^k \mathbf{w}_{i,\rho_i} \mathbf{w}_{i,\rho_i}^T \right)$$

and that the latter polynomial is real rooted.  $\square$

## 25.8 Conclusion

The major open problem left by this work is establishing the existence of regular (non-bipartite) Ramanujan graphs. The reason we can not prove this using the techniques in this lecture is that the interlacing techniques only allow us to reason about the largest or smallest eigenvalue of a matrix, but not both.

To see related papers establishing the existence of Ramanujan graphs, see [MSS15b, HPS15]. For a survey on this and related material, see [MSS14].

## References

- [BL06] Yonatan Bilu and Nathan Linial. Lifts, discrepancy and nearly optimal spectral gap\*. *Combinatorica*, 26(5):495–519, 2006.
- [Chi92] Patrick Chiu. Cubic Ramanujan graphs. *Combinatorica*, 12(3):275–285, 1992.
- [Fri08] Joel Friedman. *A Proof of Alon’s Second Eigenvalue Conjecture and Related Problems*. Number 910 in Memoirs of the American Mathematical Society. American Mathematical Society, 2008.
- [GG81] C. D. Godsil and I. Gutman. On the matching polynomial of a graph. In L. Lovász and Vera T. Sós, editors, *Algebraic Methods in graph theory*, volume I of *Colloquia Mathematica Societatis János Bolyai*, 25, pages 241–249. János Bolyai Mathematical Society, 1981.
- [HL72] Ole J Heilmann and Elliott H Lieb. Theory of monomer-dimer systems. *Communications in Mathematical Physics*, 25(3):190–232, 1972.
- [HPS15] Chris Hall, Doron Puder, and William F Sawin. Ramanujan coverings of graphs. *arXiv preprint arXiv:1506.02335*, 2015.
- [JL97] Bruce W Jordan and Ron Livné. Ramanujan local systems on graphs. *Topology*, 36(5):1007–1024, 1997.
- [LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.
- [Mar88] G. A. Margulis. Explicit group theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators. *Problems of Information Transmission*, 24(1):39–46, July 1988.
- [MNS08] Steven J. Miller, Tim Novikoff, and Anthony Sabelli. The distribution of the largest non-trivial eigenvalues in families of random regular graphs. *Experiment. Math.*, 17(2):231–244, 2008.
- [Mor94] M. Morgenstern. Existance and explicit constructions of  $q+1$  regular Ramanujan graphs for every prime power  $q$ . *Journal of Combinatorial Theory, Series B*, 62:44–62, 1994.
- [MSS14] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Ramanujan graphs and the solution of the Kadison-Singer problem. In *Proceedings of the International Congress of Mathematicians*, 2014.
- [MSS15a] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Interlacing families I: Bipartite Ramanujan graphs of all degrees. *Ann. of Math.*, 182-1:307–325, 2015.
- [MSS15b] Adam W Marcus, Nikhil Srivastava, and Daniel A Spielman. Interlacing families IV: Bipartite Ramanujan graphs of all sizes. *arXiv preprint arXiv:1505.08010*, 2015. appeared in Proceedings of the 56th IEEE Symposium on Foundations of Computer Science.

- [Obr63] Nikola Obrechkoff. *Verteilung und berechnung der Nullstellen reeller Polynome.* VEB Deutscher Verlag der Wissenschaften, Berlin, 1963.
- [Piz90] Arnold K Pizer. Ramanujan graphs and Hecke operators. *Bulletin of the AMS*, 23(1), 1990.

## 26.1 Overview

The coefficients of the matching polynomial of a graph count the numbers of matchings of various sizes in that graph. It was first defined by Heilmann and Lieb [HL72], who proved that it has some amazing properties, including that it is real rooted. They also proved that all roots of the matching polynomial of a graph of maximum degree  $d$  are at most  $2\sqrt{d-1}$ . Our proofs today come from a different approach to the matching polynomial that appears in the work of Godsil [God93, God81]. A theorem of Godsil and Gutman [GG81] implies that the expected characteristic polynomial of a randomly signed adjacency matrix is the matching polynomial of a graph. Last lecture we used these results to establish the existence of infinite families of bipartite Ramanujan graphs.

## 26.2 $2\sqrt{d-1}$

We begin by explaining where the number  $2\sqrt{d-1}$  comes from: it is an upper bound on the eigenvalues of a tree of maximum degree at most  $d$ . One can also show that the largest eigenvalue of an  $d$ -ary tree approaches  $2\sqrt{d-1}$  as the depth of the tree (and number of vertices) increases.

We prove this statement in two steps. The first is similar to proofs we saw at the beginning of the semester.

**Lemma 26.2.1.** *Let  $\mathbf{M}$  be a (not necessarily symmetric) nonnegative matrix. Let  $s = \|\mathbf{M}\mathbf{1}\|_\infty$  be the maximum row sum of  $\mathbf{M}$ . Then,  $|\lambda| \leq s$  for every eigenvalue of  $\mathbf{M}$ .*

*Proof.* Let  $\mathbf{M}\psi = \lambda\psi$ , and let  $a$  be an entry of  $\psi$  of largest absolute value. Then,

$$\begin{aligned} |\lambda| |\psi(a)| &= |\lambda\psi(a)| \\ &= |(\mathbf{M}\psi)(a)| \\ &= \left| \sum_b \mathbf{M}(b, a)\psi(b) \right| \\ &\leq \left| \sum_b \mathbf{M}(b, a) \right| |\psi(b)| \\ &\leq s |\psi(b)|. \end{aligned}$$

This implies  $|\lambda| \leq s$ . □

**Theorem 26.2.2.** *Let  $T$  be a tree in which every vertex has degree at most  $d$ . Then, all eigenvalues of  $\chi_x(\mathbf{M}_T)$  have absolute value at most  $2\sqrt{d-1}$ .*

*Proof.* Let  $\mathbf{M}$  be the adjacency matrix of  $T$ . Choose some vertex to be the root of the tree, and define its height to be 0. For every other vertex  $a$ , define its height,  $h(a)$ , to be its distance to the root. Define  $\mathbf{D}$  to be the diagonal matrix with

$$\mathbf{D}(a, a) = \left( \sqrt{d-1} \right)^{h(a)}.$$

Recall that the eigenvalues of  $\mathbf{M}$  are the same as the eigenvalues of  $\mathbf{DMD}^{-1}$ . We will use the fact that all eigenvalues of a nonnegative matrix are upper bounded in absolute value by its maximum row sum.

So, we need to prove that all row sums of  $\mathbf{DMD}^{-1}$  are at most  $2\sqrt{d-1}$ . There are three types of vertices to consider. First, the row of the root has up to  $d$  entries that are all  $1/\sqrt{d-1}$ . For  $d \geq 2$ ,  $d/\sqrt{d-1} \leq 2\sqrt{d-1}$ . The intermediate vertices have one entry in their row that equals  $\sqrt{d-1}$ , and up to  $d-1$  entries that are equal to  $1/\sqrt{d-1}$ , for a total of  $2\sqrt{d-1}$ . Finally, every leaf only has one nonzero entry in its row, and that entry equals  $\sqrt{d-1}$ .  $\square$

## 26.3 The Matching Polynomial

A matching in a graph  $G = (V, E)$  is a subgraph of  $G$  in which every vertex has degree 1. We say that a matching has size  $k$  if it has  $k$  edges. We let

$$m_k(G)$$

denote the number of matchings in  $G$  of size  $k$ . Throughout this lecture, we let  $|V| = n$ . Observe that  $m_1(G)$  is the number of edges in  $G$ , and that  $m_{n/2}(G)$  is the number of perfect matchings in  $G$ . By convention we set  $m_0(G) = 1$ , as the empty set is matching with no edges. Computing the number of perfect matchings is a  $\#P$ -hard problem [Val79]. This means that it is much harder than solving  $NP$ -hard problems, so you shouldn't expect to do it quickly on large graphs.

The *matching polynomial* of  $G$ , written  $\mu_x[G]$ , is

$$\mu_x[G] \stackrel{\text{def}}{=} \sum_{k=0}^{n/2} x^{n-2k} (-1)^k m_k(G).$$

Our convention that  $m_0(G) = 1$  ensures that this is a polynomial of degree  $n$ .

This is a fundamental example of a polynomial that is defined so that its coefficients count something. When the “something” is interesting, the polynomial usually is as well.

Godsil and Gutman [GG81] proved that this is equal to the matching polynomial of  $G$ !

**Lemma 26.3.1.** *Let  $G$  be a graph and let  $\mathbf{S}$  be a uniform random signed adjacency matrix of  $G$ . Then,*

$$\mathbb{E} [\chi_x(\mathbf{S})] = \mu_x[G].$$

*Proof.* Expand the expected characteristic polynomial as

$$\begin{aligned}
\mathbb{E} [\chi_x(\mathbf{S})] &= \mathbb{E} [\det(x\mathbf{I} - \mathbf{S})] \\
&= \mathbb{E} [\det(x\mathbf{I} + \mathbf{S})] \\
&= \mathbb{E} \left[ \sum_{\pi \in S_n} \text{sgn}(\pi) x^{|\{a: \pi(a)=a\}|} \prod_{a: \pi(a) \neq a} (\mathbf{S}(a, \pi(a))) \right] \\
&= \sum_{\pi \in S_n} \text{sgn}(\pi) x^{|\{a: \pi(a)=a\}|} \mathbb{E} \left[ \prod_{a: \pi(a) \neq a} (\mathbf{S}(a, \pi(a))) \right].
\end{aligned}$$

As  $\mathbb{E} [\mathbf{S}(a, \pi(a))] = 0$  for every  $a$  so that  $\pi(a) \neq a$ , the only way we can get a nonzero contribution from a permutation  $\pi$  is if for all  $a$  so that  $\pi(a) \neq a$ ,

- a.  $(a, \pi(a)) \in E$ , and
- b.  $\pi(\pi(a)) = a$ .

The latter condition guarantees that whenever  $\mathbf{S}(a, \pi(a))$  appears in the product,  $\mathbf{S}(\pi(a), a)$  does as well. As these entries are constrained to be the same, their product is 1.

Thus, the only permutations that count are the involutions (the permutations in which all cycles have length 1 or 2). These correspond exactly to the matchings in the graph. Finally, the sign of an involution is exactly its number of two-cycles, which is exactly its number of edges.  $\square$

We will prove that the matching polynomial of every  $d$ -regular graph divides the matching polynomial of a larger tree of maximum degree  $d$ .

The matching polynomials of trees are very special—they are exactly the same as the characteristic polynomial of the adjacency matrix.

**Theorem 26.3.2.** *Let  $G$  be a tree and let  $\mathbf{M}$  be its adjacency matrix. Then*

$$\mu_x[G] = \chi_x(\mathbf{M}).$$

*Proof.* Expand

$$\chi_x(\mathbf{M}) = \det(x\mathbf{I} - \mathbf{M})$$

by summing over permutations. We obtain

$$\sum_{\pi \in S_n} \text{sgn}(\pi) x^{|\{a: \pi(a)=a\}|} \prod_{a: \pi(a) \neq a} (-\mathbf{M}(a, \pi(a))).$$

We will prove that the only permutations that contribute to this sum are those for which  $\pi(\pi(a)) = a$  for every  $a$ . And, these correspond to matchings.

If  $\pi$  is a permutation for which there is an  $a$  so that  $\pi(\pi(a)) \neq a$ , then there are  $a = a_1, \dots, a_k$  with  $k > 2$  so that  $\pi(a_i) = a_{i+1}$  for  $1 \leq i < k$ , and  $\pi(a_k) = a_1$ . For this term to contribute, it must be the case that  $M(a_i, a_{i+1}) = 1$  for all  $i$ , and that  $M(a_k, a_1) = 1$ . For  $k > 2$ , this would be a cycle of length  $k$  in  $G$ . However,  $G$  is a tree and so cannot have a cycle.

So, the only permutations that contribute are the *involutions*: the permutations  $\pi$  that are their own inverse. An involution has only fixed points and cycles of length 2. Each cycle of length 2 that contributes a nonzero term corresponds to an edge in the graph. Thus, the number of permutations with  $k$  cycles of length 2 is equal to the number of matchings with  $k$  edges. As the sign of an involution with  $k$  cycles of length 2 is  $(-1)^k$ , the coefficient of  $x^{n-2k}$  is  $(-1)^k m_k(G)$ .  $\square$

## 26.4 Properties of the Matching Polynomial

We begin by establishing some fundamental properties of the matching polynomial. For graphs  $G$  and  $H$  on different vertex sets, we write  $G \cup H$  for their disjoint union.

**Lemma 26.4.1.** *Let  $G$  and  $H$  be graphs on different vertex sets. Then,*

$$\mu_x[G \cup H] = \mu_x[G] \mu_x[H].$$

*Proof.* Every matching in  $G \cup H$  is the union of a matching in  $G$  and a matching in  $H$ . Thus,

$$m_k(G \cup H) = \sum_{j=0}^k m_j(G)m_{k-j}(H).$$

The lemma follows.  $\square$

For  $a$  a vertex of  $G = (V, E)$ , we write  $G - a$  for the graph  $G(V - \{a\})$ . This notation will prove very useful when reasoning about matching polynomials. Fix a vertex  $a$  of  $G$ , and divide the matchings in  $G$  into two classes: those that involve vertex  $a$  and those that do not. The number of matchings of size  $k$  that do not involve  $a$  is  $m_k(G - a)$ . On the other hand, those that do involve  $a$  connect  $a$  to one of its neighbors. To count these, we enumerate the neighbors  $b$  of  $a$ . A matching of size  $k$  that includes edge  $(a, b)$  can be written as the union of  $(a, b)$  and a matching of size  $k - 1$  in  $G - a - b$ . So, the number of matchings that involve  $a$  is

$$\sum_{b \sim a} m_{k-1}(G - a - b).$$

This gives a recurrence for the number of matchings of size  $k$  in  $G$ :

$$m_k(G) = m_k(G - a) + \sum_{b \sim a} m_{k-1}(G - a - b).$$

To turn this into a recurrence for  $\mu_x[G]$ , write

$$x^{n-2k}(-1)^k m_k(G) = x \cdot x^{(n-1)-2k}(-1)^k m_k(G - a) - x^{(n-2)-2(k-1)}(-1)^{k-1} m_{k-1}(G - a - b).$$

This establishes the following formula.

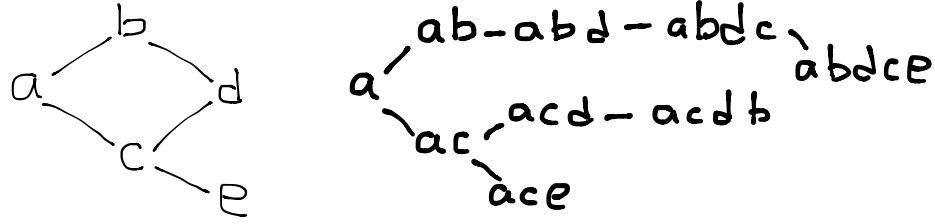
**Lemma 26.4.2.**

$$\mu_x[G] = x\mu_x[G - a] - \sum_{b \sim a} \mu_x[G - a - b].$$

## 26.5 The Path Tree

Godsil proves that the matching polynomial of a graph is real rooted by proving that it divides the matching polynomial of a tree. Moreover, the maximum degree of vertices in the tree is at most the maximum degree of vertices in the graph. As the matching polynomial of a tree is the same as its characteristic polynomial, and all zeros of the characteristic polynomial of a tree of maximum degree at most  $d$  have absolute value at most  $2\sqrt{d-1}$ , all the zeros of the matching polynomial of a  $d$ -regular graph have absolute value at most  $2\sqrt{d-1}$ .

The tree that Godsил uses is the *path tree* of  $G$  starting at a vertex of  $G$ . For  $a$  a vertex of  $G$ , the path tree of  $G$  starting at  $a$ , written  $T_a(G)$  is a tree whose vertices correspond to paths in  $G$  that start at  $a$  and do not contain any vertex twice. One path is connected to another if one extends the other by one vertex. For example, here is a graph and its path tree starting at  $a$ .



When  $G$  is a tree,  $T_a(G)$  is isomorphic to  $G$ .

Godsil's proof begins by deriving a somewhat strange equality. Since I haven't yet found a better proof, I'll take this route too.

**Theorem 26.5.1.** *For every graph  $G$  and vertex  $a$  of  $G$ ,*

$$\frac{\mu_x[G]}{\mu_x[G - a]} = \frac{\mu_x[T_a(G)]}{\mu_x[T_a(G) - a]}.$$

The term on the upper-right hand side is a little odd. It is a forest obtained by removing the root of the tree  $T_a(G)$ . We may write it as a disjoint union of trees as

$$T_a(G) - a = \bigcup_{b \sim a} T_b(G - a).$$

Before proving this, we use it to prove our main theorem.

**Theorem 26.5.2.** *For every vertex  $a$  of  $G$ , the polynomial  $\mu_x[G]$  divides the polynomial  $\mu_x[T_a(G)]$ .*

*Proof.* We prove this by induction on the number of vertices in  $G$ , using as our base case graphs with at most 2 vertices. We then know, by induction, that for  $b \sim a$ ,

$$\mu_x [G - a] \text{ divides } \mu_x [T_b(G - a)].$$

As

$$\begin{aligned} T_a(G) - a &= \cup_{b \sim a} T_b(G - a), \\ \mu_x [T_b(G - a)] &\text{ divides } \mu_x [T_a(G) - a]. \end{aligned}$$

Thus,

$$\mu_x [G - a] \text{ divides } \mu_x [T_a(G) - a],$$

and so

$$\frac{\mu_x [T_a(G) - a]}{\mu_x [G - a]}$$

is a polynomial in  $x$ . To finish the proof, we apply Theorem 26.5.1, which implies

$$\mu_x [T_a(G)] = \mu_x [T_a(G) - a] \frac{\mu_x [G]}{\mu_x [G - a]} = \mu_x [G] \frac{\mu_x [T_a(G) - a]}{\mu_x [G - a]}.$$

□

*Proof of Theorem 26.5.1.* If  $G$  is a tree, then the left and right sides are identical, and so the equality holds. As the only graphs on less than 3 vertices are trees, the theorem holds for all graphs on at most 2 vertices. We will now prove it by induction on the number of vertices.

We may use Lemma 26.4.2 to expand the left-hand side:

$$\frac{\mu_x [G]}{\mu_x [G - a]} = \frac{x\mu_x [G - a] - \sum_{b \sim a} \mu_x [G - a - b]}{\mu_x [G - a]} = x - \sum_{b \sim a} \frac{\mu_x [G - a - b]}{\mu_x [G - a]}.$$

By applying the inductive hypothesis to  $G - a$ , we see that this equals

$$x - \sum_{b \sim a} \frac{\mu_x [T_b(G - a) - b]}{\mu_x [T_b(G - a)]}. \quad (26.1)$$

To simplify this expression, we examine these graphs carefully. By the observation we made before the proof,

$$T_b(G - a) - b = \bigcup_{c \sim b, c \neq a} T_c(G - a - b).$$

Similarly,

$$T_a(G) - a = \bigcup_{c \sim a} T_c(G - a),$$

which implies

$$\mu_x [T_a(G) - a] = \prod_{c \sim a} \mu_x [T_c(G - a)].$$

Let  $ab$  be the vertex in  $T_a(G)$  corresponding to the path from  $a$  to  $b$ . We also have

$$\begin{aligned} T_a(G) - a - ab &= \left( \bigcup_{c \sim a, c \neq b} T_c(G - a) \right) \cup \left( \bigcup_{c \sim b, c \neq a} T_c(G - a - b) \right) \\ &= \left( \bigcup_{c \sim a, c \neq b} T_c(G - a) \right) \cup (T_b(G - a) - b). \end{aligned}$$

which implies

$$\mu_x [T_a(G) - a - ab] = \left( \prod_{c \sim a, c \neq b} \mu_x [T_c(G - a)] \right) \mu_x [T_b(G - a) - b].$$

Thus,

$$\begin{aligned} \frac{\mu_x [T_a(G) - a - ab]}{\mu_x [T_a(G) - a]} &= \frac{\left( \prod_{c \sim a, c \neq b} \mu_x [T_c(G - a)] \right) \mu_x [T_b(G - a) - b]}{\prod_{c \sim a} \mu_x [T_c(G - a)]} \\ &= \frac{\mu_x [T_b(G - a) - b]}{\mu_x [T_b(G - a)]}. \end{aligned}$$

Plugging this in to (26.1), we obtain

$$\begin{aligned} \frac{\mu_x [G]}{\mu_x [G - a]} &= x - \sum_{b \sim a} \frac{\mu_x [T_a(G) - a - ab]}{\mu_x [T_a(G) - a]} \\ &= \frac{x \mu_x [T_a(G) - a] - \sum_{b \sim a} \mu_x [T_a(G) - a - ab]}{\mu_x [T_a(G) - a]} \\ &= \frac{\mu_x [T_a(G)]}{\mu_x [T_a(G) - a]}. \end{aligned}$$

□

## References

- [GG81] C. D. Godsil and I. Gutman. On the matching polynomial of a graph. In L. Lovász and Vera T. Sós, editors, *Algebraic Methods in graph theory*, volume I of *Colloquia Mathematica Societatis János Bolyai*, 25, pages 241–249. János Bolyai Mathematical Society, 1981.
- [God81] C. D. Godsil. Matchings and walks in graphs. *Journal of Graph Theory*, 5(3):285–297, 1981.
- [God93] Chris Godsil. *Algebraic Combinatorics*. Chapman & Hall, 1993.
- [HL72] Ole J Heilmann and Elliott H Lieb. Theory of monomer-dimer systems. *Communications in Mathematical Physics*, 25(3):190–232, 1972.
- [Val79] Leslie G Valiant. The complexity of computing the permanent. *Theoretical computer science*, 8(2):189–201, 1979.

---

# **Chapter 16**

---

## **Spectral Graph Theory**

**Daniel Spielman**

*Yale University*

16.1	Introduction .....	1
16.2	Preliminaries .....	2
16.3	The matrices associated with a graph .....	2
16.3.1	Operators on the vertices .....	3
16.3.2	The Laplacian Quadratic Form .....	4
16.3.3	The Normalized Laplacian .....	5
16.3.4	Naming the Eigenvalues .....	5
16.4	Some Examples .....	6
16.5	The Role of the Courant-Fischer Theorem .....	9
16.5.1	Low-Rank Approximations .....	9
16.6	Elementary Facts .....	10
16.7	Spectral Graph Drawing .....	11
16.8	Algebraic Connectivity and Graph Partitioning .....	11
16.8.1	Convergence of Random Walks .....	14
16.8.2	Expander Graphs .....	14
16.8.3	Ramanujan Graphs .....	16
16.8.4	Bounding $\lambda_2$ .....	16
16.9	Coloring and Independent Sets .....	17
16.10	Perturbation Theory and Random Graphs .....	18
16.11	Relative Spectral Graph Theory .....	20
16.12	Directed Graphs .....	21
16.13	Concluding Remarks .....	22
	Bibliography .....	23

---

### **16.1 Introduction**

Spectral graph theory is the study and exploration of graphs through the eigenvalues and eigenvectors of matrices naturally associated with those graphs. It is intuitively related to attempts to understand graphs through the simulation of processes on graphs and through the consideration of physical systems related to graphs. Spectral graph theory provides many useful algorithms, as well as some that can be rigorously analyzed. We begin this chapter by providing intuition as to why interesting properties of graphs should be revealed by these eigenvalues and eigenvectors. We then survey a few applications of spectral graph theory.

The figures in this chapter are accompanied by the Matlab code used to generate them.

---

## 16.2 Preliminaries

We ordinarily view an undirected graph<sup>1</sup>  $G$  as a pair  $(V, E)$ , where  $V$  denotes its set of vertices and  $E$  denotes its set of edges. Each edge in  $E$  is an unordered pair of vertices, with the edge connecting distinct vertices  $a$  and  $b$  written as  $(a, b)$ . A weighted graph is a graph in which a weight (typically a real number) has been assigned to every edge. We denote a weighted graph by a triple  $(V, E, w)$ , where  $(V, E)$  is the associated unweighted graph, and  $w$  is a function from  $E$  to the real numbers. We restrict our attention to weight functions  $w$  that are strictly positive. We reserve the letter  $n$  for the number of vertices in a graph. The *degree* of a vertex in an unweighted graph is the number of edges in which it is involved. We say that a graph is *regular* if every vertex has the same degree, and  $d$ -regular if that degree is  $d$ .

We denote vectors by bold letters, and denote the  $i$ th component of a vector  $\mathbf{x}$  by  $\mathbf{x}(i)$ . Similarly, we denote the entry in the  $i$ th row and  $j$ th column of a matrix  $M$  by  $M(i, j)$ .

If we are going to discuss the eigenvectors and eigenvalues of a matrix  $M$ , we should be sure that they exist. When considering undirected graphs, most of the matrices we consider are symmetric, and thus they have an orthonormal basis of eigenvectors and  $n$  eigenvalues, counted with multiplicity. The other matrices we associate with undirected graphs are *similar* to symmetric matrices, and thus also have  $n$  eigenvalues, counted by multiplicity, and possess a basis of eigenvectors. In particular, these matrices are of the form  $MD^{-1}$ , where  $M$  is symmetric and  $D$  is a non-singular diagonal matrix. In this case,  $D^{-1/2}MD^{-1/2}$  is symmetric, and we have

$$D^{-1/2}MD^{-1/2}\mathbf{v}_i = \lambda_i \mathbf{v}_i \implies MD^{-1}(D^{1/2}\mathbf{v}_i) = \lambda_i (D^{1/2}\mathbf{v}_i).$$

So, if  $\mathbf{v}_1, \dots, \mathbf{v}_n$  form an orthonormal basis of eigenvectors of  $D^{-1/2}MD^{-1/2}$ , then we obtain a basis (not necessarily orthonormal) of eigenvectors of  $MD^{-1}$  by multiplying these vectors by  $D^{1/2}$ . Moreover, these matrices have the same eigenvalues.

The matrices we associate with directed graphs will not necessarily be diagonalizable.

---

<sup>1</sup>Strictly speaking, we are considering simple graphs. These are the graphs in which all edges go between distinct vertices and in which there can be at most one edge between a given pair of vertices. Graphs that have multiple-edges or self-loops are often called multi-graphs.

### 16.3 The matrices associated with a graph

Many different matrices arise in the field of Spectral Graph Theory. In this section we introduce the most prominent.

#### 16.3.1 Operators on the vertices

Eigenvalues and eigenvectors are used to understand what happens when one repeatedly applies an operator to a vector. If  $A$  is an  $n$ -by- $n$  matrix having a basis of right-eigenvectors  $\mathbf{v}_1, \dots, \mathbf{v}_n$  with

$$A\mathbf{v}_i = \lambda_i \mathbf{v}_i,$$

then we can use these eigenvectors to understand the impact of multiplying a vector  $\mathbf{x}$  by  $A$ . We first express  $\mathbf{x}$  in the eigenbasis

$$\mathbf{x} = \sum_i c_i \mathbf{v}_i$$

and then compute

$$A^k \mathbf{x} = \sum_i c_i A^k \mathbf{v}_i = \sum_i c_i \lambda_i^k \mathbf{v}_i.$$

If we have an operator that is naturally associated with a graph  $G$ , then properties of this operator, and therefore of the graph, will be revealed by its eigenvalues and eigenvectors. The first operator one typically associates with a graph  $G$  is its *adjacency operator*, realized by its *adjacency matrix*  $A_G$  and defined by

$$A_G(i, j) = \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

To understand spectral graph theory, one must view vectors  $\mathbf{x} \in \mathbb{R}^n$  as functions from the vertices to the Reals. That is, they should be understood as vectors in  $\mathbb{R}^V$ . When we apply the adjacency operator to such a function, the resulting value at a vertex  $a$  is the sum of the values of the function  $\mathbf{x}$  over all neighbors  $b$  of  $a$ :

$$(A_G \mathbf{x})(a) = \sum_{b:(a,b) \in E} \mathbf{x}(b).$$

This is very close to one of the most natural operators on a graph: the *diffusion operator*. Intuitively, the diffusion operator represents a process in which “stuff” or “mass” moves from vertices to their neighbors. As mass should be conserved, the mass at a given vertex is distributed evenly among

its neighbors. Formally, we define the *degree* of a vertex  $a$  to be the number of edges in which it participates. We naturally encode this in a vector, labeled  $\mathbf{d}$ :

$$\mathbf{d}(a) = |\{b : (a, b) \in E\}|,$$

where we write  $|S|$  to indicate the number of elements in a set  $S$ . We then define the *degree matrix*  $D_G$  by

$$D_G(a, b) = \begin{cases} \mathbf{d}(a) & \text{if } a = b \\ 0 & \text{otherwise.} \end{cases}$$

The *diffusion matrix* of  $G$ , also called the *walk matrix* of  $G$ , is then given by

$$W_G \stackrel{\text{def}}{=} A_G D_G^{-1}. \quad (16.1)$$

It acts on a vector  $\mathbf{x}$  by

$$(W_G \mathbf{x})(a) = \sum_{b:(a,b) \in E} \mathbf{x}(b) / \mathbf{d}(b).$$

This matrix is called the *walk matrix* of  $G$  because it encodes the dynamics of a random walk on  $G$ . Recall that a random walk is a process that begins at some vertex, then moves to a random neighbor of that vertex, and then a random neighbor of that vertex, and so on. The walk matrix is used to study the evolution of the probability distribution of a random walk. If  $\mathbf{p} \in \mathbb{R}^n$  is a probability distribution on the vertices, then  $W_G \mathbf{p}$  is the probability distribution obtained by selecting a vertex according to  $\mathbf{p}$ , and then selecting a random neighbor of that vertex. As the eigenvalues and eigenvectors of  $W_G$  provide information about the behavior of a random walk on  $G$ , they also provide information about the graph.

Of course, adjacency and walk matrices can also be defined for weighted graphs  $G = (V, E, w)$ . For a weighted graph  $G$ , we define

$$A_G(a, b) = \begin{cases} w(a, b) & \text{if } (a, b) \in E \\ 0 & \text{otherwise.} \end{cases}$$

When dealing with weighted graphs, we distinguish between the *weighted degree* of a vertex, which is defined to be the sum of the weights of its attached edges, and the *combinatorial degree* of a vertex, which is the number of such edges. We reserve the vector  $\mathbf{d}$  for the weighted degree, so

$$\mathbf{d}(a) = \sum_{b:(a,b) \in E} w(a, b).$$

The random walk on a weighted graph moves from a vertex  $a$  to a neighbor  $b$  with probability proportional to  $w(a, b)$ , so we still define its walk matrix by equation (16.1).

### 16.3.2 The Laplacian Quadratic Form

Matrices and spectral theory also arise in the study of quadratic forms. The most natural quadratic form to associate with a graph is the *Laplacian*, which is given by

$$\mathbf{x}^T L_G \mathbf{x} = \sum_{(a,b) \in E} w(a,b)(\mathbf{x}(a) - \mathbf{x}(b))^2. \quad (16.2)$$

This form measures the smoothness of the function  $\mathbf{x}$ . It will be small if the function  $\mathbf{x}$  does not jump too much over any edge. The matrix defining this form is the *Laplacian matrix* of the graph  $G$ ,

$$L_G \stackrel{\text{def}}{=} D_G - A_G.$$

The Laplacian matrices of weighted graphs arise in many applications. For example, they appear when applying the certain discretization schemes to solve Laplace's equation with Neumann boundary conditions. They also arise when modeling networks of springs or resistors. As resistor networks provide a very useful physical model for graphs, we explain the analogy in more detail. We associate an edge of weight  $w$  with a resistor of resistance  $1/w$ , since higher weight corresponds to higher connectivity which corresponds to less resistance.

When we inject and withdraw current from a network of resistors, we let  $\mathbf{i}_{ext}(a)$  denote the amount of current we inject into node  $a$ . If this quantity is negative then we are removing current. As electrical flow is a potential flow, there is a vector  $\mathbf{v} \in \mathbb{R}^V$  so that the amount of current that flows across edge  $(a, b)$  is

$$\mathbf{i}(a, b) = (\mathbf{v}(a) - \mathbf{v}(b)) / r(a, b),$$

where  $r(a, b)$  is the resistance of edge  $(a, b)$ . The Laplacian matrix provides a system of linear equations that may be used to solve for  $\mathbf{v}$  when given  $\mathbf{i}_{ext}$ :

$$\mathbf{i}_{ext} = L_G \mathbf{v}. \quad (16.3)$$

We refer the reader to [1] or [2] for more information about the connections between resistor networks and graphs.

### 16.3.3 The Normalized Laplacian

When studying random walks on a graph, it often proves useful to normalize the Laplacian by its degrees. The *normalized Laplacian* of  $G$  is defined by

$$N_G = D_G^{-1/2} L_G D_G^{-1/2} = I - D_G^{-1/2} A_G D_G^{-1/2}.$$

It should be clear that normalized Laplacian is closely related to the walk matrix of a graph. Chung's monograph on spectral graph theory focuses on the normalized Laplacian [3].

### 16.3.4 Naming the Eigenvalues

When the graph  $G$  is understood, we will always let

$$\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_n$$

denote the eigenvalues of the adjacency matrix. We order the eigenvalues of the Laplacian in the other direction:

$$0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n.$$

We will always let

$$0 = \nu_1 \leq \nu_2 \leq \cdots \leq \nu_n$$

denote the eigenvalues of the normalized Laplacian. Even though  $\omega$  is not a Greek variant of  $w$ , we use

$$1 = \omega_1 \geq \omega_2 \geq \cdots \geq \omega_n$$

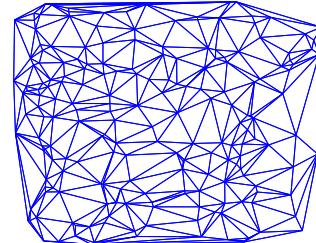
to denote the eigenvalues of the walk matrix. It is easy to show that  $\omega_i = 1 - \nu_i$ .

For graphs in which every vertex has the same weighted degree the degree matrix is a multiple of the identity; so,  $A_G$  and  $L_G$  have the same eigenvectors. For graphs that are not regular, the eigenvectors of  $A_G$  and  $L_G$  can behave very differently.

## 16.4 Some Examples

The most striking demonstration of the descriptive power of the eigenvectors of a graph comes from Hall's spectral approach to graph drawing [4]. To begin a demonstration of Hall's method, we generate the Delaunay graph of 200 randomly chosen points in the unit square.

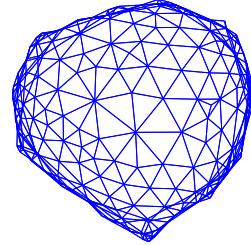
```
xy = rand(200,2);
tri = delaunay(xy(:,1),xy(:,2));
elem = ones(3)-eye(3);
for i = 1:length(tri),
    A(tri(i,:),tri(i,:)) = elem;
end
A = double(A > 0);
gplot(A,xy)
```



We will now discard the information we had about the coordinates of the vertices, and draw a picture of the graph using only the eigenvectors of its Laplacian matrix. We first compute the adjacency matrix  $A$ , the degree matrix  $D$ , and the Laplacian matrix  $L$  of the graph. We then compute the

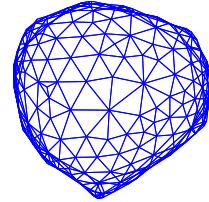
eigenvectors of the second and third smallest eigenvalues of  $L$ ,  $\mathbf{v}_2$  and  $\mathbf{v}_3$ . We then draw the same graph, using  $\mathbf{v}_2$  and  $\mathbf{v}_3$  to provide the coordinates of vertices. That is, we locate vertex  $a$  at position  $(\mathbf{v}_2(a), \mathbf{v}_3(a))$ , and draw the edges as straight lines between the vertices.

```
D = diag(sum(A));
L = D - A;
[v,e] = eigs(L, 3, 'sm');
gplot(A,v(:,[2 1]))
```



Amazingly, this process produces a very nice picture of the graph, in spite of the fact that the coordinates of the vertices were generated solely from the combinatorial structure of the graph. Note that the interior is almost planar. We could have obtained a similar, and possibly better, picture from the left-eigenvectors of the walk matrix of the graph.

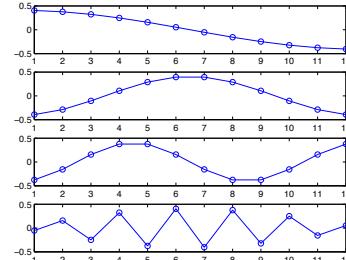
```
W = A * inv(D);
[v,e] = eigs(W', 3);
gplot(A,v(:,[2 3]));
```



We defer the motivation for Hall's graph drawing technique to Section 16.7, so that we may first explore other examples.

One of the simplest graphs is the path graph. In the following figure, we plot the 2nd, 3rd, 4th, and 12th eigenvectors of the Laplacian of the path graph on 12 vertices. In each plot, the  $x$ -axis is the number of the vertex, and the  $y$ -axis is the value of the eigenvector at that vertex. We do not bother to plot the 1st eigenvector, as it is a constant vector.

```
A = diag(ones(1,11),1);
A = A + A';
D = diag(sum(A));
L = D - A;
[v,e] = eig(L);
plot(v(:,2), 'o'); hold on;
plot(v(:,2));
plot(v(:,3), 'o'); hold on;
plot(v(:,3));
```

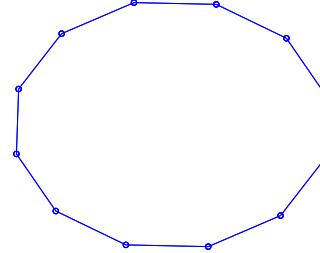


Observe that the 2nd eigenvector is monotonic along the path, that the second changes sign twice, and that the 12th alternates negative and positive. This can be explained by viewing these eigenvectors as the fundamental modes

of vibration of a discretization of a string. We recommend [5] for a formal treatment.

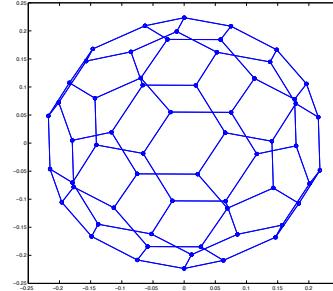
By now, the reader should not be surprised to see that ring graphs have the obvious spectral drawings. In this case, we obtain the ring from the path by adding an edge between vertex 1 and 12.

```
A(1,12) = 1; A(12,1) = 1;
D = diag(sum(A));
L = D - A;
[v,e] = eig(L);
gplot(A,v(:,[2 3]))
hold on
gplot(A,v(:,[2 3]),'o')
```



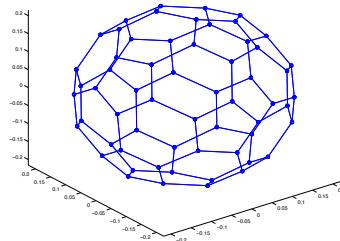
Our last example comes from the skeleton of the ‘‘Buckyball’’. This is the same as the graph between the corners of the Buckminster Fuller geodesic dome and of the seams on a standard Soccer ball.

```
A = full(bucky);
D = diag(sum(A));
L = D - A;
[v,e] = eig(L);
gplot(A,v(:,[2 3]))
hold on;
gplot(A,v(:,[2 3]),'o')
```



Note that the picture looks like a squashed Buckyball. The reason is that there is no canonical way to choose the eigenvectors  $v_2$  and  $v_3$ . The smallest non-zero eigenvalue of the Laplacian has multiplicity three. This graph should really be drawn in three dimensions, using any set of orthonormal vectors  $v_2, v_3, v_4$  of the smallest non-zero eigenvalue of the Laplacian. As this picture hopefully shows, we obtain the standard embedding of the Buckyball in  $\mathbb{R}^3$ .

```
[x,y] = gplot(A,v(:,[2 3]));
[x,z] = gplot(A,v(:,[2 4]));
plot3(x,y,z)
```



The Platonic solids and all vertex-transitive convex polytopes in  $\mathbb{R}^d$  display similar behavior. We refer the reader interested in learning more about this phenomenon to either Godsil's book [6] or to [7].

## 16.5 The Role of the Courant-Fischer Theorem

Recall that the Rayleigh quotient of a non-zero vector  $\mathbf{x}$  with respect to a symmetric matrix  $A$  is

$$\frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

The Courant-Fischer characterization of the eigenvalues of a symmetric matrix  $A$  in terms of the maximizers and minimizers of the Rayleigh quotient (see [8]) plays a fundamental role in spectral graph theory.

**Theorem 3 (Courant-Fischer)** *Let  $A$  be a symmetric matrix with eigenvalues  $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$ . Then,*

$$\alpha_k = \max_{\substack{S \subseteq \mathbb{R}^n \\ \dim(S)=k}} \min_{\substack{\mathbf{x} \in S \\ \mathbf{x} \neq 0}} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \min_{\substack{T \subseteq \mathbb{R}^n \\ \dim(T)=n-k+1}} \max_{\substack{\mathbf{x} \in T \\ \mathbf{x} \neq 0}} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

The maximum in the first expression is taken over all subspaces of dimension  $k$ , and the minimum in the second is over all subspaces of dimension  $n-k+1$ .

Henceforth, whenever we minimize or maximize Rayleigh quotients we will only consider non-zero vectors, and thus will drop the quantifier " $\mathbf{x} \neq 0$ ".

For example, the Courant-Fischer Theorem tells us that

$$\alpha_1 = \max_{\mathbf{x} \in \mathbb{R}^n} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \quad \text{and} \quad \alpha_n = \min_{\mathbf{x} \in \mathbb{R}^n} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}.$$

We recall that a symmetric matrix  $A$  is *positive semidefinite*, written  $A \succcurlyeq 0$ , if all of its eigenvalues are non-negative. From (16.2) we see that the Laplacian is positive semidefinite. Adjacency matrices and walk matrices of non-empty graphs are not positive semidefinite as the sum of their eigenvalues equals their trace, which is 0. For this reason, one often considers the *lazy random walk* on a graph instead of the ordinary random walk. This walk stays put at each step with probability 1/2. This means that the corresponding matrix is  $(1/2)I + (1/2)W_G$ , which can be shown to positive semidefinite.

### 16.5.1 Low-Rank Approximations

One explanation for the utility of the eigenvectors of extreme eigenvalues of matrices is that they provide low-rank approximations of a matrix. Recall

that if  $A$  is a symmetric matrix with eigenvalues  $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$  and a corresponding orthonormal basis of column eigenvectors  $\mathbf{v}_1, \dots, \mathbf{v}_n$ , then

$$A = \sum_i \alpha_i \mathbf{v}_i \mathbf{v}_i^T.$$

We can measure how well a matrix  $B$  approximates a matrix  $A$  by either the operator norm  $\|A - B\|$  or the Frobenius norm  $\|A - B\|_F$ , where we recall

$$\|M\| \stackrel{\text{def}}{=} \max_{\mathbf{x}} \frac{\|M\mathbf{x}\|}{\|\mathbf{x}\|} \quad \text{and} \quad \|M\|_F \stackrel{\text{def}}{=} \sqrt{\sum_{i,j} M(i,j)^2}.$$

Using the Courant-Fischer Theorem, one can prove that for every  $k$ , the best approximation of  $A$  by a rank- $k$  matrix is given by summing the terms  $\alpha_i \mathbf{v}_i \mathbf{v}_i^T$  over the  $k$  values of  $i$  for which  $|\alpha_i|$  is largest. This holds regardless of whether we measure the quality of approximation in the operator or Frobenius norm.

When the difference between  $A$  and its best rank- $k$  approximation is small, it explains why the eigenvectors of the largest  $k$  eigenvalues of  $A$  should provide a lot of information about  $A$ . However, one must be careful when applying this intuition as the analogous eigenvectors of the Laplacian correspond to its smallest eigenvalues. Perhaps the best way to explain the utility of these small eigenvectors is to observe that they provide the best low-rank approximation of the pseudoinverse of the Laplacian.

## 16.6 Elementary Facts

We list some elementary facts about the extreme eigenvalues of the Laplacian and adjacency matrices. We recommend deriving proofs yourself, or consulting the suggested references.

1. The all-1s vector is always an eigenvector of  $L_G$  of eigenvalue 0.
2. The largest eigenvalue of the adjacency matrix is at least the average degree of a vertex of  $G$  and at most the maximum degree of a vertex of  $G$  (see [9] or [10, Section 3.2]).
3. If  $G$  is connected, then  $\alpha_1 > \alpha_2$  and the eigenvector of  $\alpha_1$  may be taken to be positive (this follows from the Perron-Frobenius theory; see [11]).
4. The all-1s vector is an eigenvector of  $A_G$  with eigenvalue  $\alpha_1$  if and only if  $G$  is an  $\alpha_1$ -regular graph.
5. The multiplicity of 0 as an eigenvalue of  $L_G$  is equal to the number of connected components of  $L_G$ .

6. The largest eigenvalue of  $L_G$  is at most twice the maximum degree of a vertex in  $G$ .
  7.  $\alpha_n = -\alpha_1$  if and only if  $G$  is bipartite (see [12], or [10, Theorem 3.4]).
- 

## 16.7 Spectral Graph Drawing

We can now explain the motivation behind Hall's spectral graph drawing technique [4]. Hall first considered the problem of assigning a real number  $\mathbf{x}(a)$  to each vertex  $a$  so that  $(\mathbf{x}(a) - \mathbf{x}(b))^2$  is small for most edges  $(a, b)$ . This led him to consider the problem of minimizing (16.2). So as to avoid the degenerate solutions in which every vertex is mapped to zero, or any other value, he introduces the restriction that  $\mathbf{x}$  be orthogonal to  $\mathbf{b}_1$ . As the utility of the embedding does not really depend upon its scale, he suggested the normalization  $\|\mathbf{x}\| = 1$ . By the Courant-Fischer Theorem, the solution to the resulting optimization problem is precisely an eigenvector of the second-smallest eigenvalue of the Laplacian.

But, what if we want to assign the vertices to points in  $\mathbb{R}^2$ ? The natural minimization problem,

$$\min_{\mathbf{x}, \mathbf{y} \in \mathbb{R}^V} \sum_{(a, b) \in E} \|(\mathbf{x}(a), \mathbf{y}(a)) - (\mathbf{x}(b), \mathbf{y}(b))\|^2$$

such that

$$\sum_a (\mathbf{x}(a), \mathbf{y}(a)) = (0, 0)$$

typically results in the degenerate solution  $\mathbf{x} = \mathbf{y} = \mathbf{v}_2$ . To ensure that the two coordinates are different, Hall introduced the restriction that  $\mathbf{x}$  be orthogonal to  $\mathbf{y}$ . One can use the Courant-Fischer Theorem to show that the optimal solution is then given by setting  $\mathbf{x} = \mathbf{v}_2$  and  $\mathbf{y} = \mathbf{v}_3$ , or by taking a rotation of this solution.

Hall observes that this embedding seems to cluster vertices that are close in the graph, and separate vertices that are far in the graph. For more sophisticated approaches to drawing graphs, we refer the reader to Chapter 15.

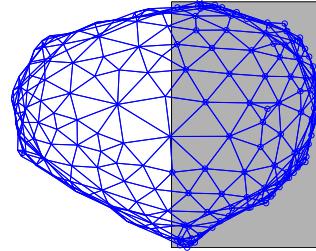
---

## 16.8 Algebraic Connectivity and Graph Partitioning

Many useful ideas in spectral graph theory have arisen from efforts to find quantitative analogs of qualitative statements. For example, it is easy to show

that  $\lambda_2 > 0$  if and only if  $G$  is connected. This led Fiedler [13] to label  $\lambda_2$  the *algebraic connectivity* of a graph, and to prove in various ways that better connected graphs have higher values of  $\lambda_2$ . This also led Fiedler to consider dividing the nodes of a graph into two pieces by choosing a real number  $t$ , and partitioning the nodes depending on whether or not  $v_2(a) \geq t$ . For  $t = 0$ , this corresponds to selecting all vertices in the right-half of the spectral embedding of the graph.

```
S = find(v(:,2) >= 0);
plot(v(S,2),v(S,1),'o')
```



Fiedler proved [14] that for all  $t \leq 0$ , the set of nodes  $a$  for which  $v_2(a) \geq t$  forms a connected component. This type of “nodal domain theorem” was extended by van der Holst [15] to the set of  $a$  such that  $v(a) > 0$ , when  $v$  is an eigenvector of  $\lambda_2$  of minimal support.

The use of graph eigenvectors to partition graphs was also pioneered by Donath and Hoffman [16, 17] and Barnes [18]. It was popularized by experimental studies showing that it could give very good results [19, 20, 21, 22].

In many applications, one wants to partition the nodes of a graph into a few pieces of roughly equal size without removing too many edges (see Chapters 10 and 13). For simplicity, consider the problem of dividing the vertices of a graph into two pieces. In this case, we need merely identify one piece  $S \subset V$ . We then define  $\partial(S)$  to be the set of edges with exactly one endpoint in  $S$ . We will also refer to  $S$  as a *cut*, as it implicitly divides the vertices into  $S$  and  $V - S$ , cutting all edges in  $\partial(S)$ . A tradeoff between the number of edges cut and the balance of the partition is obtained by dividing the first by a measure of the second, resulting in quantities called *cut ratio*, *sparsity*, *isoperimetric number*, and *conductance*, although these terms are sometimes used interchangeably. Wei and Cheng [23] suggested measuring the *ratio* of a cut, which they defined to be

$$R(S) \stackrel{\text{def}}{=} \frac{|\partial(S)|}{|S||V - S|}.$$

Hagen and Kahng [24] observe that this quantity is always at least  $\lambda_2/n$ , and that  $v_2$  can be described as a relaxation of the characteristic vector<sup>2</sup> of the set  $S$  that minimizes  $R(S)$ .

Let  $\chi_S$  be the characteristic vector of a set  $S$ . For an unweighted graph  $G$

---

<sup>2</sup>Here, we define the characteristic vector of a set to be the vector that is one at vertices inside the set and zero elsewhere.

we have

$$\chi_S^T L_G \chi_S = |\partial(S)|,$$

and

$$\sum_{a < b} (\chi_S(a) - \chi_S(b))^2 = |S| |V - S|.$$

So,

$$R(S) = \frac{\chi_S^T L_G \chi_S}{\sum_{a < b} (\chi_S(a) - \chi_S(b))^2}.$$

On the other hand, Fiedler [14] proved that

$$\lambda_2 = n \min_{\mathbf{x} \neq \mathbf{0}} \frac{\mathbf{x}^T L_G \mathbf{x}}{\sum_{a < b} (\mathbf{x}(a) - \mathbf{x}(b))^2}.$$

If we impose the restriction that  $\mathbf{x}$  be a zero-one valued vector and then minimize this last expression, we obtain the characteristic vector of the set of minimum ratio. As we have imposed a constraint on the vector  $\mathbf{x}$ , the minimum ratio obtained must be larger than  $\lambda_2$ . Hagen and Kahng make this observation, and suggest using  $\mathbf{v}_2$  to try to find a set of low ratio by choosing some value  $t$ , and setting  $S = \{a : \mathbf{v}_2(a) \geq t\}$ .

One may actually prove that the set obtained in this fashion does not have ratio too much worse than the minimum. Statements of this form follow from discrete versions of Cheeger's inequality [25]. The cleanest version relates to the the *conductance* of a set  $S$

$$\phi(S) \stackrel{\text{def}}{=} \frac{w(\partial(S))}{\min(\mathbf{d}(S), \mathbf{d}(V - S))},$$

where  $\mathbf{d}(S)$  denotes the sum of the degrees of the vertices in  $S$  and  $w(\partial(S))$  denotes the sum of the weights of the edges in  $\partial(S)$ . The conductance of the graph  $G$  is defined by

$$\phi_G = \min_{\emptyset \subset S \subset V} \phi(S).$$

By a similar relaxation argument, one can show

$$2\phi_G \geq \nu_2.$$

Sinclair and Jerrum's discrete version of Cheeger's inequality [26] says that

$$\nu_2 \leq \phi_G^2 / 2.$$

Moreover, their proof reveals that if  $\mathbf{v}_2$  is an eigenvector of  $\nu_2$ , then there exists a  $t$  so that

$$\phi \left( \left\{ a : \mathbf{d}^{-1/2}(a) \mathbf{v}_2(a) \geq t \right\} \right) \leq \sqrt{2\nu_2}.$$

Other discretizations of Cheeger's inequality were proved around the same

time by a number of researchers. See [27, 28, 29, 30, 31]. We remark that Lawler and Sokal define conductance by

$$\frac{w(\partial(S))}{\mathbf{d}(S)\mathbf{d}(V-S)},$$

which is proportional to the *normalized cut* measure

$$\frac{w(\partial(S))}{\mathbf{d}(S)} + \frac{w(\partial(V-S))}{\mathbf{d}(V-S)}$$

popularized by Shi and Malik [21]. The advantage of this later formulation is that it has an obvious generalization to partitions into more than two pieces.

In general, the eigenvalues and entries of eigenvectors of Laplacian matrices will not be rational numbers; so, it is unreasonable to hope to compute them exactly. Mihail [32] proves that an approximation of the second-smallest eigenvector suffices. While her argument was stated for regular graphs, one can apply it to irregular, weighted graphs to show that for every vector  $\mathbf{x}$  orthogonal to  $\mathbf{d}^{1/2}$  there exists a  $t$  so that

$$\phi\left(\left\{a : \mathbf{d}^{-1/2}(a)\mathbf{x}(a) \geq t\right\}\right) \leq \sqrt{2\frac{\mathbf{x}^T N_G \mathbf{x}}{\mathbf{x}^T \mathbf{x}}}.$$

While spectral partitioning heuristics are easy to implement, they are neither the most effective in practice or in theory. Theoretically better algorithms have been obtained by linear programming [33] and by semi-definite programming [34]. Fast variants of these algorithms may be found in [35, 36, 37, 38, 39]. More practical algorithms are discussed in Chapters 10 and 13.

### 16.8.1 Convergence of Random Walks

If  $G$  is a connected, undirected graph, then the largest eigenvalue of  $W_G$ ,  $\omega_1$ , has multiplicity 1, equals 1, and has eigenvector  $\mathbf{d}$ . We may convert this eigenvector into a probability distribution  $\boldsymbol{\pi}$  by setting

$$\boldsymbol{\pi} = \frac{\mathbf{d}}{\sum_a \mathbf{d}(a)}.$$

If  $\omega_n \neq -1$ , then the distribution of every random walk eventually converges to  $\boldsymbol{\pi}$ . The rate of this convergence is governed by how close  $\max(\omega_2, -\omega_n)$  is to  $\omega_1$ . For example, let  $\mathbf{p}_t$  denote the distribution after  $t$  steps of a random walk that starts at vertex  $a$ . Then for every vertex  $b$ ,

$$|\mathbf{p}_t(b) - \boldsymbol{\pi}(b)| \leq \sqrt{\frac{\mathbf{d}(b)}{\mathbf{d}(a)}} (1 - \max(\omega_2, -\omega_n))^t.$$

One intuition behind Cheeger's inequality is that sets of small conductance are precisely the obstacles to the convergence of random walks.

For more information about random walks on graphs, we recommend the survey of Lovàsz [40] and the book by Doyle and Snell [2].

### 16.8.2 Expander Graphs

Some of the most fascinating graphs are those on which random walks mix quickly and which have high conductance. These are called *expander graphs*, and may be defined as the  $d$ -regular graphs for which all non-zero Laplacian eigenvalues are bounded away from zero. In the better expander graphs, all the Laplacian eigenvalues are close to  $d$ . One typically considers infinite families of such graphs in which  $d$  and a lower bound on the distance of the non-zero eigenvalues from  $d$  remain constant. These are counter-examples to many naive conjectures about graphs, and should be kept in mind whenever one is thinking about graphs. They have many amazing properties, and have been used throughout Theoretical Computer Science. In addition to playing a prominent role in countless theorems, they are used in the design of pseudo-random generators [41, 42, 43], error-correcting codes [44, 45, 46, 47, 48], fault-tolerant circuits [49] and routing networks [50].

The reason such graphs are called *expanders* is that all small sets of vertices in these graphs have unusually large numbers of neighbors. That is, their neighborhoods expand. For  $S \subset V$ , let  $N(S)$  denote the set of vertices that are neighbors of vertices in  $S$ . Tanner [51] provides a lower bound on the size of  $N(S)$  in bipartite graphs. In general graphs, it becomes the following.

**Theorem 4** *Let  $G = (V, E)$  be a  $d$ -regular graph on  $n$  vertices and set*

$$\epsilon = \max \left( 1 - \frac{\lambda_2}{d}, \frac{\lambda_n}{d} - 1 \right)$$

*Then, for all  $S \subseteq V$ ,*

$$|N(S)| \geq \frac{|S|}{\epsilon^2(1 - \alpha) + \alpha},$$

*where  $|S| = \alpha n$ .*

The term  $\epsilon$  is small when all of the eigenvalues are close to  $d$ . Note that when  $\alpha$  is much less than  $\epsilon^2$ , the term on the right is approximately  $|S|/\epsilon^2$ , which can be much larger than  $|S|$ .

An example of the pseudo-random properties of expander graphs is the “Expander Mixing Lemma”. To understand it, consider choosing two subsets of the vertices,  $S$  and  $T$  of sizes  $\alpha n$  and  $\beta n$ , at random. Let  $\vec{E}(S, T)$  denote the set of ordered pairs  $(a, b)$  with  $a \in S$ ,  $b \in T$  and  $(a, b) \in E$ . The expected size of  $\vec{E}(S, T)$  is  $\alpha\beta dn$ . This theorem tells us that for *every* pair of large sets  $S$  and  $T$ , the number of such pairs is approximately this quantity. Alternatively, one may view an expander as an approximation of the complete graph. The fraction of edges in the complete graph going from  $S$  to  $T$  is  $\alpha\beta$ . The following theorem says that the same is approximately true for all sufficiently large sets  $S$  and  $T$ .

**Theorem 5 (Expander Mixing Lemma)** *Let  $G = (V, E)$  be a  $d$ -regular*

graph and set

$$\epsilon = \max \left( 1 - \frac{\lambda_2}{d}, \frac{\lambda_n}{d} - 1 \right)$$

Then, for every  $S \subseteq V$  and  $T \subseteq V$ ,

$$\left| |\vec{E}(S, T)| - \alpha\beta dn \right| \leq \epsilon dn \sqrt{(\alpha - \alpha^2)(\beta - \beta^2)},$$

where  $|S| = \alpha n$  and  $|T| = \beta n$ .

This bound is a slight extension by Beigel, Margulis and Spielman [52] of a bound originally proved by Alon and Chung [53]. Observe that when  $\alpha$  and  $\beta$  are greater than  $\epsilon$ , the term on the right is less than  $\alpha\beta dn$ . Theorem 4 may be derived from Theorem 5.

We refer readers who would like to learn more about expander graphs to the survey of Hoory, Linial and Wigderson [54].

### 16.8.3 Ramanujan Graphs

Given the importance of  $\lambda_2$ , we should know how close it can be to  $d$ . Nilli [55] shows that it cannot be much closer than  $2\sqrt{d-1}$ .

**Theorem 6** Let  $G$  be an unweighted  $d$ -regular graph containing two edges  $(u_0, u_1)$  and  $(v_0, v_1)$  whose vertices are at distance at least  $2k+2$  from each other. Then

$$\lambda_2 \leq d - 2\sqrt{d-1} + \frac{2\sqrt{d-1}-1}{k+1}.$$

Amazingly, Margulis [56] and Lubotzky, Phillips and Sarnak [57] have constructed infinite families of  $d$ -regular graphs, called Ramanujan graphs, for which  $\lambda_2 \geq d - 2\sqrt{d-1}$ .

However, this is not the end of the story. Kahale [58] proves that vertex expansion by a factor greater than  $d/2$  cannot be derived from bounds on  $\lambda_2$ . Expander graphs that have expansion greater than  $d/2$  on small sets of vertices have been derived by Capalbo *et. al.* [59] through non-spectral arguments.

### 16.8.4 Bounding $\lambda_2$

I consider  $\lambda_2$  to be the most interesting parameter of a connected graph. If it is large, the graph is an expander. If it is small, then the graph can be cut into two pieces without removing too many edges. Either way, we learn something about the graph. Thus, it is very interesting to find ways of estimating the value of  $\lambda_2$  for families of graphs.

One way to explain the success of spectral partitioning heuristics is to prove that the graphs to which they are applied have small values of  $\lambda_2$  or  $\nu_2$ . A line of work in this direction was started by Spielman and Teng [60], who proved upper bounds on  $\lambda_2$  for planar graphs and well-shaped finite element meshes.

**Theorem 7 ([60])** *Let  $G$  be a planar graph with  $n$  vertices of maximum degree  $d$ , and let  $\lambda_2$  be the second-smallest eigenvalue of its Laplacian. Then,*

$$\lambda_2 \leq \frac{8d}{n}.$$

This theorem has been extended to graphs of bounded genus by Kelner [61]. Entirely new techniques were developed by Biswal, Lee and Rao [62] to extend this bound to graphs excluding bounded minors. Bounds on higher Laplacian eigenvalues have been obtained by Kelner, Lee, Price and Teng [63].

**Theorem 8 ([63])** *Let  $G$  be a graph with  $n$  vertices and constant maximum degree. If  $G$  is planar, has constant genus, or has a constant-sized forbidden minor, then*

$$\lambda_k \leq O(k/n).$$

Proving lower bounds on  $\lambda_2$  is a more difficult problem. The dominant approach is to relate the graph under consideration to a graph with known eigenvalues, such as the complete graph. Write

$$L_G \succcurlyeq cL_H$$

if  $L_G - cL_H \succcurlyeq 0$ . In this case, we know that

$$\lambda_i(G) \geq c\lambda_i(H),$$

for all  $i$ . Inequalities of this form may be proved by identifying each edge of the graph  $H$  with a path in  $G$ . The resulting bounds are called Poincaré inequalities, and are closely related to the bounds used in the analysis of preconditioners in Chapter 12 and in related works [64, 65, 66, 67]. For examples of such arguments, we refer the reader to one of [68, 69, 70].

## 16.9 Coloring and Independent Sets

In the graph coloring problem one is asked to assign a color to every vertex of a graph so that every edge connects vertices of different colors, while using as few colors as possible. Replacing colors with numbers, we define a  $k$ -coloring of a graph  $G = (V, E)$  to be a function  $c : V \rightarrow \{1, \dots, k\}$  such that

$$c(i) \neq c(j), \text{ for all } (i, j) \in E.$$

The *chromatic number* of a graph  $G$ , written  $\chi(G)$ , is the least  $k$  for which  $G$  has a  $k$ -coloring. Wilf [71] proved that the chromatic number of a graph may be bounded above by its largest adjacency eigenvalue.

**Theorem 9 ([71])**

$$\chi(G) \leq \alpha_1 + 1.$$

On the other hand, Hoffman [72] proved a lower bound on the chromatic number in terms of the adjacency matrix eigenvalues. When reading this theorem, recall that  $\alpha_n$  is negative.

**Theorem 10** *If  $G$  is a graph with at least one edge, then*

$$\chi(G) \geq \frac{\alpha_1 - \alpha_n}{-\alpha_n} = 1 + \frac{\alpha_1}{-\alpha_n}.$$

In fact, this theorem holds for arbitrary weighted graphs. Thus, one may prove lower bounds on the chromatic number of a graph by assigning a weight to every edge, and then computing the resulting ratio.

It follows from Theorem 10 that  $G$  is not bipartite if  $|\alpha_n| < \alpha_1$ . Moreover, as  $|\alpha_n|$  becomes closer to 0, more colors are needed to properly color the graph. Another way to argue that graphs with small  $|\alpha_n|$  are far from being bipartite was found by Trevisan [73]. To be precise, Trevisan proves a bound, analogous to Cheeger's inequality, relating  $|E| - \max_{S \subset V} |\partial(S)|$  to the smallest eigenvalue of the *signless Laplacian matrix*,  $D_G + A_G$ .

An *independent set* of vertices in a graph  $G$  is a set  $S \subseteq V$  such that no edge connects two vertices of  $S$ . The size of the largest independent set in a graph is called its *independence number*, and is denoted  $\alpha(G)$ . As all the nodes of one color in a coloring of  $G$  are independent, we know

$$\alpha(G) \geq n/\chi(G).$$

For regular graphs, Hoffman derived the following upper bound on the size of an independent set.

**Theorem 11** *Let  $G = (V, E)$  be a  $d$ -regular graph. Then*

$$\alpha(G) \leq n \frac{-\alpha_n}{d - \alpha_n}.$$

This implies Theorem 10 for regular graphs.

## 16.10 Perturbation Theory and Random Graphs

McSherry [74] observes that the spectral partitioning heuristics and the related spectral heuristics for graph coloring can be understood through matrix perturbation theory. For example, let  $G$  be a graph and let  $S$  be a subset

of the vertices of  $G$ . Without loss of generality, assume that  $S$  is the set of the first  $|S|$  vertices of  $G$ . Then, we can write the adjacency matrix of  $G$  as

$$\begin{bmatrix} A(S) & 0 \\ 0 & A(V - S) \end{bmatrix} + \begin{bmatrix} 0 & A(S, V - S) \\ A(V - S, S) & 0 \end{bmatrix},$$

where we write  $A(S)$  to denote the restriction of the adjacency matrix to the vertices in  $S$ , and  $A(S, V - S)$  to capture the entries in rows indexed by  $S$  and columns indexed by  $V - S$ . The set  $S$  can be discovered from an examination of the eigenvectors of the left-hand matrix: it has one eigenvector that is positive on  $S$  and zero elsewhere, and another that is positive on  $V - S$  and zero elsewhere. If the right-hand matrix is a “small” perturbation of the left-hand matrix, then we expect similar eigenvectors to exist in  $A$ . It seems reasonable that the right-hand matrix should be small if it contains few edges. Whether or not this may be made rigorous depends on the locations of the edges. We will explain McSherry’s analysis, which makes this rigorous in certain random models.

We first recall the basics perturbation theory for matrices. Let  $A$  and  $B$  be symmetric matrices with eigenvalues  $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$  and  $\beta_1 \geq \beta_2 \geq \dots \geq \beta_n$ , respectively. Let  $M = A - B$ . Weyl’s Theorem, which follows from the Courant-Fischer Theorem, tells us that

$$|\alpha_i - \beta_i| \leq \|M\|$$

for all  $i$ . As  $M$  is symmetric,  $\|M\|$  is merely the largest absolute value of an eigenvalue of  $M$ .

When some eigenvalue  $\alpha_i$  is well-separated from the others, one can show that a small perturbation does not change the corresponding eigenvector too much. Demmel [75, Theorem 5.2] proves the following bound.

**Theorem 12** *Let  $\mathbf{v}_1, \dots, \mathbf{v}_n$  be an orthonormal basis of eigenvectors of  $A$  corresponding to  $\alpha_1, \dots, \alpha_n$  and let  $\mathbf{u}_1, \dots, \mathbf{u}_n$  be an orthonormal basis of eigenvectors of  $B$  corresponding to  $\beta_1, \dots, \beta_n$ . Let  $\theta_i$  be the angle between  $\mathbf{v}_i$  and  $\mathbf{u}_i$ . Then,*

$$\frac{1}{2} \sin 2\theta_i \leq \frac{\|M\|}{\min_{j \neq i} |\alpha_j - \alpha_i|}.$$

McSherry applies these ideas from perturbation theory to analyze the behavior of spectral partitioning heuristics on random graphs that are generated to have good partitions. For example, he considered the planted partition model of Boppana [76]. This is defined by a weighted complete graph  $H$  determined by a  $S \subset V$  in which

$$w(a, b) = \begin{cases} p & \text{if both or neither of } a \text{ and } b \text{ are in } S, \text{ and} \\ q & \text{if exactly one of } a \text{ and } b \text{ are in } S, \end{cases}$$

for  $q < p$ . A random unweighted graph  $G$  is then constructed by including

edge  $(a, b)$  in  $G$  with probability  $w(a, b)$ . For appropriate values of  $q$  and  $p$ , the cut determined by  $S$  is very likely to be the sparsest. If  $q$  is not too close to  $p$ , then the largest two eigenvalues of  $H$  are far from the rest, and correspond to the all-1s vector and a vector that is uniform and positive on  $S$  and uniform and negative on  $V - S$ . Using results from random matrix theory of Füredi and Komlós [77], Vu [78], and Alon, Krievlevich and Vu [79], McSherry proves that  $G$  is a slight perturbation of  $H$ , and that the eigenvectors of  $G$  can be used to recover the set  $S$ , with high probability.

Both McSherry [74] and Alon and Kahale [80] have shown that the eigenvectors of the smallest adjacency matrix eigenvalues may be used to  $k$ -color randomly generated  $k$ -colorable graphs. These graphs are generated by first partitioning the vertices into  $k$  sets,  $S_1, \dots, S_k$ , and then adding edges between vertices in different sets with probability  $p$ , for some small  $p$ .

For more information on these and related results, we suggest the book by Kannan and Vempala [81].

### 16.11 Relative Spectral Graph Theory

Preconditioning (see Chapter 12) has inspired the study of the *relative eigenvalues of graphs*. These are the eigenvalues of  $L_G L_H^+$ , where  $L_G$  is the Laplacian of a graph  $G$  and  $L_H^+$  is the pseudo-inverse of the Laplacian of a graph  $H$ . We recall that the pseudo-inverse of a symmetric matrix  $L$  is given by

$$\sum_{i:\lambda_i \neq 0} \frac{1}{\lambda_i} \mathbf{v}_i \mathbf{v}_i^T,$$

where the  $\lambda_i$  and  $\mathbf{v}_i$  are the eigenvalues and eigenvectors of the matrix  $L$ . The eigenvalues of  $L_G L_H^+$  reveal how well  $H$  approximates  $G$ .

Let  $K_n$  denote the complete graph on  $n$  vertices. All of the non-trivial eigenvalues of the Laplacian of  $K_n$  equal  $n$ . So,  $L_{K_n}$  acts as  $n$  times the identity on the space orthogonal to  $\mathbf{b1}$ . Thus, for every  $G$  the eigenvalues of  $L_G L_{K_n}^+$  are just the eigenvalues of  $L_G$  divided by  $n$ , and the eigenvectors are the same. Many results on expander graphs, including those in Section 16.8.2, can be derived by using this perspective to treat an expander as an approximation of the complete graph (see [82]).

Recall that when  $L_G$  and  $L_H$  have the same range,  $\kappa_f(L_G, L_H)$  is defined to be the largest non-zero eigenvalue of  $L_G L_H^+$  divided by the smallest. The Ramanujan graphs are  $d$ -regular graphs  $G$  for which

$$\kappa_f(L_G, L_{K_n}) \leq \frac{d + 2\sqrt{d - 1}}{d - 2\sqrt{d - 1}}.$$

Batson, Spielman and Srivastava [82] prove that every graph  $H$  can be approximated by a sparse graph almost as well as this.

**Theorem 13** *For every weighted graph  $G$  on  $n$  vertices and every  $d > 1$ , there exists a weighted graph  $H$  with at most  $\lceil d(n-1) \rceil$  edges such that*

$$\kappa_f(L_G, L_H) \leq \frac{d+1+2\sqrt{d}}{d+1-2\sqrt{d}}.$$

Spielman and Srivastava [83] show that if one forms a graph  $H$  by sampling  $O(n \log n / \epsilon^2)$  edges of  $G$  with probability proportional to their effective resistance and rescaling their weights, then with high probability  $\kappa_f(L_G, L_H) \leq 1 + \epsilon$ .

Spielman and Woo [84] have found a characterization of the well-studied *stretch* of a spanning tree with respect to a graph in terms of relative graph spectra. For simplicity, we just define it for unweighted graphs. If  $T$  is a spanning tree of a graph  $G = (V, E)$ , then for every  $(a, b) \in E$  there is a unique path in  $T$  connecting  $a$  to  $b$ . The stretch of  $(a, b)$  with respect to  $T$ , written  $\text{st}_T(a, b)$ , is the number of edges in that path in  $T$ . The stretch of  $G$  with respect to  $T$  is then defined to be

$$\text{st}_T(G) \stackrel{\text{def}}{=} \sum_{(a,b) \in E} \text{st}_T(a, b).$$

**Theorem 14 ([84])**

$$\text{st}_T(G) = \text{trace}(L_G L_T^+).$$

See Chapter 12 for a proof.

## 16.12 Directed Graphs

There has been much less success in the study of the spectra of directed graphs, perhaps because the nonsymmetric matrices naturally associated with directed graphs are not necessarily diagonalizable. One naturally defines the adjacency matrix of a directed graph  $G$  by

$$A_G(a, b) = \begin{cases} 1 & \text{if } G \text{ has a directed edge from } b \text{ to } a \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, if we let  $\mathbf{d}(a)$  denote the number of edges leaving vertex  $a$  and define  $D$  as before, then the matrix realizing the random walk on  $G$  is

$$W_G = A_G D_G^{-1}.$$

The Perron-Frobenius Theorem (see [11, 8]) tells us that if  $G$  is strongly connected, then  $A_G$  has a unique positive eigenvector  $\mathbf{v}$  with a positive eigenvalue  $\lambda$  such that every other eigenvalue  $\mu$  of  $A$  satisfies  $|\mu| \leq \lambda$ . The same holds for  $W_G$ . When  $|\mu| < \lambda$  for all other eigenvalues  $\mu$ , this vector is proportional to the unique limiting distribution of the random walk on  $G$ .

These Perron-Frobenius eigenvectors have proved incredibly useful in a number of situations. For instance, they are at the heart of Google's PageRank algorithm for answering web search queries (see [85, 86]). This algorithm constructs a directed graph by associating vertices with web pages, and creating a directed edge for each link. It also adds a large number of low-weight edges by allowing the random walk to move to a random vertex with some small probability at each step. The PageRank score of a web page is then precisely the value of the Perron-Frobenius vector at the associated vertex. Interestingly, this idea was actually proposed by Bonacich [87, 88, 89] in the 1970's as a way of measuring the centrality of nodes in a social network. An analogous measure, using the adjacency matrix, was proposed by Berge [90, Chapter 4, Section 5] for ranking teams in sporting events. Palacios-Huerta and Volij [91] and Altman and Tennenholz [92] have given abstract, axiomatic descriptions of the rankings produced by these vectors.

An related approach to obtaining rankings from directed graphs was proposed by Kleinberg [93]. He suggested using singular vectors of the directed adjacency matrix. Surprising, we are unaware of other combinatorially interesting uses of the singular values or vectors of matrices associated with directed graphs.

To avoid the complications of non-diagonalizable matrices, Chung [94] has defined a symmetric Laplacian matrix for directed graphs. Her definition is inspired by the observation that the degree matrix  $D$  used in the definition of the undirected Laplacian is the diagonal matrix of  $\mathbf{d}$ , which is proportional to the limiting distribution of a random walk on an undirected graph. Chung's Laplacian for directed graphs is constructed by replacing  $\mathbf{d}$  by the Perron-Frobenius vector for the random walk on the graph. Using this Laplacian, she derives analogs of Cheeger's inequality, defining conductance by counting edges by the probability they appear in a random walk [95].

### 16.13 Concluding Remarks

Many fascinating and useful results in Spectral Graph Theory are omitted in this survey. For those who want to learn more, the following books and survey papers take an approach in the spirit of this Chapter: [96, 97, 98, 81, 3, 40]. I also recommend [10, 99, 6, 100, 101].

Anyone contemplating Spectral Graph Theory should be aware that there are graphs with very pathological spectra. Expanders could be considered ex-

amples. But, Strongly Regular Graphs (which only have 3 distinct eigenvalues) and Distance Regular Graphs should also be considered. Excellent treatments of these appear in some of the aforementioned works, and also in [6, 102].

## Bibliography

- [1] Bollobás, B., *Modern graph theory*, Springer-Verlag, New York, 1998.
- [2] Doyle, P. G. and Snell, J. L., *Random Walks and Electric Networks*, Vol. 22 of *Carus Mathematical Monographs*, Mathematical Association of America, 1984.
- [3] Chung, F. R. K., *Spectral Graph Theory*, American Mathematical Society, 1997.
- [4] Hall, K. M., “An r-dimensional quadratic placement algorithm,” *Management Science*, Vol. 17, 1970, pp. 219–229.
- [5] Gould, S. H., *Variational Methods for Eigenvalue Problems*, Dover, 1995.
- [6] Godsil, C., *Algebraic Combinatorics*, Chapman & Hall, 1993.
- [7] van der Holst, H., Lovász, L., and Schrijver, A., “The Colin de Verdière Graph Parameter,” *Bolyai Soc. Math. Stud.*, Vol. 7, 1999, pp. 29–85.
- [8] Horn, R. A. and Johnson, C. R., *Matrix Analysis*, Cambridge University Press, 1985.
- [9] Collatz, L. and Sinogowitz, U., “Spektren endlicher Grafen,” *Abh. Math. Sem. Univ. Hamburg*, Vol. 21, 1957, pp. 63–77.
- [10] Cvetković, D. M., Doob, M., and Sachs, H., *Spectra of Graphs*, Academic Press, 1978.
- [11] Bapat, R. B. and Raghavan, T. E. S., *Nonnegative Matrices and Applications*, No. 64 in Encyclopedia of Mathematics and its Applications, Cambridge University Press, 1997.
- [12] Hoffman, A. J., “On the Polynomial of a Graph,” *The American Mathematical Monthly*, Vol. 70, No. 1, 1963, pp. 30–36.
- [13] Fiedler, M., “Algebraic connectivity of graphs,” *Czechoslovak Mathematical Journal*, Vol. 23, No. 98, 1973, pp. 298–305.
- [14] Fiedler, M., “A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory,” *Czechoslovak Mathematical Journal*, Vol. 25, No. 100, 1975, pp. 618–633.

- [15] van der Holst, H., “A Short Proof of the Planarity Characterization of Colin de Verdière,” *Journal of Combinatorial Theory, Series B*, Vol. 65, No. 2, 1995, pp. 269 – 272.
- [16] Donath, W. E. and Hoffman, A. J., “Algorithms for Partitioning Graphs and Computer Logic Based on Eigenvectors of Connection Matrices,” *IBM Technical Disclosure Bulletin*, Vol. 15, No. 3, 1972, pp. 938–944.
- [17] Donath, W. E. and Hoffman, A. J., “Lower Bounds for the Partitioning of Graphs,” *IBM Journal of Research and Development*, Vol. 17, No. 5, Sept. 1973, pp. 420–425.
- [18] Barnes, E. R., “An Algorithm for Partitioning the Nodes of a Graph,” *SIAM Journal on Algebraic and Discrete Methods*, Vol. 3, No. 4, 1982, pp. 541–550.
- [19] Simon, H. D., “Partitioning of Unstructured Problems for Parallel Processing,” *Computing Systems in Engineering*, Vol. 2, 1991, pp. 135–148.
- [20] Pothen, A., Simon, H. D., and Liou, K.-P., “Partitioning Sparse Matrices with Eigenvectors of Graphs,” *SIAM Journal on Matrix Analysis and Applications*, Vol. 11, No. 3, 1990, pp. 430–452.
- [21] Shi, J. B. and Malik, J., “Normalized Cuts and Image Segmentation,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, Aug. 2000, pp. 888–905.
- [22] Ng, A. Y., Jordan, M. I., and Weiss, Y., “On Spectral Clustering: Analysis and an algorithm,” *Adv. in Neural Inf. Proc. Sys. 14*, 2001, pp. 849–856.
- [23] Wei, Y.-C. and Cheng, C.-K., “Ratio cut partitioning for hierarchical designs,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 10, No. 7, Jul 1991, pp. 911–921.
- [24] Hagen, L. and Kahng, A. B., “New Spectral Methods for Ratio Cut Partitioning and Clustering,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 11, 1992, pp. 1074–1085.
- [25] Cheeger, J., “A lower bound for smallest eigenvalue of the Laplacian,” *Problems in Analysis*, Princeton University Press, 1970, pp. 195–199.
- [26] Sinclair, A. and Jerrum, M., “Approximate Counting, Uniform Generation and Rapidly Mixing Markov Chains,” *Information and Computation*, Vol. 82, No. 1, July 1989, pp. 93–133.
- [27] Lawler, G. F. and Sokal, A. D., “Bounds on the  $L^2$  Spectrum for Markov Chains and Markov Processes: A Generalization of Cheeger’s Inequality,” *Transactions of the American Mathematical Society*, Vol. 309, No. 2, 1988, pp. 557–580.

- [28] Alon, N. and Milman, V. D., “ $\lambda_1$ , Isoperimetric inequalities for graphs, and superconcentrators,” *J. Comb. Theory, Ser. B*, Vol. 38, No. 1, 1985, pp. 73–88.
- [29] Alon, N., “Eigenvalues and expanders,” *Combinatorica*, Vol. 6, No. 2, 1986, pp. 83–96.
- [30] Dodziuk, J., “Difference Equations, Isoperimetric Inequality and Transience of Certain Random Walks,” *Transactions of the American Mathematical Society*, Vol. 284, No. 2, 1984, pp. 787–794.
- [31] Varopoulos, N. T., “Isoperimetric inequalities and Markov chains,” *Journal of Functional Analysis*, Vol. 63, No. 2, 1985, pp. 215 – 239.
- [32] Mihail, M., “Conductance and Convergence of Markov Chains—A Combinatorial Treatment of Expanders,” *30th Annual IEEE Symposium on Foundations of Computer Science*, 1989, pp. 526–531.
- [33] Leighton, T. and Rao, S., “Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms,” *Journal of the ACM*, Vol. 46, No. 6, Nov. 1999, pp. 787–832.
- [34] Arora, S., Rao, S., and Vazirani, U., “Expander flows, geometric embeddings and graph partitioning,” *J. ACM*, Vol. 56, No. 2, 2009, pp. 1–37.
- [35] Khandekar, R., Rao, S., and Vazirani, U., “Graph partitioning using single commodity flows,” *J. ACM*, Vol. 56, No. 4, 2009, pp. 1–15.
- [36] Sherman, J., “Breaking the Multicommodity Flow Barrier for  $O(\sqrt{\log n})$ -approximations to Sparsest Cut,” *Proceedings of the 50th IEEE Symposium on Foundations of Computer Science*, 2009, pp. 363 – 372.
- [37] Arora, S., Hazan, E., and Kale, S., “ $O(\sqrt{\log n})$  Approximation to Sparsest Cut in  $\tilde{O}(n^2)$  Time,” *45th IEEE Symposium on Foundations of Computer Science*, 2004, pp. 238–247.
- [38] Arora, S. and Kale, S., “A combinatorial, primal-dual approach to semidefinite programs,” *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, 2007, pp. 227–236.
- [39] Orecchia, L., Schulman, L. J., Vazirani, U. V., and Vishnoi, N. K., “On partitioning graphs via single commodity flows,” *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, 2008, pp. 461–470.
- [40] Lovász, L., “Random walks on graphs: a survey,” *Combinatorics, Paul Erdős is Eighty*, Vol. 2, edited by T. S. D. Miklós, V. T. Sos, Janos Bolyai Mathematical Society, Budapest, 1996, pp. 353–398.
- [41] Impagliazzo, R. and Zuckerman, D., “How to recycle random bits,” *30th annual IEEE Symposium on Foundations of Computer Science*, 1989, pp. 248–253.

- [42] Karp, R. M., Pippenger, N., and Sipser, M., “A time randomness trade-off,” *AMS Conf. on Probabilistic Computational Complexity*, Durham, New Hampshire, 1985.
- [43] Ajtai, M., Komlós, J., and Szemerédi, E., “Deterministic Simulation in LOGSPACE,” *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, 1987, pp. 132–140.
- [44] Alon, N., Bruck, J., Naor, J., Naor, M., and Roth, R. M., “Construction of Asymptotically Good Low-Rate Error-Correcting Codes through Pseudo-Random Graphs,” *IEEE Transactions on Information Theory*, Vol. 38, No. 2, March 1992, pp. 509–516.
- [45] Sipser, M. and Spielman, D., “Expander codes,” *IEEE Transactions on Information Theory*, Vol. 42, No. 6, Nov 1996, pp. 1710–1722.
- [46] Zemor, G., “On expander codes,” *IEEE Transactions on Information Theory*, Vol. 47, No. 2, Feb 2001, pp. 835–837.
- [47] Barg, A. and Zemor, G., “Error exponents of expander codes,” *IEEE Transactions on Information Theory*, Vol. 48, No. 6, Jun 2002, pp. 1725–1729.
- [48] Guruswami, V., “Guest column: error-correcting codes and expander graphs,” *SIGACT News*, Vol. 35, No. 3, 2004, pp. 25–41.
- [49] Pippenger, N., “On networks of noisy gates,” *Proceedings of the 26th Ann. IEEE Symposium on Foundations of Computer Science*, 1985, pp. 30–38.
- [50] Pippenger, N., “Self-Routing Superconcentrators,” *Journal of Computer and System Sciences*, Vol. 52, No. 1, Feb. 1996, pp. 53–60.
- [51] Tanner, R. M., “Explicit construction of concentrators from generalized n-gons,” *SIAM J. Algebraic Discrete Methods*, Vol. 5, 1984, pp. 287–293.
- [52] Beigel, R., Margulis, G., and Spielman, D. A., “Fault diagnosis in a small constant number of parallel testing rounds,” *SPAA '93: Proceedings of the fifth annual ACM symposium on Parallel algorithms and architectures*, ACM, New York, NY, USA, 1993, pp. 21–29.
- [53] Alon, N. and Chung, F., “Explicit Construction of Linear Sized Tolerant Networks,” *Discrete Mathematics*, Vol. 72, 1988, pp. 15–19.
- [54] Hoory, S., Linial, N., and Wigderson, A., “Expander Graphs and Their Applications,” *Bulletin of the American Mathematical Society*, Vol. 43, No. 4, 2006, pp. 439–561.
- [55] Nilli, A., “On the second eigenvalue of a graph,” *Discrete Math*, Vol. 91, 1991, pp. 207–210.

- [56] Margulis, G. A., “Explicit group theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators,” *Problems of Information Transmission*, Vol. 24, No. 1, July 1988, pp. 39–46.
- [57] Lubotzky, A., Phillips, R., and Sarnak, P., “Ramanujan Graphs,” *Combinatorica*, Vol. 8, No. 3, 1988, pp. 261–277.
- [58] Kahale, N., “Eigenvalues and expansion of regular graphs,” *J. ACM*, Vol. 42, No. 5, 1995, pp. 1091–1106.
- [59] Capalbo, M., Reingold, O., Vadhan, S., and Wigderson, A., “Randomness conductors and constant-degree lossless expanders,” *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, 2002, pp. 659–668.
- [60] Spielman, D. A. and Teng, S.-H., “Spectral partitioning works: Planar graphs and finite element meshes,” *Linear Algebra and its Applications*, Vol. 421, No. 2–3, 2007, pp. 284 – 305, Special Issue in honor of Miroslav Fiedler.
- [61] Kelner, J. A., “Spectral partitioning, eigenvalue bounds, and circle packings for graphs of bounded genus,” *SIAM J. Comput.*, Vol. 35, No. 4, 2006, pp. 882–902.
- [62] Biswal, P., Lee, J., and Rao, S., “Eigenvalue bounds, spectral partitioning, and metrical deformations via flows,” *Journal of the ACM*, 2010, to appear.
- [63] Kelner, J. A., Lee, J., Price, G., and Teng, S.-H., “Higher Eigenvalues of Graphs,” *Proceedings of the 50th IEEE Symposium on Foundations of Computer Science*, 2009, pp. 735–744.
- [64] Vaidya, P. M., “Solving linear equations with symmetric diagonally dominant matrices by constructing good preconditioners.” Unpublished manuscript UIUC 1990. A talk based on the manuscript was presented at the IMA Workshop on Graph Theory and Sparse Matrix Computation, October 1991, Minneapolis.
- [65] Bern, M., Gilbert, J., Hendrickson, B., Nguyen, N., and Toledo, S., “Support-graph preconditioners,” *SIAM Journal on Matrix Analysis and Applications*, Vol. 27, No. 4, 2006, pp. 930–951.
- [66] Boman, E. G. and Hendrickson, B., “Support Theory for Preconditioning,” *SIAM Journal on Matrix Analysis and Applications*, Vol. 25, No. 3, 2003, pp. 694–717.
- [67] Spielman, D. A. and Teng, S.-H., “Nearly-Linear Time Algorithms for Preconditioning and Solving Symmetric, Diagonally Dominant

Linear Systems,” *CoRR*, Vol. abs/cs/0607105, 2009, Available at <http://www.arxiv.org/abs/cs.NA/0607105>.

- [68] Diaconis, P. and Stroock, D., “Geometric Bounds for Eigenvalues of Markov Chains,” *The Annals of Applied Probability*, Vol. 1, No. 1, 1991, pp. 36–61.
- [69] Guattery, S., Leighton, T., and Miller, G. L., “The Path Resistance Method for Bounding the Smallest Nontrivial Eigenvalue of a Laplacian,” *Combinatorics, Probability and Computing*, Vol. 8, 1999, pp. 441–460.
- [70] Guattery, S. and Miller, G. L., “Graph Embeddings and Laplacian Eigenvalues,” *SIAM Journal on Matrix Analysis and Applications*, Vol. 21, No. 3, 2000, pp. 703–723.
- [71] Wilf, H. S., “The Eigenvalues of a Graph and its Chromatic Number,” *J. London math. Soc.*, Vol. 42, 1967, pp. 330–332.
- [72] Hoffman, A. J., “On eigenvalues and colorings of graphs,” *Graph Theory and its Applications*, Academic Press, New York, 1970, pp. 79–92.
- [73] Trevisan, L., “Max cut and the smallest eigenvalue,” *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, 2009, pp. 263–272.
- [74] McSherry, F., “Spectral Partitioning of Random Graphs,” *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, 2001, pp. 529–537.
- [75] Demmel, J., *Applied Numerical Linear Algebra*, SIAM, 1997.
- [76] Boppana, R. B., “Eigenvalues and graph bisection: an average-case analysis,” *Proc. 28th IEEE Symposium on Foundations of Computer Science*, 1987, pp. 280–285.
- [77] Füredi, Z. and Komlós, J., “The eigenvalues of random symmetric matrices,” *Combinatorica*, Vol. 1, No. 3, 1981, pp. 233–241.
- [78] Vu, V., “Spectral Norm of Random Matrices,” *Combinatorica*, Vol. 27, No. 6, 2007, pp. 721–736.
- [79] Alon, N., Krivelevich, M., and Vu, V. H., “On the concentration of eigenvalues of random symmetric matrices,” *Israel Journal of Mathematics*, Vol. 131, No. 1, 2002, pp. 259–267.
- [80] Alon, N. and Kahale, N., “A Spectral Technique for Coloring Random 3-Colorable Graphs,” *SIAM Journal on Computing*, Vol. 26, 1997, pp. 1733–1748.

- [81] Kannan, R. and Vempala, S., “Spectral Algorithms,” *Foundations and Trends in Theoretical Computer Science*, Vol. 4, No. 3-4, 2008, pp. 132–288.
- [82] Batson, J. D., Spielman, D. A., and Srivastava, N., “Twice-Ramanujan sparsifiers,” *Proceedings of the 41st Annual ACM Symposium on Theory of computing*, 2009, pp. 255–262.
- [83] Spielman, D. A. and Srivastava, N., “Graph sparsification by effective resistances,” *SIAM Journal on Computing*, 2010, To appear. Preliminary version appeared in the Proceedings of the 40th Annual ACM Symposium on Theory of Computing.
- [84] Spielman, D. A. and Woo, J., “A Note on Preconditioning by Low-Stretch Spanning Trees,” *CoRR*, Vol. abs/0903.2816, 2009, Available at <http://arxiv.org/abs/0903.2816>.
- [85] Page, L., Brin, S., Motwani, R., and Winograd, T., “The PageRank Citation Ranking: Bringing Order to the Web,” Tech. rep., Computer Science Department, Stanford University, 1998.
- [86] Langville, A. N. and Meyer, C. D., *Google’s PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press, Princeton, NJ, USA, 2006.
- [87] Bonacich, P., “Technique for Analyzing Overlapping Memberships,” *Sociological Methodology*, Vol. 4, 1972, pp. 176–185.
- [88] Bonacich, P., “Factoring and Weighting Approaches to Status Scores and Clique Identification,” *Journal of Mathematical Sociology*, Vol. 2, 1972, pp. 113–120.
- [89] Bonacich, P., “Power and Centrality: A family of Measures,” *American Journal of Sociology*, Vol. 92, 1987, pp. 1170–1182.
- [90] Berge, C., *Graphs*, North-Holland, 1985.
- [91] Palacios-Huerta, I. and Volij, O., “The Measurement of Intellectual Influence,” *Econometrica*, Vol. 72, No. 3, 2004, pp. 963–977.
- [92] Altman, A. and Tennenholz, M., “Ranking systems: the PageRank axioms,” *Proceedings of the 6th ACM Conference on Electronic Commerce*, 2005, pp. 1–8.
- [93] Kleinberg, J., “Authoritative sources in a hyperlinked environment,” *Journal of the ACM*, Vol. 48, 1999, pp. 604–632.
- [94] Chung, F., “The Diameter and Laplacian Eigenvalues of Directed Graphs,” *The Electronic Journal of Combinatorics*, Vol. 13, No. 1, 2006.

- [95] Chung, F., “Laplacians and the Cheeger Inequality for Directed Graphs,” *Annals of Combinatorics*, Vol. 9, No. 1, 2005, pp. 1–19.
- [96] Mohar, B. and Poljak, S., “Eigenvalues in combinatorial optimization,” *Combinatorial and graph-theoretical problems in linear algebra*, IMA Volumes in Mathematics and Its Applications, Springer-Verlag, 1993, pp. 107–151.
- [97] Mohar, B., “The Laplacian spectrum of graphs,” *Graph Theory, Combinatorics, and Applications*, Wiley, 1991, pp. 871–898.
- [98] Brouwer, A. E. and Haemers, W. H., “Spectra of Graphs,” Electronic Book. Available at <http://www.win.tue.nl/~aeb/ipm.pdf>.
- [99] Godsil, C. and Royle, G., *Algebraic Graph Theory*, Graduate Texts in Mathematics, Springer, 2001.
- [100] Biggs, N. L., *Algebraic graph theory*, Cambridge Tracts in Math., Cambridge University Press, London, New York, 1974.
- [101] Brualdi, R. A. and Ryser, H. J., *Combinatorial Matrix Theory*, Cambridge University Press, New York, 1991.
- [102] Brouwer, A. E., Cohen, A. M., and Neumaier, A., *Distance-Regular Graphs*, Springer Verlag, 1989.